第36卷 第19期 计 算 机 工 程 2010年10月
Vol.36 No.19 Computer Engineering October 2010
·软件技术与数据库· 文章编号：1000—3428(2010)19—0093—03 文献标识码：A 中图分类号：TP391

基于同义词链的中文关键词提取算法

张颖颖，谢 强，丁秋林
(南京航空航天大学信息科学与技术学院，南京 210016)

摘 要：针对传统中文关键词提取对语义和同义词的不重视而导致的精确度和召回率低的问题，提出基于同义词链的中文关键词提取算法。利用上下文窗口和消歧算法解决词语在上下文中的语义问题，利用文档中的同义词构建同义词链，简化候选词的选取。根据同义词链的特征，得到相应的权重计算公式，对候选词进行过滤。实验结果表明，该算法在同义词较多的文档中精确度和召回率有较大的提高，平均性能也有明显改善。

关键词：关键词提取；同义词链；语义；消歧

Chinese Keyword Extraction Algorithm Based on Synonym Chains

ZHANG Ying-ying, XIE Qiang, DING Qiu-lin
(College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

【Abstract】To solve the problem of low precision rate and recall rate in the traditional Chinese keyword extraction resulted from indifference of semantic and synonym, Chinese keyword extraction algorithm based on synonym chains is proposed. In the algorithm, the problem of word semantic in the context is solved by using the word of context window and word sense disambiguation algorithm. Synonym chains are built by using synonym of the document which simplifies the selection of candidate words, and the weight formula of keyword which can filter candidate word is brought out by the characteristics of synonym chains. Experimental results show that the proposed algorithm has more precision rate and recall rate in the document with much more synonym, and the average performance can be obviously improved.

【Key words】keyword extraction; synonym chains; semantic; disambiguation

1 概述

关键词表征文档的重要信息和核心内容，便于得到文档的摘要信息和检索具体文档。传统的关键词提取一般采用人工提取，而人工提取关键词非常费时，随着文档数量的剧增，人工提取关键词越来越不能满足实际应用的需求。因此，如何自动提取关键词成为目前计算机领域的一个研究热点。

现有的关键词自动提取算法可以分为3大类：

(1)基于统计的方法，该方法简单易行不需要复杂的训练过程，比如基于词共线的方法等[1]。

(2)基于规则的方法，国外已经建立了一些实用或实验系统，采用朴素贝叶斯技术对短语离散的特征值进行训练，获取模型的权值，开发了系统KEA；国内同样利用朴素贝叶斯模型对中文关键词提取进行了研究。这2类方法都是从频度或规则上提取关键词，没有考虑词的语义、词性等信息，精确度不高。

(3)基于自然语言理解的方法[2-4]。该方法主要利用词义或语义和词性特征来提取关键词，能从文档中提取出较高正确率的关键词，已成为自动提取关键词的主要研究方向。

文献[2]以词语的权重公式为中心，利用遗传算法训练、优化公式中参数的方法提取关键词，但未对文档中的同义词现象进行处理。人工提取关键词时，不仅考虑文档的概念层，还对文档的理解层有比较深入的考虑。已有的自动提取关键词方法主要从概念层进行提取，对于理解层考虑较少。文献[3]利用文档的语义信息提取关键词，考虑词汇的理解层，提取用词义代替词，通过消歧算法和上下文得到候选词的词义，然后进行词合并、特征提取。但算法是针对英文文献进行的关键词提取，采用已有的消歧算法精度不够高。文献[4]利用词汇链提取关键词，通过计算词汇相似度构建词汇链，然后结合词频和区域特征进行关键词选择。该方法构建的词汇链对候选词的过滤有很好的作用，但该算法在构建词汇链时，对该词的所有词义进行词义相似度计算，没有考虑该词在文档上下文的信息，也没有考虑文档中的同义词。

因此，本文提出了一种基于同义词链(Synonym Chains, SC)的中文关键词提取算法(Chinese Keyword Extraction based on Synonym Chains, CKESC)。在CKESC算法中，对消歧算法进行改进，提高利用上下文信息获取词义的精度。然后利用词义相似度计算公式，组建SC，得到候选词。最后利用改进的权重计算公式过滤候选词，提取关键词。

2 CKESC算法

2.1 CKESC算法思想

CKESC算法主要包括2个步骤：候选词选取和候选词过滤。候选词选取最重要的是构建同义词链SC。构建SC时，首先对文档进行分词，选出粗候选词集，根据词义相似度和上下文窗口得到粗候选词集上下文中的词义，利用词义相似度计算出粗候选词集中相似的词语构成SC。然后对SC进行

作者简介：张颖颖(1984－)，女，硕士研究生，主研方向：知识工程，信息系统集成，人机交互；谢 强，副教授、博士；丁秋林，教授、博士生导师

收稿日期：2010-04-10 **E-mail：**winying_0329@163.com

分析，选出符合要求的关键词。

算法流程如图 1 所示。

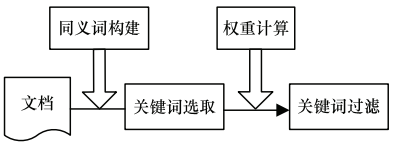

图 1　CKESC 算法流程

2.2　同义词链的构建

同义词链是指文档中根据上下文信息确定词义相同或相似的词的集合。基本思想是：根据上下文信息，利用词义相似度确定词汇在具体语境下的词义，用该词义代替该词，利用义项相似度计算得到一系列的同义词，组成同义词链。相比计算所有的词义相似度来说，该方法精确度较高。

2.2.1　基于知网的词义相似度计算

词义相似度计算是构建同义词链的基础。知网把词汇语义的描述定义为义项(概念)。每一个词有一个或多个词义，所以知网中每一个词有一个或几个义项表示。义项是用一种“知识表示语言”来表示的，这种“知识表示语言”所用的词汇就是义原。义原是知网中最小的有意义单位[5]。

设中文词语 W_1 和 W_2，如果 W_1 有 n 个义项(概念)：S_{11}, $S_{12},\cdots,S_{1n}$，W_2 有 m 个义项(概念)：$S_{21},S_{22},\cdots,S_{2n}$，规定 W_1 和 W_2 的词语相似度为各个概念的相似度的最大值[5]：

$$Sim(W_1,W_2)=\max_{i=1,2,\cdots,n,j=1,2,\cdots,m} SimS(S_{1i},S_{2j}) \tag{1}$$

设义项 S_1 和 S_2，定义 P_1 和 P_2 分别为 S_1 和 S_2 的某个义原。则 2 个义项的语义相似度为[5]：

$$SimS(S_1,S_2)=\sum_{i=1}^{4}\beta_i\prod_{j=1}^{i}SimP_j(P_1,P_2) \tag{2}$$

其中，$\beta_i\,(1\leqslant i\leqslant 4)$ 是可调节的参数，且有：$\beta_1+\beta_2+\beta_3+\beta_4=1,\beta_1\geqslant\beta_2\geqslant\beta_3\geqslant\beta_4$。

$$SimP(p_1,p_2)=\frac{\alpha}{d+\alpha} \tag{3}$$

其中，d 是 P_2 和 P_2 在义原层次体系中的路径长度，是一个正整数；α 是一个可调节的参数[5]。

2.2.2　消歧算法的改进

消歧的目的是为了确定粗候选词在上下文中的词义。对粗候选词集进行词义相似度计算时，会减少计算的难度，同时提高计算的精确度。消歧算法认为词义是由其周围的单词决定的，因此消歧算法计算被消歧的词的所有可能词义和它周围单词的语义相关度，并且认为相关度最大的词义就是该词在现在语境下的正确词义。将消歧词周围的单词称为上下文窗口，用 W 表示。消歧词 V 词义的确定与 W 的大小和 W 的内容有关系。假设 W 的大小为 n，由单词集合 C 组成，则 W 可以表示为：$W=\sum_{i=1}^{i=n}C_i$。设 V 有 m 个义项 $S_1,S_2,\cdots,S_m$，对于义项 S_i 的义原用 $P_{i1},P_{i2},\cdots,P_{in}$ 来表示。确定 V 的词义，即确定 V 在上下文中的义项，也就是 S_i 在 W 中词义相似度最大的某个义项。则式(1)可以表示为：

$$Sim(V,W)=\max_{i=1,2,\cdots,n}Sim_1(S_i,W)=\max_{i=1,2,\cdots,n}\sum_{j=1}^{j=n}SimS(S_i,C_j) \tag{4}$$

利用式(4)，确定消歧词 V 在上下文中的词义 S_i。当粗候选词集中所有词的词义确定后，构建同义词链时，设粗候选词集中任意 2 个词 V_1、V_2，由式(4)确定 V_1、V_2 在上下文中的词义为 S_1、S_2，则 V_1、V_2 的词义相似度可以表示为：

$$Sim(V_1,V_2)=SimS(S_1,S_2) \tag{5}$$

2.2.3　同义词链的构建算法

构建同义词链的目的是为了将文档中表达相同意思的词组成一个链，即将文档抽象为多个 SC。在构建同义词链时，首先要做的是对文档进行分词，进行词性标注。然后选择名词和动词作为粗候选词集，根据粗候选词集所在上下文环境确定该粗候选词在该文档的词义。再对这些候选词进行词义相似度计算，得到同义词链。定义文档为 T，$T=\bigcup_{i=1}^{\infty}T_i$，$T_i$ 代表文档中的句子，句子由标点符号和一些字符串组成。

具体算法如下：

(1)分词

对文档 T 进行分词和词性标注。为了能在分词的同时识别出复合词，采用海量分词方法。该分词方法可以加载用户词典，方便用户自定义复合词。在分词的过程中，标点符号保留，以便上下文窗口 W 的选择。

(2)粗候选词集的选取

定义粗候选词集 D 为六元组：$D=(w,n,t,p,s,f)$。其中，w 为词汇；n 为词性；t 为词频，即 w 出现的次数；p 代表 w 出现在文档中的位置；s 为义项；f 为是否已构成同义词链中的元素，初始值为 0。对词性标注后的文档提取出名词和动词，组成粗候选词集 D，此时 s 为 w 在知网中所有的义项。

(3)粗候选词集 D 词义的确定

利用消歧算法确定每个粗候选词集 D 中的 s。消歧算法中 W 的选择，首先是选择粗候选词所在标点符号之间的词或多个标点符合之间的词作为 W 的内容。对 D 中的 s 进行消歧，最后 s 为某一个确定的义项。

(4)SC 的构建

根据 D 中 t 的大小对 D 中所有的词集按降序排列，假设排序后 $D=\{D_1,D_2,\cdots,D_n\}$。由于 D 中义项 s 都已确定，因此在计算粗候选词集词义相似度时，只要利用式(2)和式(3)即可。将 D 中 f=0 的 $D_i\ (1\leqslant i\leqslant n)$ 作为 SC 集中某个链 SC_i 的表头 L_1，将 D_i 与 $D_j\ (i\leqslant j\leqslant n)$ 进行词义相似度计算，将词义相似度大于某一阈值 σ 的词插入 SC_i 中，并将 D_j 中的 f 改为 1。一直重复这个过程直到 D 中所有的 f 为 1，组链结束。最后按链长降序排列，设排序结果为 $SC=\bigcup_{i=1}^{i=n}SC_i$。

2.3　候选词的提取

关键词提取可以分为 2 个步骤：候选词的提取和过滤。在过滤阶段常采用统计的方法筛选候选词的主要特征：*TF*×*IDF* 和 *First occurrence*。*TF*×*IDF* 表示一个词在文档中出现的频率，并且和训练集中出现该词的文档数作比较。如果一个词语的 *TF*×*IDF* 值越高，表示这个词语越有可能是关键词。*First occurrence* 表示词语在文章中第一次出现的平均位置。出现在开头和结尾的词语是关键词的概率比较大。如果该词重复以不同的形式出现，说明该词是关键词的概率比较大。本文采用同义词链的方法提取关键词，不仅考虑了候选词在文档中的位置信息和出现的次数，还考虑候选词的同义词信息。所以，为了能更好地对候选词进行过滤，在进行候选词提取时，记录候选词的一些基本特征。具体步骤如下：

步骤 1　按照 2.2.3 节构建同义词链 SC。

步骤 2　根据粗候选词集 SC 进行候选词的提取：设候选词集为 CW，定义 CW 为五元组：$CW=(w,t,p,ap,l)$。其中，w 为该词，定义为粗候选词集中 SC_i 表头 L_{i1} 中的 w；t 为词频；p 为位置，均为 L_{i1} 中相应 w 的 t、p；ap 平均位置记

为 SC_i 中所有词中 p 的平均值；l 为 TC_i 的链长。

2.4 候选词的过滤

综合 2.3 节候选词的特征项和 CW 记录的内容，根据权重计算式对候选词进行过滤，定义权重计算式为：

$$Weight_i(w)=\alpha\times l+\beta\times(tf\times idf)+\gamma\times p\times+\delta\times ap \tag{6}$$

其中，$\alpha,\beta,\gamma,\delta$ 为调节因子，最大值为 1。为了保证平衡设定 $\alpha+\beta+\gamma+\delta=1$，$tf\times idf$ 为传统方法常用的统计量，$tf=\dfrac{t}{all\_num}$；$idf=-\mathrm{lb}\dfrac{doc\_num}{all\_doc}$；$all\_num$ 为该文档的词语总数；doc_num 为训练语料中出现该词的文档数；all_doc 为训练语料中文档的数目。式(6)归一化处理后为：

$$Weight_i(w)=\frac{\alpha\times l+\beta\times(tf\times idf)+\gamma\times p+\delta\times ap}{l+tf\times idf+p+ap} \tag{7}$$

根据式(7)，计算出候选词集 CW 中所有 w 的权重，然后按照权重大小进行降序排列，根据需求选定相应数量的候选词作为关键词。

3 实验结果与分析

3.1 实验结果的评估方法

一般关键词提取算法的评估都是与人工提取进行比较，但其本身也有自己的评价函数，通常通过召回率(*Recall*)、精确度(*Precious*)和平衡两者的综合指标 *F-measure* 来评价函数的性能。

$$Recall=\frac{\alpha_{\text{corect}}}{\gamma_{\text{manual-all}}},\ Precision=\frac{\alpha_{\text{correct}}}{\beta_{\text{auto-all}}},$$

$$F-measure=\frac{2\times Precision\times Recall}{Precision+Recall}$$

其中，α_{correct} 为自动提取的关键词和人工提取的关键词完全匹配的数目；$\beta_{\text{cuto-all}}$ 为自动提取关键词的数目；$\gamma_{\text{manual-all}}$ 为手工提取关键词的数目。

3.2 实验结果

为了便于比较实验结果，在网上选择 300 篇左右的带有关键词的文章作为该算法的语料集，对每一篇文档进行分词和词性标注后，构建同义词链。在分词前，在用户自定义词典中加入复合词或专业词语便于复合词的提取；利用知网计算词义相似度，选取 $\alpha=1.60$，$\beta_1=0.50$，$\beta_2=0.20$，$\beta_3=0.17$，$\beta_4=0.13$ 对粗候选词集词义的确定。

为了能精确地构建同义词链，词义相似度阈值 σ 设定为 1，这样可以构建类似“快乐-高兴-愉快”等的同义词链，因为“快乐”和“高兴”的词义相似度为 1.00，“快乐”和“愉快”的词义相似度为 1.00，三者可以构成同义词链，而“快乐”和“悲伤”的词义相似度为 0.286，所以两者不能构成同义词链。同义词链构建好后，进行候选词的选取，再对候选词集进行权重计算时，$\alpha,\beta,\gamma,\delta$ 的选择根据文档的形式进行设定。如果该文章同义词比较多，α 的值可以比较大；如果文档的格式比较固定，一般特征词出现在开头或结尾等，则 β 值可以较大些。可以通过不断地改变 $\alpha,\beta,\gamma,\delta$，求得最佳的关键词。实验中选取 $\alpha=0.55$，$\beta=0.25$，$\gamma=0.1$，$\delta=0.1$。当所有的文档构建完同义词链后，可以看出有些文档同义词较多，有的较少。根据同义词的多少进行统计，如图 2 所示。可以发现：对同义词比较多的文档来说，本文方法在召回率和准确率方面有显著的提高，且随着提取关键词数目的增多，准确率也越来越精确。对同义词较少的文档，本文方法效果一般，和基于词汇链的方法精确率差不多，而本文方法的平均性能相对词汇链方法在召回率和准确率方面有所提高，如表 1 所示。

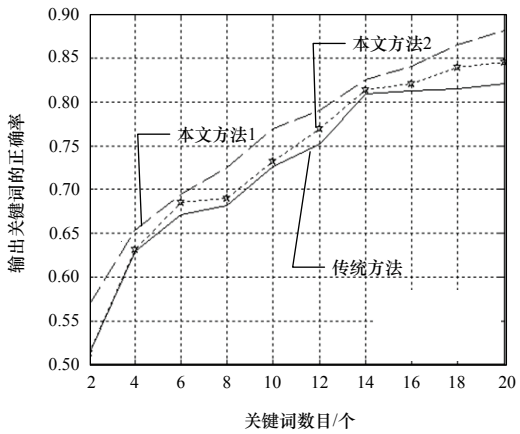

图 2 输出关键词正确率对比

表 1 本文方法和词汇链方法性能的比较

方法	召回率	准确率	*F-measure*
本文方法	0.842 7	0.856 2	0.849 4
词汇链方法	0.812 7	0.821 6	0.817 1

4 结束语

本文针对中文关键词的自动提取，提出一种基于同义词链的中文关键词提取算法。该算法利用同义词链的构建来提取和过滤候选词，可以充分利用文档中的同义词，更有效地提取候选词，同时省略候选词提取后的合并。在构建同义词链时，考虑语境，有助于解决关键词提取中的语义问题。最后通过结果分析证明该算法的精确度和召回率相对传统方法有明显提高。

参考文献

[1] Matsuo Y, Ishizuka M. Keyword Extraction from a Single Document Using Co-occurrence Statistical Information[J]. International Journal on Artificial Intelligence Tools, 2004, 13(1): 157-169.

[2] 张 虹. 基于自动文本分类的关键词抽取算法[J]. 计算机工程, 2009, 35(12): 145-147.

[3] Medelyan O, Witten I H. Thesaurus Based Automatic Keyphrase Indexing[C]//Proc. of the Joint Conference on Digital Libraries. Chapel Hill, NC, USA: [s. n.], 2006: 296-297.

[4] Ercan G, Ciekli I. Using Lexical Chains for Keyword Extraction[J]. Information Processing and Management, 2007, 43(6): 1705-1714.

[5] 刘 群, 李素建. 基于《知网》的词汇语义相似度计算[EB/OL]. (2009-11-19). http://wenku.baidu.com/view/22590b4c2e3f5727a5e9626a. html.

编辑 顾逸斐