

基于 HowNet 词汇相关性的文本聚类

周爱武, 汪贤惠, 刘慧婷

(安徽大学 计算机科学与技术学院, 安徽 合肥 230601)

摘 要: 提出基于 HowNet 词汇相关性的聚类方法, 该方法通过统计学的 Z 分数来消除孤立点, 根据文档的稀疏分布程度, 选择初始聚类中心, 并且考虑词与词的相关性和词与词的语义相似性, 使得文本聚类的精确度得到了提升, 时间消耗上也大大减少。

关键词: 知网; 词汇相关性; Z 分数; 义原; 孤立点; 聚类

中图分类号: TP301.6

文献标识码: A

文章编号: 1000-7180(2015)04-0090-04

Vocabulary Correlation Text Clustering Based on HowNet

ZHOU Ai-wu, WANG Xian-hui, LIU Hui-ting

(College of Computer Science and Technology, Anhui University, Hefei 230601, China)

Abstract: The vocabulary correlation method based on HowNet is given out in this paper. This method removes isolated points through statistical z -score. The initial clustering centers are selected according to the sparse division of documents. The relevance and semantic similarity of words are studied, which improve the precision of document clustering and reduce the time consumption.

Key words: HowNet; vocabulary correlation; z -score; sememes; isolated point; clustering

1 引言

文本聚类是在没有任何先验知识的条件下, 将大量文本按照一定的相似进行归类。与结构化的数据挖掘对象不同, 文本聚类所处理的对象主要是一些非结构化, 或者是半结构化的数据, 这些数据具有很大的稀疏性^[1], 由于数据的稀疏性导致聚类所花费的时间非常长。传统的文本聚类算法将文本进行分词, 除停用词, 然后计算每个词的 $tfidf$ 值(tf (term frequency)词频, idf (inverse document frequency)逆向文档词频)。这种方法没有考虑词与词的相关性(词与词反映同一个主题, 或者词与词出现在同一篇文章的概率非常大), 很有可能将相关的两个词当着无关词, 这在长文档中影响也许不是很大, 但是对于短文本的影响则是非常巨大的。如果不考虑词与词的相关性, 也就意味着很有可能把两篇相关的文档当成无关的文档来考虑。还有传统的聚类算法, 对

初始聚类中心选择很是盲目, 初始聚类中心的选择不当, 容易产生局部最优解, 严重影响聚类结果的精度^[2]。同样孤立点的存在也严重影响聚类的结果。针对上述提到的问题, 本文提出解决办法, 即基于 HowNet 词汇相关性的聚类方法考虑词与词的相关性, 为要聚类的文档构造了一个替代词库, 这个替代词库的每一行是由相关度非常大的一组词组成的。这个替代词库类似于领域词典, 但是它的优势是具有很强的针对性, 它是由要进行聚类的文档内容构建成的, 相比领域词典, 它非常小, 在处理时, 时间消耗方面大大缩小。使用这个替代词库对每篇文档进行分词, 并将分词得到结果进行相应词的替换。处理孤立点是使用统计学中的 Z 分数, 相比传统除去孤立点的方法, 该方法相对比较智能。在初始聚类中心的选择上, 本文根据要进行聚类的文档的稀疏分布程度, 进行初始聚类中心的选择。

收稿日期: 2014-07-23; 修回日期: 2014-09-15

基金项目: 国家自然科学基金项目(61202227)

2 相关介绍

2.1 HowNet

HowNet 是用汉语与英语所代表的概念对象,以表示概念与概念之间,以及概念所具有的属性之间的关系为基本内容的常识的知识库. HowNet 是一个比较详尽的语义知识词典,受到人们的普遍的重视,不过 HowNet 对一个词的语义采用的是一种多维的表示,这给词之间的相似度计算带来一定的困难. 这与 WordNet 与同义词词林不同. 在 WordNet 与同义词词林中,所有相同的语义项构成一个树状的结构,要计算语义项之间的距离,只要计算树状中结构相应节点的距离即可. 而在知网中词的语义是由多个义原构成的,计算词的语义相似性是相当麻烦的.^[3]

2.2 原理的介绍

(1) 语义相似性

每个词的词义都是有多个义原组成的,考虑词与词的语义相似性是综合考虑每个词的义原集,但是每个词的义原集第一义原占的比重是非常大的.

(2) 词汇的相关度

对于两个词语 w_1, w_2 , 如果 w_1 有 n 个义原: $s_{11}, s_{12}, \dots, s_{1n}$, w_2 有 m 个义原: $s_{21}, s_{22}, \dots, s_{2m}$, 我们规定 w_1 与 w_2 相似度是取各个义原相似度的最大值:

$$\text{sim}(w_1, w_2) = \max_{i=1, \dots, n, j=1, \dots, m} \text{sim}(s_{1i}, s_{2j}) \quad (1)$$

该式仅仅从义原方面考虑词汇的相似性,摒弃了第一义原的重要位置^[4]. 本文主要就是从义原相似度方面进行考虑的.

(3) 义原相似度计算

所有的义原是根据上下位的关系,构成一个树状的层次体系,假设两义原在这个层次体系的距离为 d ,那么两个义原的语义距离:

$$\text{Sim}(p_1, p_2) = \frac{\alpha}{d + \alpha} \quad (2)$$

式中, p_1 与 p_2 分别代表义原, α 是一个可调节参数.

(4) 改进除去孤立点的方法

孤立点对聚类结果的精确度影响是非常大的,传统的排除孤立的方法是计算每个点到所有点的距离的和,然后对这些距离进行排序,选出距离较大 $u\%$ 的值,然后排除掉,但是对于这个 u 是不好掌控的. 在本文中,通过 Z 分数也叫标准分数的方法来排除孤立点. 设 $P[i][k]$ 表示第 i 个点在第 k 维的数值, i 与 j 的欧式距离,可以通过下面的公式进行计算:

$$D[i][j] = \sqrt{\sum_{k=1}^n (p_o[i][k] - p_o[j][k])^2} \quad (3)$$

这里的 n 表示这个点有 n 个维度, i 点到所有点

$$\text{的距离的和: } \text{Dis}[i] = \sum_{m=1}^N D[i][m] \quad (4)$$

i 点的 Z 分数是这样定义的:

$$z[i] = \frac{1}{\sigma} (\text{Dis}[i] - \mu) \quad (5)$$

这里的 $\mu = \frac{1}{N} \sum_{k=1}^N \text{Dis}[i]$, 就是所有点到其他点

距离和的平均值, $\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (\text{Dis}[i] - \mu)^2}$ 为标准差^[5]. 在这里,将求得所有点到其他点距离,然后

根据这些距离求得它们的 Z 分数. 然后再对这些点进行聚类,聚类的簇数选择 2. 聚类得到结果为 C_1, C_2 两个集合,假定 C_1 的包含点的数目 $> C_2$ 包含点的数目,认为 C_2 中对应的点出现孤立点的概率较大. 然后结合上面求得的 Z 分数,根据 Z 分数进行判断,判断规则如下:

① 如果点属于 C_2 并且它的 Z 分数 > 0 , 就认为它是孤立点,将其对应的点删除.

② 其他的情况都认为它不是孤立点.

(5) 构建替代词库

每个词的词义都是由义原集组成的,对于任意的两个词,如果仅仅从语义方面考虑,有可能它们的语义相似度非常低,但是它们出现在同一篇文章的概率或者反映同一个主题的概率也许非常大,所以在这篇文章中要从词的义原方面来考虑,换句话说就是从词的相关性方面来考虑. 对于给定的一组词,首先获得它们当中每一个词对应的义原集,然后从第一个词依次循环遍历这些词与义原集. 假设一组词与义原集分别为: $w_i(y_1, y_2, y_3), w_j(y_1), w_k(y_6, y_3), w_m(y_{10}, y_3), w_n(y_3)$, 如果发现一个词与其他词不相同,但是它们的义原集中含有相同的部分,我们就把二者共同含有的义原分别提出出来,并将二词分别放在它们共同含有的义原后面. 得到 $y_1[w_i, w_j] y_3[w_i, w_k, w_m, w_n]$, 但是如果发现一个词在不同行中重复出现,就将其从重复的行中删除,得到初步结果为: $y_1[w_j] y_3[w_k, w_m, w_n]$. 根据领域词典进一步判断每一项的义原是否跟一些主题(服务业,医疗……)相关,如果不相关将其对应行删除掉,这样将得到初步的替代词库. 然后再从语义方面考虑 y_1, y_3 的语义相似性,如果发现 y_1 与 y_3 语义非常相近,我们将它们进一步合并得到 $y_1[w_j, w_k, w_m,$

w_m], 如果不相近就不合并, 这样就得到一个替代词库. 具体算法描述如下:

输入: m 个词语以及它们的义原集

输出: n 行相关词的集合

算法:

For $i = 1:m$

① 取得对应行的词语和它的义原集.

For $j = 2:m$

② 取得对应的词语与其义原集.

③ 比较它们的义原集是否含有相同的部分. 如果有, 先判断是否有已经建立好的相关词, 如果没有, 就建立一个相关词, 这相关词第一项是它们共同含有义原构成的, 然后在其后面, 加上对应的两个词语. 如果已经含有相关词见步骤④.

④ 先判断已有的相关词中第一项与步骤③中二个词语共同含有的义原是否相同, 如果相同就直接将两个词语添加在已有的相关词后面, 如果没有就按照步骤③重新建立一行相关词.

End

利用此算法就得到了一个替代词库.

(6) 初始聚类中心的选择

传统的 k -means 聚类算法, 初始聚类中心选取是随机地选取 K 个样本点作为初始聚类中心, 每个样本点分别代表一个簇. K -means 算法对选取的初始聚类中心很是敏感, 初始聚类中心的选取不当会使得整个算法陷入局部最优解^[6,10]. 本文提出了基于 Z 分数的方法, 选择初始聚类中心. 算法描述如下:

输入: N 个数据对象和聚类簇个数 K

输出: K 个初始聚类中心

算法:

For $i = 1:k$

① 首先求得每个数据到其他数据的距离和, 得到一个距离和矩阵 $D[i=1 \cdots N]$.

② 对这个距离和矩阵按照从小到大的顺序进行排序, 得到一个相对于排序前原数据的索引编号.

③ 取得索引编号的第一个编号, 并取得这个编号对应的原数据, 然后求得这个数据到其他数据距离, 得到一个距离矩阵 $J[1 \cdots N]$.

④ 根据 J 这个距离矩阵求得其对应的 Z 分数, 从 Z 分数中取得 Z 分数小于 0 行号, 根据这些行号取得这些行号对应的原数据, 也就是与步骤③取得的编号相近的数据, 计算出这些数据的平均值即得到一个初始聚类中心. 删除取得的行号的对应数据, 按照这种方法继续循环下去就可以得到所有的初始

聚类中心.

End

3 改进聚类算法描述

本文针对传统聚类算法的缺点, 做了以下处理:

输入: N 篇文档, 聚类簇数 K .

输出: K 个集合

算法:

① 根据输入的 N 个文档对象, 利用正则表达式判断一下每篇文档是否含有标题, 如果含有, 就提取这些标题, 然后进一步判断标题中是否出现标点符号. 例如, 如果出现引号, 就将引号与它包含的内容删掉. 同样, 如果出现破折号, 就将破折号与破折号前面的内容删掉, 这是从中文标点符号意义上考虑的, 如果不除掉这些词有可能影响对文档特征词的正确选取. 如果没有标题, 就提取文章的内容. 通过这种方法能够实现降维的目的, 然后使用中科大的分词工具 (ICT-CLAS) 进行分词并且标注所有的词的词性, 然后依据哈工大的停用词表除掉停用词.

② 使用 TF-IDF (term frequency-inverse document frequency) 的统计方法, 计算每篇文档中词的 tf-idf 的值, 依据这些值选出每篇文档的特征词.

③ 根据每篇文档的特征词, 构建这些要进行聚类文档的特征空间.

④ 根据 HowNet 得到这个特征空间中每个词的义原集, 然后使用 2.2 节中提到的构建替代词库的方法构造一个替代词库.

⑤ 利用④中构造的替代词库, 判断步骤①分词得到的词是否出现在替代词库的中, 如果出现了就用替代词库中的对应的相关词的第一项进行替换.

⑥ 利用替换后的词重新构造一个特征空间, 利用这个特征空间将每篇文档数值化.

⑦ 利用 2.2 节中提到的改进的除去孤立点的方法, 将孤立点消除掉.

⑧ 利用 2.2 节中提到的选择初始聚类中心方法, 选出初始聚类中心, 然后进行聚类.

4 实验分析

为检测改进算法的有效性, 本实验使用常规的文本聚类分类的语料库 (包含政治、艺术、医药、体育、军事、经济、教育七个类别), 它里面的文本长度不一. 文本的预处理采用 java 语言编写的, 除孤立点, 初始聚类中心的选择和文本聚类使用的 Matlab 语言. 在 Pentium(R)4 3.0 GHz, 2 GB 内存 MyEclipse6.5 Matlab R2010(a) 下运行的. 本实验在语料库中随机选择一些文本进行了 7 次实验, 实验效果比较见图 1 与图 2.

图 1 中的虚线部分表示改进前的算法, 由于传

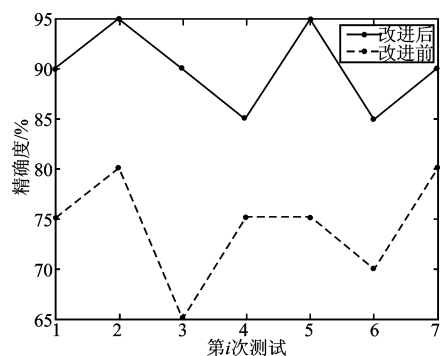


图1 精确度对比图

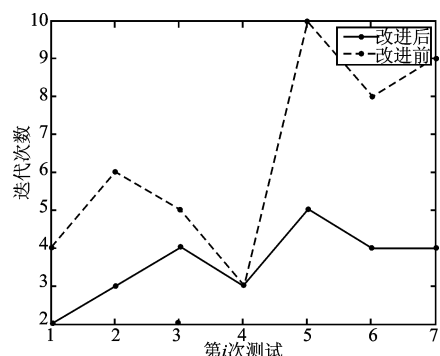


图2 迭代次数对比图

统的算法对初始聚类中心的选择和孤立点的消除采用的方法不当,并且没有考虑词与词的相关性导致实验结果的波动非常大,聚类的精确度也非常低。而实线部分是改进后的算法,但是由于受到分词工具的影响,还有就是 HowNet 出现了一些未登录词导致其精确度也受到一定影响,但是相比之前精确度有了明显的提高。

图2中的虚线部分表示改进前的算法,实线表示改进后的算法。改进前聚类算法是随机地选择初始聚类中心导致它的聚类迭代次数是非常多的。而改进后的算法,利用数据分布的稀疏程度然后采用 Z 分数选择初始聚类中心,所以相比改进前聚类的迭代次数大大减少了,迭代次数的减少也就意味着

聚类所需要的时间大大的减少了。

5 结束语

本文采用了 Z 分数消除孤立点,根据词汇相关性,构建一个替代词库,它类似于领域词典,提高了聚类的精确度,并且采用根据数据的分布的稀疏程度选择初始聚类中心,使得聚类的迭代次数减少,缩短了聚类的时间。但是由于分词工具准确率不是那么完美,还有 HowNet 出现的一些未登录词导致改进算法受到一定的影响,这在今后将进一步的研究。

参考文献:

- [1] 苏锦旗,薛惠峰,詹海亮. 基于划分的 K-均值初始聚类中心的优化算法[J]. 微电子学与计算机, 2009, 26(1): 8-11.
- [2] 周昭涛. 文本分析聚类分析效果评价及文本表示法[D]. 北京:中国科学院技术研究所, 2005.
- [3] Alexander Budanitsky, Graeme Hirst. Evaluating wordNet-based measures of lexical semantic relatedness[J]. Computational Linguistics, 2006 2(1):13-47.
- [4] 刘群,李素建. 基于知网的词汇语义相似度的计算[C] // 第三届汉语词汇语义学研讨会. 台北:[s. n.], 2002: 59-76.
- [5] 贾俊平. 统计学[M]. 北京:中国人民大学出版社, 2006.
- [6] 钟勇,赵向辉. 一种优化初始中心点的 k-means 文本聚类算法[J]. 郑州大学学报:理学版, 2009, 41(2): 30-30.

作者简介:

周爱武 女, (1965-), 副教授, 硕士生导师. 研究方向为数据库与 WEB 技术. E-mail: 943619770@qq. com.

汪贤惠 男, (1989-), 硕士生. 研究方向为数据挖掘. E-mail: 1027422682@qq. com.

刘惠婷 女, (1978-), 副教授, 硕士生导师. 研究方向为数据挖掘.

(上接第 89 页)

IEEE, 2001: 228-231.

- [5] Nan Yan, Hao Min. A high efficiency All-PMOS charge pump for low-voltage operation[C] // Proc IEEE Asian Solid-State Circuits Conf. Hsinchu, Taiwan: IEEE, 2005: 361-364.
- [6] Hongchin Lin, Nai-Hsien Chen. An efficient clock scheme for low-voltage four-phase charge pumps[C] // proceedings of the International Symposium on VLSI Technology System, and Applications. Hsinchu,

作者简介:

李卿 男, (1989-), 硕士研究生. 研究方向为模拟集成电路设计. E-mail: liqing. ic@foxmail. com.

周健军 男, (1970-), 教授, 博士生导师. 研究方向为模拟射频集成电路设计、混合集成电路设计.