西北大学

硕士学位论文

# 基于k-means的中文文本聚类算法的研究与实现

姓名：张睿

申请学位级别：硕士

专业：计算机软件与理论

指导教师：刘晓霞

20090601

# 摘要

在机器学习、数据挖掘等领域得到普遍应用的k-means算法由于具有时间复杂度低的优点，在文本聚类领域也得到了广泛的应用。论文对文本聚类的相关技术与算法进行研究，针对文本数据高维性和稀疏性的缺点，改进了文本聚类中的特征选择方法，以及与k-means相关的算法，并在此基础上设计并实现了一个中文文本聚类原型系统。主要工作有：

1) 聚类领域进行特征选择时由于缺乏类信息而难以选择出最具类区分能力的特征词。在文档频率，单词贡献度两种特征选择方法的基础上，利用贪心算法对特征进行增量选择。实验表明改进的算法可以在保证聚类质量的前提下过滤更多的特征词。

2) 文本数据高维性和稀疏性的特点使得文本对象间的相似度不易度量，根据文本间的相似度为k-means算法选择的始聚类中心时可能不能很好的代表整个文本集。针对该缺点，对k-means算法中的初始化问题，提出一个改进的初始聚类中心选择方法。实验表明改进的方法选择到初始聚类中心比较分散且代表性好。

3) 为了提高聚类中簇的质量，通过引入共享最近邻相似度中邻居的概念，对bisecting k-means算法进行改进，实验结果表明该算法的聚类质量较原算法有一定的提高。

在以上研究工作的基础上，实现了基于k-means的中文文本聚类原型系统。通过实验对系统中的各个算法进行了评测和比较。

**关键词：** 文本聚类，k-means，bisecting k-means，共享最近邻

# Abstract

As a widely used algorithm in machine learning and data-mining, k-means is also used in document clustering for its low time complexity .This paper mainly focus on the how to improve the performance of document clustering algorithm. Based on existing research, improved k-means algorithms and new feature selection method are proposed. Design and implement a Chinese document clustering System on the basis of the proposed algorithms. Works achieved in this paper are as follow:

1) It is hard to select features for unsupervised feature selection methods used in clustering due to the lack of class label information. Based on document frequency and term contribution, greedy algorithm is introduced to select features incrementally .Experiments show that the proposed method can remove more features than traditional methods without degrading the clustering quality.

2) In order to improve the clustering quality of k-means, well separated initial centroids should be selected. Initial centroids are aurally hard to select due to the high dimensionality and sparseness of document data. A new method for selecting initial centroids is proposed. Experiment show that the centroids selected by the proposed method are well separated and with high representative.

3) In order to improve clusters quality of the bisecting k-means, neighbor used in shared nearest neighbor is introduced. Experiments show that the improved algorithm performs better than the original one.

Design and implement a document clustering system using the algorithm mentioned above. Each algorithm in the system is contrasted and evaluated through experiments.



**Key words:** Document Clustering, k-means, bisecting k-means, Shared Nearest Neighbor

## 西北大学学位论文知识产权声明书

本人完全了解西北大学关于收集、保存、使用学位论文的规定。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文被查阅和借阅。本人授权西北大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。同时授权中国科学技术信息研究所等机构将本学位论文收录到《中国学位论文全文数据库》或其它相关数据库。

保密论文待解密后适用本声明。

学位论文作者签名：张睿　　指导教师签名：刘晓鹏

2009年 6月 22日　　2009 年 6月22日

---

## 西北大学学位论文独创性声明

本人声明：所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，本论文不包含其他人已经发表或撰写过的研究成果，也不包含为获得西北大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：张睿

2009年 6 月 22 日

# 第一章 绪论

## 1.1 研究背景与意义

当今是一个信息主宰的时代，文本作为重要的信息载体之一，其数量正以惊人速度增长。如何从这些庞杂的信息中快速有效地寻找到满足需求的信息，对人们来说是一个很大的挑战，依靠手工对这些信息进行处理是不可能的。人们迫切需要计算机自动地对这些大规模的文本集合进行有效的处理和分析。文本聚类作为处理和组织大量文本数据的关键技术，能够在很大程度上解决信息爆炸和信息杂乱所带来的问题。文本聚类依据著名的聚类假设：同类文档的相似度较大，不同类文档的相似度较小。文本聚类的目标是将文本集合分成多个簇，使得在同一个簇中的文本内容具有较高的相似度，而不同簇中的文本内容差别较大。

文本聚类与文本分类不同，后者由于有训练集的存在，可以在分析文本内容的基础上根据事先确定的分类目录给文本分配一个或多个比较合适的类别。文本聚类技术可以对未知类别标识的文本集合进行分析，根据文本集自身的结构发现其中的类别信息，并对其中的文本进行类别标识。分析和标识文本的类别有助于发现文本集内部的信息，进而可以作为多文档自动摘要、文本消歧等自然语言处理的预处理步骤[1]。文本聚类技术还可以用来改善分类的结果，提高检索的性能，在检索系统中应用聚类技术可以帮助用户快速准确地查找到其所需信息，提高系统分类和检索的精度[2]。文本聚类技术还可以通过对用户感兴趣的文档进行聚类，从中发现用户的兴趣模式并用于信息过滤和信息主动推荐等服务[3]。用于文档集合的自动整理与数字图书馆服务[4][5]。用于热点主题发现和流行串预警[6]，快速发现网络中的热点话题并进行追踪，识别网络中大规模流行的病毒或蠕虫特征，对维护社会和谐以及保护国家安全具有重要的意义。

中文在构词成句上比英文复杂，且理论研究上还不成熟，随着时代的发展和信息的全球化，互联网上中文信息急剧增加，它的作用越来越重要，然而人们从中获取需要的信息的难度却有所增加。因此，研究中文文本聚类技术，提高中文文本聚类的效率和准确率已经成为促进我国经济发展和国际知识交流的迫切要求，具有重要的现实意义。

## 1.2 研究现状

国外对英文文本聚类已经进行了大量的研究，并已将文本聚类成功地应用于文本挖

掘和信息检索等领域。上个世纪 90 年代以来，文本聚类研究更多地集中在对大规模文档集合的浏览[5],[7]、对搜索引擎返回的查询结果重新组织中[8]。近年来，文本聚类算法在话题检测与跟踪(TDT，Top Detection and Tracking)领域中得到了进一步研究与应用。与国外相比，国内对中文文本聚类的研究和应用起步较晚，主要的研究机构集中在高等院校、科研院所和信息公司[9]，如：东北大学、复旦大学、哈尔滨工业大学、中科院计算所及北京拓尔思公司 TRS 等。

在聚类算法方面，基于机器学习的文本聚类方法得到了广泛的研究，包括基于划分的方法[10],[11]、基于层次的方法[12]、基于密度的方法[13]、基于模型的方法[14]、模糊聚类方法[15]等等。

steinbach 等人比较了基于层次的方法和基于划分的方法在文本聚类中的适用程度[16][17]，认为 k-means 和 bisecting k-means 算法不但聚类结果较好，且处理时间和文本数量呈线性关系，适用于大规模文本的聚类。文本数据不但维数很高且具有方向性数据的某些特征，在文本聚类中使用余弦夹角作为相似性度量得到的聚类结果要明显好于使用欧氏距离[18]。BanejeeA.等人提出了针对文本数据和基因表达数据的混合 vMF 密度模型聚类算法[19]。虽然这些聚类方法能够完成有效聚类，但是还存在着对初始化敏感的问题，不能够自动判断簇的数目。

针对文本数据维数很高的问题，后来提出了新的聚类方法，如关联聚类算法，它利用频繁项集来表示文本，从而大幅度降低了文本的维度。HFTC 每次选择下一个较频繁的项集来构造下一个簇[20]。FIHC 认为同一个簇内的文档之所以相似是因为它们共享了较多的频繁项集，利用频繁项集之间的包含关系来生成文本的层次[21]。这些算法克服了高维数据处理的复杂性问题，但有研究者认为这些算法的结果比 bisecting k-means 和 UPGMA 差[22]。

R-means 算法采用 k-means 的聚类过程，融入关联聚类的观点，利用频繁项集构成的规则集作为簇中心[23]。MFI k-means[24]算法利用最大频繁词集得出 k-means 初始条件，从而解决了 k-means 的初始化问题，但该算法容易受文档长度的影响。

bisecting k-means[16]应用于文本领域时可以得到比较好的聚类结果，并且得到嵌套的文档类别结构，以树的形式表示，复杂度也比较低。该方法得到的结果中各个簇大小相差不大。由于文本数据具有高维性和稀疏性的特点，传统的选择下一次分裂的簇的方法在应用于文本领域时出现了一些问题。

针对文本数据高维性和稀疏性的特点，本文研究如何解决 k-means 的初始化问题和

bisecting k-means 如何选择下一次分裂的簇的问题。

## 1.3 文本聚类研究的难点

文本对象作为非结构化的数据，与传统的结构化数据不同，同时具有高维性和稀疏性的特点，对文本对象进行聚类分析时，存在着其特有的挑战。

1) 文本对象的高维性

文本集经过预处理后，经常包括成千上万甚至几十万个特征词。随着数据对象维度的增高，一般的数据挖掘方法由于计算量过大或代价高昂而不具有可行性，使得一些在低维数据上运行得很好的聚类算法，在面对高维数据的时候，无论在性能上还是效果上都有很大的下降。因而有必要对现有方法加以改进以适应高计算量、高资源消耗的文本处理问题；同时还可以研究文本表示的新方法或者有效的维数约简方法。

2) 文本对象的稀疏性

一个文本集经过预处理后，通常都包括成千上万个特征词。当采用向量模型表示文本时，每个文本都是由整个文本集所包含的特征进行描述，由于文本集中通常包含多个主题，每个主题均由不同的特征词描述，对于一个文本来说，它所包含的特征词只是整个文本集的特征词中很少一部分，因此文本向量的非零项通常非常少，从而导致了文本表示形式的高度稀疏性。而稀疏性又导致了对象之间的关系仅仅取决于很小一部分的属性，进而导致文本对象间的相似度非常低，相似度趋向于更加一致。这些对基于相似度的聚类算法产生了很大的负面影响，直接影响了聚类算法的质量与性能。

3) 语义问题

文本数据还具有同义词和近义词等特有的自然语言现象。同义词和近义词的现象是指可以用多种不同的方式来描述同一个主题或者内容。它们的存在极大地降低了文本聚类的精确率和效率。通过使用潜在语义索引的方法对文本对象进行处理，可以达到特征词降维和增强语义关联的目的。虽然该方法会耗费大量时间，但有研究者认为在通过潜在语义索引压缩后的特征空间聚类所节省的时间可以弥补压缩特征空间时所花费的时间。

4) 聚类描述问题

聚类描述是帮助用户迅速确认生成的文档类相关与否的重要信息。生成可以理解的聚类描述可以帮助用户理解得出的结果，引导用户浏览聚簇，因此聚类描述对于非专业用户也应该是可以理解的。

## 1.4 本文的研究内容

对基于 k-means 的中文文本聚类相关算法进行了深入的探讨，对影响文本聚类性能的主要技术，包括停用词处理、特征选择方法以及聚类算法等，进行了理论分析和实验研究。具体工作如下。

1) 对几种典型的特征选择方法进行了研究。针对在聚类领域进行特征选择时由于没有类信息可以使用，阈值难确定，在文档频率，单词贡献度两种特征选择方法的基础上，通过引入贪心算法的思想，提出了基于文档频率和单词贡献度的改进的特征选择方法。实验表明改进的算法可以在保证聚类质量的前提下过滤更多的特征词。

2) 文本数据高维性和稀疏性的特点使得文本对象间的相似度不易度量，因此，根据文本间的相似度为 k-means 算法选择的初始聚类中心可能不能很好的代表文本集中的各个类别。针对该问题，结合共享最近邻与最大最小原则对聚类中心分别进行选择，提出一个改进的初始聚类中心选择方法。实验表明改进的方法选择到初始聚类中心比较分散且代表性好。

3) 针对文本对象间的相似度不易度量的特点，引入共享最近邻中邻居的概念对簇的紧密性进行刻画，每次选择紧密性最差的簇进行分裂。利用以上选择分裂簇的方法对 bisecting k-means 算法进行改进。实验结果表明改进的算法的聚类质量较原算法有一定的提高。

4) 在以上研究工作的基础上，实现了原型系统，能够对文本进行预处理并进行聚类分析。系统包括文本解析模块和聚类分析模块，文本解析模块实现了文档频率，单词贡献度以及基于文档频率和单词贡献度的改进的特征选择方法。聚类模块实现了基于共享最近邻的改进的 k-means 算法及基于共享最近邻的改进的 bisecting k-means 算法。实验表明，改进算法的使用提高了文本聚类的质量。

# 第二章　中文文本聚类的主要技术

对文本聚类过程及相关技术进行研究，包括文本预处理、降维方法、文本表示和聚类算法等几个方面。

## 2.1 文本聚类过程

文本聚类是指将文本集聚合为由若干个文本簇组成的集合的过程。文本聚类大体包括两个步骤，首先，文本是非结构化的数据，无法直接使用数据挖掘中的算法对文本对象进行处理，必须对文本进行预处理，并将文本表示为计算机可以处理的形式；其次，再对文本对象进行聚类分析。文本的预处理技术对于文本挖掘来说是一个非常重要的环节，将文本表示成计算机所能够处理的形式时，需要保证这种形式能够充分体现出文本对象自己的特点，突出文本对象间的差异，以便于区分文本。预处理的质量直接影响最终的聚类结果。具体流程如图 2.1 所示。

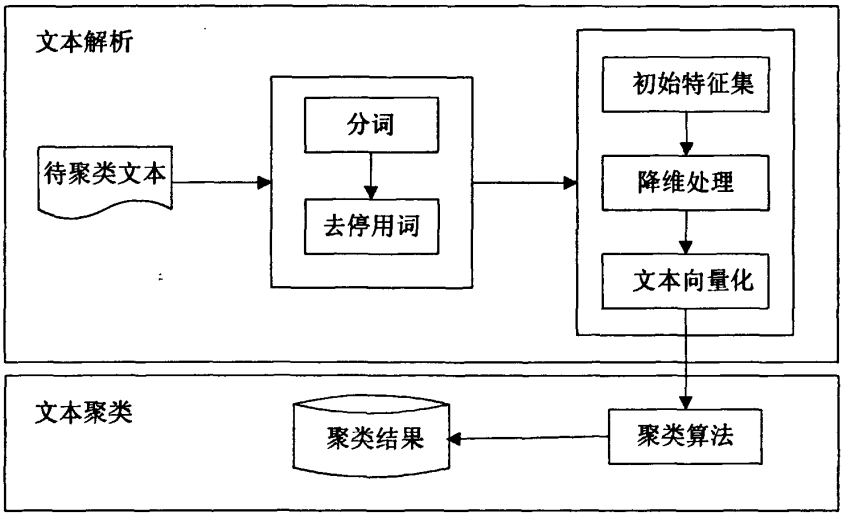


**图 2.1 文本聚类的过程**

## 2.2 文本预处理

文本由大量字符所构成的字符串组成，它属于一种非结构化的数据，为了将其变为计算机可以处理的形式，必须进行预处理。首先对文本进行分词处理，将连续的字序列按一定规则分隔为分散的有独立意义的词序列，然后过滤其中的停用词(Stop Word)，得到文本的关键词集合，这些关键词构成了初始特征项(词)集合。

### 2.2.1 中文分词

预处理的第一个关键步骤就是分词。在文本信息处理过程中，一般可以选择字、词或词组作为文本的特征项，由于字所代表的信息量太少，以及多义字的存在，用字作为特征项会导致特征向量庞大，造成特征空间的维灾难。词组携带有足够的信息量，但词组在文本中出现的机率不多，将其作为特征项会导致特征项量稀少，无法很好的代表文本，因此，选取词作为特征项要优于字和词组[25]。为了得到文本的特征词集，需要对中文文本进行分词，称为中文分词(Chinese Word Segmentation)。分词就是将连续的语句按照一定的规范分隔为分散的具有独立意义的词序列的过程。另外，与英文不同，汉语的形态不丰富，词语基本上没有形态变化，每个词之间也没有空格间隔。在这种情况下，书面汉语词法分析的主要任务不是分析单词的形态变化，而是进行单词的自动切分，使前后相继的词与词之间的界限暴露出来。词是语言中最小的能独立运用的单位，利用计算机把汉语的一个句子、一篇文章、一部著作中的单词，逐一地切分出来，才有可能对汉语进一步分析。

英文单词之间使用空格分隔开，从语义准确性及技术复杂度来讲都比较简单；中文文本是由连续文字组成，只有句、段分隔，但在文本信息处理过程中需要以词作为最小单位，因此，相对西方文本语言来讲，更加的复杂和困难。本文采用中国科学院计算机技术研究所的 ICTCLAS 系统进行分词，该系统对分好的词进行了词性标注。

### 2.2.2 停用词处理

停用词是指那些对文本类别标识没有太大作用的单词，大致可以包括两类：第一类是弱词性词，包括助词、连词、介词等表征能力比较弱的词性，这些词本身并无实际意义，和类别信息没有关联；第二类是均匀分布在各类型文本中的词汇，由于它们在所有类的文本中都会出现，这些词区分类别的能力普遍很弱。将这些单词过率，可以降低特征空间的维数和减少噪音。

英文停用词的研究比较成熟，已经有了一些公开发表的英文停用词表，其中比较著名的是 van Rifsbergen 发表的停用词表以及 Brown corpus 停用词表。由于中文词汇量比英文词汇量要大的多，再加上歧义词的影响，使得中文停用词表的研究成果还非常少，目前还没有一个被广泛认可的中文停用词表，往往需要自己手工构造适合特定词语集的停用词表。

停用词表的来源包括人工构造和基于统计的自动学习两种方式。人工构造是根据主

观判断选择特征词集或针对某一领域选择领域常用词来构成停用词表。基于统计的自动学习方法使用词频信息构建停用词表，或者通过对分词后的结果进行统计分析得到停用词，然后在分词过程中不断更新频率并根据切分结果进行验证。此外还可以依据词语和句子的联合熵构建停用词表[25]。

## 2.3 文本表示

文本表示也称为文本特征的表达。文本是由大量字符构成的字符串，属于非结构化的数据，无法直接用于聚类分析。需要对其进行转化，即用简单而准确的方法将文本表示成计算机所能够处理的形式。目前提出了多种文本表示方法，包括布尔模型，概率模型和向量空间模型等文本表示方法。这些模型从不同的角度出发，使用不同的方法处理特征权重和相似计算等问题。向量空间模型(vector space model，简称 VSM)将文本表示为特征项和特征项权重组成的向量。布尔模型也将文本表示称为特征空间上的一个向量，只是向量中每个分量的权重为 0 或者 1。由于权重的二值性，布尔模型无法计算两个文档之间更深层面的相似度。概率模型综合考虑了词频、文档频率和文档长度等因素，把文档和用户兴趣(查询)按照一定的概率关系融合，在概率测度空间上通过概率来衡量两个文本的语义相似度，在信息检索领域得到了较为成功的应用。这几种模型中，向量空间模型是最简便有效的文本表示模型之一。以下详细介绍本文聚类所用的向量空间模型。

1) 向量空间模型

向量空间模型[27]由 Gerard Salton 等人于 60 年代末提出，成功的应用于著名的 SMART 文本检索系统，后来又在文本聚类领域得到了广泛的应用。该模型的主要思想是，将每一文本都映射为由一组规范化正交特征向量所组成的向量空间中的一个点。向量的各维对应文本中的一个特征项，特征项是指构成文本的各种单位，如词、词组等，特征项的权重反映它对于所在文本内容的重要程度。向量空间模型假设:一个文本所属的类别仅与组成文本的词或词组在文本中出现的频率有关，而与出现的位置或顺序无关。对于中文文本而言，可以选择字、词、词组、短语、句子作为文本特征项，由于词汇是文本的最基本表示项，在文本中出现的频率较高而且呈现一定的统计规律，因此在向量空间模型中一般选择词作为特征项。对于一个文本 $d_i$，$t_j(j=1,\cdots,n)$ 为一些互不相同的词条，特征项 $t_j$ 在文本 $d_i$ 中的权重记为 $w_{ij}$，文本 $d_i$ 表示为：

$$V(d_i)=((t_1,w_{i1}),(t_2,w_{i2}),\cdots,(t_n,w_{in})) \tag{2.1}$$

2) 特征项的权重计算

特征项的权重综合反映了该特征对标识文本内容的贡献程度和区分不同文本的能力。常用的表示法有两种，一种是以布尔值 1 或 0 分别表示某个特征词在文档中出现或不出现;另一种是用[0，1]区间内的实数来表示某个特征词在文档中的权重。常用的权重表示方法有以下几种：

(A)开根号权重。对词频开根号计算，即

$$w_{ij}=\sqrt{tf_{ij}} \tag{2.2}$$

$tf_{ij}$ 指特征词 $t_j$ 在文本 $d_i$ 中出现的次数，一个词在文档中出现的次数对该文档和所属类别的识别都有着很重要的作用。然而直接使用词频计算出的权值会很高，如果个别项的权值很高，在聚类过程中往往会抑制其他项的作用，因此在计算各特征项权重时，需对统计出的词频做适当的均衡处理，可以通过对词频开根号并将其作为权重。

(B) $tf\times idf$ 权重。$tf\times idf$ 权重方法用词频和文档频率共同表示特征词对文本的贡献，如(式 2.4)所示。文档频率(Document Frequency，简称 DF)，指文本集中含有该特征的文档数目。反文档频率(inverse Document Frequency，简称 IDF)反映特征词在整个文档集合中的分布情况，一定程度上体现了该特征的区分能力。如(式 2.3)所示。

$$idf_j=\log(N/df_j) \tag{2.3}$$

$$w_{ij}=tf_{ij}\times idf_j=tf_{ij}\times\log(N/df_j) \tag{2.4}$$

$tf_{ij}$ 指特征词 $t_j$ 的词频，$df_j$ 指特征词 $t_j$ 的文档频率，$N$ 指文本集中文本的数目。

在一个文档中出现频率高的词，说明它在区分该文档内容方面的能力强，应该赋予较大的权重，用 $tf_{ij}$ 保证该词权重较大；但是如果这个词在各类型文本中都出现，就很难说这样的特征词到底代表哪篇文档，所以应该降低它的权值，用反文档频率 $idf_j$ 来保证这类词语的权值较小，一个词条在所有文本集中出现的范围越广，$df_j$ 值越高，$N/df_j$ 值越低，$idf_j$ 值越低。

(C) ltc 权重

$$w_{ij}=\log\ (tf_{ij}+1.0)*(\log N/df_j) \tag{2.5}$$

ltc 权重结合了(式 2.3)和(式 2.4)，在 $tf \times idf$ 权重的基础之上进一步降低词频 tf 的作用，也就是，理论上是一种更科学的计算词权重的公式。是目前应用较多的权重计算方式。

## 2.4 降维处理

经过预处理得到关键词的集合构成了初始特征项(词)集合，简称特征集。直接使用预处理模块生成的特征集对文本进行描述，可能造成文本向量的维数过大而导致计算复杂。即使是一个小规模的文本集，经过预处理也会得到几万甚至几十万的特征词，其中有些词在文本中出现次数极少，不能代表该文本，称为低频弱关联词，有些词则出现频率较高，蕴含了大量和类别有关的信息，称为高频强关联词。特征处理，就是通过降维技术将弱关联词去掉，抽取强关联词构成用于学习的特征集。降维技术包括特征选择(Feature Selection)和特征抽取 (Feature Abstraction)。在文本聚类研究领域，较多的工作集中在特征选择的研究方面。

1) 特征选择

特征选择一般是通常根据某个特征评估函数计算各个特征的评分值，然后按评分值对这些特征进行排序，选取若干个评分值最高的作为特征词，或者设置一个阈值，若某特征的评分值大于阈值，将该特征项作为向量空间中模型的一项，如果小于阈值，则忽略该特征项，从而降低向量空间维数。目前，存在多种评估函数，有文档频数(DoeumentFrequeney，DF)、单词权(Term Strength，TS)、单词熵(Entropy-based Feature Ranking，EN) 、单词贡献度(Term Contribution，TC) 、信息增益 (Information Gain，IG)等。特征选择在文本分类上已经得到了非常成功的应用，但是在文本聚类中的研究不如在文本分类领域成熟。文本分类领域中优秀的方法，如 IG 和 CHI 由于需要类信息不能应用于文本聚类中。

2) 特征抽取。

特征抽取(FeatureExtraction，FE)，指用映射或变换的方法把原始高维特征上的数据变换到低维特征空间，分为线性和非线性的。与前面所述的方法不同，特征抽取直接使用线性或者非线性的变换将冗余或者无效的信息映射到相对较弱的维度上，在保持尽可能低的冗余信息的同时揭示隐藏在原始特征空间中的潜在特征，保证特征提取的效果。具体包括奇异值分解(SVD)、潜在语义索引(LSA)、因子分析(FA)、基于自组织映射的特

征抽取。特征抽取的最大特点在于能揭示隐藏在原始特征空间中的潜在特征，重构的新特征能最大限度地减少对象在重构特征空间表达中的信息丢失。特征抽取可以有效地降低同义词和近义词对文本分类和聚类的影响，同时能有效地降低特征空间的维数。

## 2.5 文本相似度的计算

相似度是用来衡量两个对象之间相似程度的度量。如果两文本之间相似度为 1，则说明两文本对象完全相同；如果相似度为 0，则说明两文本没有相似之处。我们也可以使用距离的概念来衡量两个对象之间的相异度。相似度和相异度是两个相对的概念，在聚类分析中这两个概念经常一起使用。常用的度量公式有:

1) 欧几里德距离，简称欧氏距离

$$dis(d_i,d_j)=\sqrt{\sum_{k=1}^{n}(w_{ik}-w_{jk})^2} \tag{2.6}$$

$w_{ik}$ 指文本 $d_i$ 的特征 $t_k$ 的权重，n 是特征总数。

2) 余弦相似度

文本的相似度可以用向量之间的夹角余弦来表示，余弦计算的好处是得到的值恰好是介于 0 到 1 的数。

$$sim(d_i,d_j)=\frac{\sum_{k=1}^{n}(w_{ik}\cdot w_{jk})}{\sqrt{\sum_{k=1}^{n}w_{ik}^2\cdot w_{jk}^2}} \tag{2.7}$$

3) 广义 Jaccard 系数

二元 Jacacdr 系数通过计算两个文本对象的布尔向量中共现词数目和非共现词数目的比值得到，1 表示该特征词出现，0 表示该特征词不出现，$f_{11}$ 表示两文本同时取 1 的特征的个数，$f_{00}$ 表示 2 文本同时取 0 的特征的个数，

$$sim^{(J)}(d_i,d_j)=\frac{f_{11}}{m-f_{00}} \quad \text{m 为特征总数} \tag{2.8}$$

广义 Jaccard 系数又称 Tanimoto 系数，可用于普通向量。

$$EJ(d_i,d_j)=\frac{\sum_{k=1}^{n}(w_{ik}\cdot w_{jk})}{\sum_{k=1}^{n}w_{ik}^2+\sum_{k=1}^{n}w_{jk}^2-\sum_{k=1}^{n}(w_{ik}\cdot w_{jk})} \tag{2.9}$$

## 2.6 聚类算法

聚类分析作为一个活跃的研究领域，已经出现了很多聚类算法，总体上聚类算法可分为基于划分的方法、基于层次的方法、基于密度的方法等等。每种算法都有各自的优缺点，都有其适用的领域，并不是每一类算法都适合于文本聚类，我们必须根据文本数据的特点对聚类算法进行分析选择。

1) 基于划分的方法

基于划分的聚类算法是应用于文本聚类的最为普遍的算法。该方法(Partitioning Method) 将数据集合水平地分割为若干类，它根据设定的划分数目 k 选出 k 个初始聚类中心，得到一个初始划分，然后采用迭代重定位技术，反复在 k 个簇之间重新计算每个簇的聚类中心，并重新分配每个簇中的对象，以改进划分的质量。使得到的划分满足“簇内相似度高，簇间相似度小”的聚类原则。基于划分的方法的基本思想如图 2.2 所示。

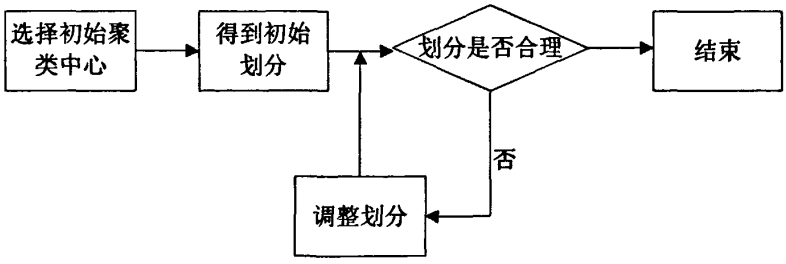


**图 2.2 划分算法基本思想**

最为常见划分方法是 k-means 算法[10]和 k-medoids 算法，两者的区别在于簇代表点的计算方法不同，k-means 以簇中对象的均值来代表每个簇，而 k-medoids 则是用该簇中离簇中心最近的对象来代表每个簇。针对 k-means 只能处理数值属性对象的不足，研究者提出了能够处理同时具有数值属性(numeric attributes)和分类属性(categorical attributes)的数据对象的方法 k-prototype[11]。一些文献提出了 bisecting k-means 算法[16]，该算法为了得到 k 个簇，使用 k-means 算法将所有对象的集合分成两个簇，再从这些簇中选一个继续分裂下去，直到生成 k 个簇为止。严格说来，bisecting k-means 应该属于层次划分的方法，但是作为 k-means 的直接扩充，我们把它和 k-means 一起考虑。

基于划分的方法的优点是运行速度快，但该方法必须事先确定 k 的取值。算法是一

个局部收敛的算法，不同的初始聚类中心选取得到的聚类结果变化很大。因此，我们可以通过研究如何选取合理的 k 值和初始聚类中心，来得到合理的结果；或者可以研究一种不需要预先指定种子数的方法，从而避开 k 值的选取对聚类结果的影响。

2) 基于层次的方法

基于层次的方法通过分解给定的数据对象集来创建一个层次。根据层次分解的方式，分为自下而上的凝聚式和自上而下的分裂式两种类型。凝聚式层次聚类最初将每个对象看作一个簇，然后逐步对簇进行合并，直到所有对象聚合为一个簇，或满足一定条件为止。分裂式层次聚类的作法与凝聚式层次聚类做法相反。它首先将所有对象看成一个簇，不断选择一个簇进行分解，直到所有对象均独自作为一个簇，或满足一定终止条件(如：需要的簇的数目，或两个最近簇的距离阈值)为止。

在两种方式中，凝聚式层次聚类较为常见，根据簇的临近性定义可以将其分为 Single-link，Average-link，Complete-link 等。层次聚类方法需要全局地比较所有类之间的相似度，算法复杂度较高，不适于大规模数据的聚类分析，比较适合于小规模的聚类分析或者对效果要求较高而对效率要求较低的聚类分析，也可与其他聚类方法结合，作为其他方法的一部分，如与 k-Means 结合，从数据集中选择一部分数据使用层次方法聚类，获得高质量的初始聚类中心，然后用这些质量较高的类中心为 k-Means 的初始类中心进行 k-Means 聚类。

层次聚类方法尽管简单，但经常会遇到如何选择合并点或分解点的问题。由于对簇进行分解或合并之后，无法回溯，所做出的合并或分解决策(在某一点上)不合适，导致聚类结果质量较差。将循环再定位与层次方法结合起来使用改变这种错误，提出了一些具有可扩展性的聚类算法，如：Birch[12]。

3) 基于密度的方法

大多数划分方法是基于对象间距离进行聚类的。这类方法仅能发现圆形或球状的聚类而较难发现具有其它形状的聚类。基于密度的聚类对噪声(异常数据)不明显，还可以发现任意形状的聚类。基于密度的聚类的指导思想是:只要一个区域中的点的密度大于某个阈值，就把它加到与之相近的聚类中去。其基本出发点是，寻找被低密度区域分离的高密度区域。

基于密度的聚类算法在当前的文献中较少被用于文本聚类中。这是由于文本间的相似度不稳定，同属一簇的文本，有些文本间的相似度较高，所以密度高；有些相似度较低，所以密度低。如果根据全局的密度参数进行判断，显然是不适合的。并且密度单元

的计算复杂度大，需要建立空间索引来降低计算量，且对数据维数的伸缩性较差。典型的基于密度的聚类算法有 DBSCAN[28]等。文献[29]给出了一种通过调节密度参数来发现不同密度文本簇的算法，把基于密度的算法应用于文本聚类。

4) 基于模型的方法

基于模型的方法基于一个假设，所有的数据都是由一些既定的概率分布混合生成。它通过为每个聚类假设一个模型来发现符合相应模型的数据对象。根据标准统计方法并综合考虑“噪声”或异常数据，该方法可以自动确定聚类个数，从而得到鲁棒性较好的聚类方法。基于模型的聚类方法主要有两种：统计方法和神经网络方法。基于SOM的文档聚类方法在数字图书馆等领域得到了较好的应用。对于高维的文本对象来说，研究表明多项式模型更加有效。

## 2.7 文本聚类质量的评价标准

在文本聚类分析中，按聚类评价指标所依据的标准，分为基于人工判定的指标和基于目标函数的指标[30]。

1) 基于目标函数的指标，聚类结果中，簇内越紧凑、簇间越分离越好；

2) 基于人工判定的指标，聚类结果与人工的判断结果越吻合越好。

两种方法对文本聚类的结果进行评价时可能得到不同的结论：在文本聚类的过程中，在采用向量空间模型表示文本时，必然导致语义信息的丢失，据此构造出来的目标函数并不能完全准确地反映人主观上的判断；同时，不同人对类别的定义会对同一个聚类结果产生不同的判断，根据标准 1)判定为好的结果在基于人工判定中不一定为优。

这两类标准各有不同的应用：基于人工判定的指标更适合于对文本聚类结果的效果进行评价，判断聚类算法是否更能满足应用的需求；基于函数的指标适合作为聚类算法本身的一部分而不是作为评价算法的标准，但基于目标函数的指标不适用于不同算法之间的横向比较。

### 2.7.1 基于人工判定的指标

对于给定的文本集合 $D$，包括类：$D=\{L_1,L_2,\cdots,L_P\}$,聚类后簇的集合定义为：$C=\{C_1,C_2,\cdots,C_q\}$。

1) 准确率和召回率及 FI 测度

召回率就是聚类正确的文本数与该类原有文本数的比值，准确率就是聚类正确的文

本数和实际分到该类的文本数的比值。准确率考察的是分类的精确度，而召回率考察的是分类的完备性。在评价聚类结果的质量时，两者必须同时考虑，由此，Van Rijsbergen 于 1979 年提出了 FI 值[31]，该方法综合考虑召回率和准确率，强调聚类的结果与事先定义的类别相似。表示形式如公式 3.3 所示。

对于一个簇$C_j$和人工分类$L_i$，$n_i$为$L_i$中文本的数目，$n_j$为$C_j$中文本的数目，$n_{ij}$为第 i 个类被划分到第 j 个簇的文本对象的数目：$|L_i \cap C_j| = n_{ij}$。

准确率　　$$presision(i,j) = \frac{n_{ij}}{n_j} \tag{2.10}$$

召回率　　$$recall(i,j) = \frac{n_{ij}}{n_i} \tag{2.11}$$

$$F(i,j) = \frac{2 * recall(i,j) * presision(i,j)}{recall(i,j) + presision(i,j)} \tag{2.12}$$

最终 F 值:　$$FI = \sum_i \frac{n_i}{n} \max\{F(i,j)\} \tag{2.13}$$

FI 值越大，聚类性能就越好。

2) 熵

熵(Entropy)[32]提供了一种对聚类结果度量的手段，用于度量某簇中包含某个相关类的程度。对于一个簇$P_j$,熵定义为：

$$E(j) = -\sum_i n_{ij} \log(n_{ij}) \tag{2.14}$$

熵越小，表示内部一致性越高。整个聚类结果的平均熵定义为:

$$E_{CS} = \sum_{j=1}^{m} \frac{n_j * E(j)}{n} \tag{2.15}$$

3) 平均准确率(Averaged Accuracy，简称 AA)，a，b，c 的定义如表 2.1 所示。

积极准确率　　$$PA = \frac{a}{a+c} \tag{2.16}$$

消极准确率　　$$NA = \frac{d}{b+d} \tag{2.17}$$

平均准确率　　$$AA = \frac{PA + NA}{2} \tag{2.18}$$

文献[33]中使用平均准确率指标评价聚类效果，平均准确率在文本聚类领域使用的并

不多。该指标强调准确率，尤其考虑了消极准确率，在实际实验分析中可以起到一定的作用，指标值相对较大。

表 2.1 评价标准的参数表

| | 属于该类的文本数 | 不属于该类的文本数 |
|---|---|---|
| 判断为属于该类的文本数 | a | b |
| 判断为不属于该类的文本数 | c | d |

### 2.7.2 基于目标函数的指标

基于函数的指标适于作为算法本身的一部分，用来判断当前的中间结果是否可以成为最终结果。近年来提出的一些有效性函数可以用来判断簇的最优个数，通过判断指标值是否大于前一次聚类结果的指标值，来决定当前的聚类数目是否正确，比如 Davies-Bouldin index，又称 DBI 指标。基于目标函数的指标还有很多，这种类型的指标不是本文讨论的重点，这里只介绍经常在 bisecting k-means 中用来衡量聚类质量的整体相似度[16](overall similarity)。

整体相似度

簇的紧凑性可以用来衡量聚类质量，用簇内各文本对象之间的相似度的均值来表示聚类结果的质量。

$$overallsim = \frac{1}{|D|^2} \sum_{\substack{d_i \in D \\ d_j \in D}} sim(d_i, d_j) \tag{2.19}$$

# 第三章 特征选择

特征选择在整个文本聚类过程有着很重要的地位。如果直接用分词和词频统计得到的特征词来表示文本向量的各个维，文本向量维数很高且很稀疏，会给聚类算法带来很大的计算量还会影响聚类结果的准确性。在聚类领域进行特征选择时，不能使用类信息来进行特征选择，针对该缺点，本文结合词性信息和文档频率方法，并且引入贪心算法的思想。提出了基于文档频率和单词贡献度的改进得特征选择方法。首先过滤对描述文本内容作用不大的词性；其次过滤文档频率过高及过低的特征；最后使用贪心算法增量的选择特征。

## 3.1 特征选择

特征选择就是按照一定的规则从原始特征集合中选择出一小部分最为有效的特征。特征选择在文本分类中已经得到了非常成功的应用[34][35]，提出了一系列优秀的方法，如信息增益和$\chi^2$可以在不降低分类性能的前提上过滤多达98%的特征词。特征选择在文本聚类的应用上不如在文本分类的应用上那么成功。由于聚类没有任何已知的信息可以利用，在文本分类领域应用很好的特征选择方法也大都不能应用于文本聚类领域。在聚类领域进行特征选择时因为缺乏类信息所以很难选择出最具区分能力的特征词。在移走的特征词到达一定比例后,在移走更多的特征词时，可能将区分能力较好的特征词移走，进而使得聚类性能急剧下降[36]。因此，用于文本聚类的无监督特征选择仍需进一步研究。

## 3.2 文本聚类中常用的特征选择方法

文本数据的特征选择研究的重点就是用特征评估函数给每个特征词评分，然后根据预先设定好的阈值来选择出其值超过这个阈值的所有单词，或者排序后选取预定数目评分最高的特征。用于文本聚类的特征选择方法主要包括两类，首先是原先应用于文本分类中但是不需要类信息的特征选择方法，比如文档频数和单词权；其次是一些为聚类问题专门提出的的特征选择算法，目前应用较多的方法包括:文档频率、单词权，单词熵和单词贡献度等。

1) 文档频率(DF)

文档频率是指在文本集中出现该词的文本数目，用 DF 表示。用 DF 进行特征选择

基于如下基本假设：出现次数过少的单词不含或含有较少的类别信息，删除这样的词条不但能够降低特征空间的维数，而且能提高聚类精度。文档频率最大的优势就是速度快，它的时间复杂度和文本数成线性关系，为 $o(n)$，所以能够容易地被用于大规模语料统计。使用文档频率删除 90%单词时，还可以保证文本聚类的性能不会大幅下降[36]。

使用文档频率可以过滤无效高频词(出现在大多数文本中的高频词)及出现次数过少的词，一般认为可以将少于三篇文章或多于30%的文章中出现的特征词忽略。姜宁，宫秀军等人使用文档频率过滤只在1篇文本出现，或在90%以上的文本中出现的特征词[37]。

本文使用文档频率进行第一步特征选择。

2) 单词权(Term Strength，TS)

单词权最早由 wilbur 和 sirotkin 提出[38]，后来应用在文本分类中，它计算的是一个词在一对相关文本中的某一个文本中出现的条件下，在另外一个文本中出现的概率。该方法认为一个词在相关的文本中出现得越多，在不相关的文本中出现得越少，就越为重要。

$$TS(t) = p(t \in d_j \mid t \in d_i) \qquad d_j, d_i \in D \cap sim(d_j, d_i) > \beta \tag{3.1}$$

$\beta$ 为相似度阈值，用来判断两个文本是否是相关的文本。

计算单词权必须首先计算所有文本之间的相似度，再根据设定好的阈值 $\beta$ 找出所有满足条件 $sim(d_j, d_i) > \beta$ 的文本对，因此，该方法时间复杂度相对较高。在进行特征选择之前文本间的相似度本身不够精确，阈值 $\beta$ 的调试也比较麻烦，但由于该方法不需要类信息，可以用于文本聚类。

3) 单词熵(Entropy-based Feature Ranking，En)

单词熵由 Dash 和 Liu 于 2000 提出[39]，它通过计算一个特征被移除之后整个数据的熵值来衡量该特征的重要性，该方法认为不同的特征对数据的结构或者说分布的影响是不同的，并且特征越重要，该特征对数据的结构的影响越大，而不重要的特征对决定数据的结构几乎没有什么贡献。

$$E(t) = -\sum_{i=1}^{n}\sum_{i=1}^{n}(S_{ij} \times \log(S_{ij}) + (1 - S_{ij}) \times \log(1 - S_{ij})) \tag{3.2}$$

$t$ 指的是某个特征，$S_{ij}$ 指数据 $d_i$ 和 $d_j$ 之间的相似度，

$$S_{ij}=e^{-\alpha\times dist_{ij}},\quad \alpha=-\frac{\ln(0.5)}{\overline{dist}}$$

$dist_{ij}$ 指的是当特征t被移除之后两个数据之间的距离，或者1减去余弦相似度；$\overline{dist_{ij}}$ 指的是当特征t被移除之后所有数据之间的平均距离。

单词熵的时间复杂度为 $o(mn^2)$，在文本数量很大且文本向量维数很高时计算量很大，通常在使用时需要用到采样技术，一般经过多次采样，将每次采样所得出的特征词的熵值累加作为该特征词最终的熵值。采样次数一般设置为35。

4) 单词贡献度(Term Contribution，TC)

单词贡献度[36]通过计算每个特征词对整个文本集相似性的贡献来衡量该特征的重要性。由下式，整个文本集的相似度可以看作所有特征词 $t$ 对整个文本集相似度贡献的累加：

$$\begin{aligned} sim(D)&=\sum_{d_i,d_j\in D\cap d_i\neq d_j} sim(d_i,d_j)\\ &=\sum_{d_i,d_j\in D\cap d_i\neq d_j}\sum_{t} f(t,d_i)\times f(t,d_j)\\ &=\sum_{t}\sum_{d_i,d_j\in D\cap d_i\neq d_j} f(t,d_i)\times f(t,d_j) \end{aligned} \tag{3.3}$$

因此，特征词 $t$ 的贡献度定义为：

$$TC(t)=\sum_{i,i\cap i\neq j} f(t,d_i)\times f(t,d_j) \tag{3.4}$$

其中 $f(t,d_i)$ 是特征 $t$ 在文本 $d_i$ 中的权重。使用 $ltc$ 模式的 $tf\times idf$ 权重计算，$idf$ 提高了文档频率低的特征词的权重，(式3.4)增加了文档频率高的特征词的累加次数，所以单词贡献度在单词的权重和文档频率之间取得一个平衡，出现次数过少和过多的特征词的贡献度都非常低，出现次数相对较多但权重也较高的特征词贡献度较高。

若令每个特征词的权重都相同，当某特征词在文本中出现时，权重置为1，$TC(t)$ 变为 $TC(t)=DF(t)(DF(t)-1)$，$TC$ 是 $DF$ 的单调递增函数，因此，$DF$ 可以看作 $TC$ 的特殊情况。

单词贡献度方法的时间复杂度为 $o(m\bar{n}^2)$，$\bar{n}$ 指所有特征词的平均文本频率，远小于文本总数 $n$。

5) 单词变化度(Term Variance，TV)

单词变化度[40]计算每个特征词在整个数据集的变化，在文本集中出现次数很少的特征及在文本集中分布方式单一的特征 TV 值低。

$$v(t)=\sum_{j=1}^{n}\left[f_j-\overline{f}\right]^2 \tag{3.5}$$

6) 信息增益(Information Gain，IG)

信息增益[41]作为一种比较好的有监督特征选择算法，由于需要类信息无法直接应用于文本聚类中，需要对公式作出一定的改变。本文将在第五章中使用信息增益为聚类后的文本簇提取关键词作为聚类标识。

信息增益在文本分类中描述为，文档特征词为整个类别所能提供的信息量。对于特征 $t$ 和文档类别 $c$，$IG$ 考察 $c$ 中出现和不出现 $t$ 的文档频率来衡量对 $c$ 的信息增益。特征 $t$ 对类别 $c$ 的信息增益记为：

$$IG(t,c)=P(t,c)\log\frac{P(t,c)}{P(t)P(c)}+P(\bar{t},c)\log\frac{P(\bar{t},c)}{P(\bar{t})P(c)} \tag{3.6}$$

$P(t)$ 表示文本集中存在特征词 $t$ 的概率，$P(c)$ 表示 $c$ 类文本出现的概率，$P(t,c)$ 表示文本属于类 $c$ 且文本中存在特征词 $t$ 的概率。

信息增益的不足之处在于它考虑了单词未发生的情况，虽然某个单词不出现也可能对判断文本类别有贡献，但实验证明，这种贡献往往远小于考虑单词不出现情况所带来的干扰。对于文本向量这样的高维数据，一个特征词出现或不出现所造成的影响是不同的，一个特征词在出现的情况下对文本的识别所起的作用要远大于该特征词在文本中不出现时的情况。

## 3.3 基于文档频率(DF)和单词贡献度(TC)的改进特征选择方法

在特征选择的过程中，我们希望所选的特征能够满足以下 3 个条件：

1) 能够覆盖到所有的文本。

2) 对于聚类来说，有利于簇的识别。

3) 最终选择的特征应该远小于初始特征集合。

针对聚类领域进行特征选择时可用信息少，阈值难确定的缺点，通过结合词性信息和文档频率方法，并引入贪心算法的思想，提出了基于 DF 和 TC 的改进的特征选择方法。方法首先使用词性信息和文档频率对特征集进行粗过滤；其次，以单词贡献度作为

评估函数，使用贪心算法增量的选择特征。算法流程如图 3.1 所示。

### 3.3.1 基于单词贡献度(TC)的增量的特征选择

目前所提出的特征选择方法一般都是利用评估函数对每个特征进行评估并选取排在前面评分值高的特征组成特征子集，其基本过程如下。

1) 初始情况下，特征集包含所有原始特征。

2) 根据评估函数给每个特征打分(并按降序排列)。

3) 选择前k个特征(k是要选取的特征数)或选择分值大于阈值的特征。

4) 根据选取的特征子集，对文本向量进行调整。

在聚类中不像分类中有类信息可以使用，仅凭词频信息无法像文本分类一样直接去除绝大多数不相关的特征。对于应该选取多少个特征目前还没有很好的方法，需要根据实验进行调试。k 值设置过高会产生过多的冗余特征，降低文本挖掘的质量；设置过低则会导致与文本内容高度相关的特征被除去，使最终得到的文本向量无法很好的代表文本。

随着特征的减少，在去掉冗余特征的同时也可能把与文本内容高度相关的特征去掉，但是如果不降低特征的数目，又会选中冗余的特征作为文本特征。通过引入贪心算法增量的选择特征[42]，每次都从当前特征集中选出最具代表性的特征，以避免这种问题。设 L 为一个正数，将所有特征词按照评估函数的得分降序排列。首先选择前 L 个分值最高的特征词，由于文本向量非常稀疏，这 L 个特征词不可能存在于所有的文本中，从文本集的词条矩阵中将包含这些特征词的文本去掉，并将这些特征词从总的特征集中去掉。如果词条矩阵中仍有文本未被覆盖，在新的词条矩阵中重新计算剩余特征词的得分并按降序排列，继续选择前 L 个特征。直到所有文本都被覆盖。

算法流程如下：

1) A 为文本集的词条矩阵，$N_d$ 为文本集 $D=\{d_1,d_2,\cdots,d_{N_d}\}$ 中文本的总数，$N_t$ 为特征集 $T=\{t_1,t_2,\cdots,t_{N_t}\}$ 中特征的总数。所选特征集合 $S$ 置为空。

2) 计算 $T$ 中所有特征的单词贡献度 TC 的值，选择得分最高的前 L 个特征，并入集合 $S$，并从特征集 $T$ 中将刚选出的特征取出。

3) 从 A 中去掉所有包含所选出的特征的文本。

4) 当 A 中没有文本时，到 5)，否则，到 2)。

5) 停止，$S$ 中即选出的特征。

### 3.3.2 基于 DF 和 TC 的改进特征选择方法的流程

由于直接对整个特征集使用基于 TC 的增量的特征选择方法计算量太大，所以应该首先对特征集进行粗过滤。在文本聚类中，由于没有类信息可以使用，很难对不相关的特征词进行过滤，词性信息的使用恰好可以帮助过滤一部分不相关的特征词。在进行特征选择时,首先去除对文本聚类没有用处的虚词,而在实词中,又以名词和动词对于文本类别特性的表现力最强，所以只选择文本中的名词和动词作为文本的特征词。由于本文使用的中科院分词系统具有词性标注的功能，可以在分词的同时进行词性过滤。

在只保留名词和动词作为特征词的基础上，过滤文档频率过低及过高的特征词。一般认为可以将少于三篇文章或多于 30%的文章中出现的特征词忽略。本文使用文档频率方法进行粗过滤时，取 0.05%至 1%中某一值，将出现频率低于该值的特征词过滤，同时过滤出现频率高于 90%的特征词。最后使用 3.3.1 节的基于单词贡献度的增量的特征选择方法对剩余特征进行筛选。因此，特征选择包括 3 个步骤，如图 3.1 所示。方法步骤如下。

1) 对整个文本集进行自动分词，在分词过程中对特征词进行词性过滤，只保留名词和动词特征。
2) 对特征词统计词频和文档频率，过滤文档频率过低及过高的特征词。
3) 使用基于 TC 的增量的特征选择方法进行第三次特征选择。


**图 3.1 基于 DF 和 TC 的改进的特征选择方法**

## 3.4 实验分析

实验中所用的文本数据由李荣陆老师收集的包括10个类别的新闻文本中选取(http://www.nlp.org.cn)。从中选择5个类别共450篇文本。经分词后共得到16467个词语，其中名词10416个，动词4304个，形容词989个，其他词类758个。只取其中的名词和动词作为特征词。聚类算法使用k-means，以F-measure作为评价指标，取运行10次的平均值作为最终结果。使用TC，DF和基于DF和TC的改进方法进行特征选择。

使用基于DF和TC的改进方法选择两组特征，第一组首先过滤在1%文本中出现的特征词，第二组首先过滤在0.3%文本中出现的特征词。再根据不同的L值进行特征选择。

特征信息如表3.1所示。

表3.1 使用基于DF和TC的改进方法所选的特征

| 第一组 | | 第二组 | |
|---|---|---|---|
| L | 特征数 | L | 特征数 |
| 2 | 296 | 1 | 156 |
| 3 | 375 | 2 | 320 |
| 6 | 468 | 4 | 424 |

实验结果如表3.2所示。从以上两组特征中各选出一个结果较好的与其他两种方法比较。使用DF进行特征选择时，随着特征数的减少，聚类质量有所提高；当特征数减少到一定程度时，随着特征的持续减少，聚类质量有所降低。使用TC得到的结果优于使用DF得到的结果，可以在保证聚类质量的情况下比DF过滤更多的不相关特征词。基于DF和TC的改进方法可以在过滤更多特征词的情况下得到较好的聚类结果。在比DF和TC过滤更多特征词的情况下，得到的结果优于DF和TC。算法在使用不同的L值时会得到不同的结果，有一定程度的波动，但聚类质量所受影响不大。

表 3.2 三种特征选择方法的 F-measure 值比较

| | 特征数 | F-measure |
|---|---|---|
| DF | 2036 | 0.68 |
| DF | 1251 | 0.71 |
| DF | 837 | 0.66 |
| TC | 800 | 0.72 |
| TC | 600 | 0.73 |
| 改进方法 | 424 | 0.73 |
| 改进方法 | 375 | 0.71 |

## 3.5 本章小结

本章对文档频率，单词贡献度，单词权等特征选择方法进行了分析与研究。对特征选则方法进行改进，针对在聚类领域进行特征选择时缺乏已知的可以利用的信息的缺点，分三步进行特征选择。第一步利用词性信息，只保留名词和动词作为特征项。第二步过滤文档频率过高及过低的特征词。第三步在使用单词贡献度对特征评分的基础上，使用贪心算法增量的选择特征。每次选择评分值最高的前 L 个特征后，同时去掉文本矩阵中包含所选特征的文本，在新的文本矩阵中对剩余的特征重新评分，迭代这个步骤，直到文本矩阵中没有剩余文本。实验表明，改进的特征选择方法与文档频率和单词贡献

度相比，在保证聚类质量的同时过滤了更多的不相关的特征词。算法在使用不同的 L 值会得到不同的结果，有小幅度的波动，但对聚类质量影响不大。

# 第四章　基于共享最近邻的改进的文本聚类算法

在第二章介绍的聚类方法中，基于划分的聚类算法 k-means 由于效率高在文本聚类领域得到了广泛的应用，但该算法需要确定初始聚类中心和簇的数目。k-means 的扩充版本 bisecting k-means 可以得到数据的层次划分，不但聚类质量较好，且算法复杂度比较低，但该算法需要在每一步选择下一次需要分裂的簇。由于文本对象具有高维性和稀疏性的特点，文本对象的邻近度趋向于一致，相似性不易度量，使依赖于相似度的聚类方法不能产生理想的聚类结果。针对以上问题，借鉴 JARVIS，Gowda 等引入的共享最近邻的概念[43]，在计算相似度时考虑了文本周围与它相似度比较大的文本。针对 k-means 提出一个新的选择初始聚类中心的方法，并对 bisecting k-means 提出一个新的选择下一次分裂的簇的方法。

## 4.1 问题的提出

由于文本对象具有高维性和稀疏性的特点，文本对象间相似度普遍较低，不同类别文本所包含的特征词存在一定程度的重叠，因此，与一个文本相似度最大的文本可能与该文本属于不同的类别。图 4.1 为几个经典数据集中文本与其最近邻属于不同类别的比例。在这种情况下，相似度成为不可靠的指导，影响聚类质量。尤其对于凝聚层次聚类，将错误的对象归为一个簇后，无法回溯[16]。

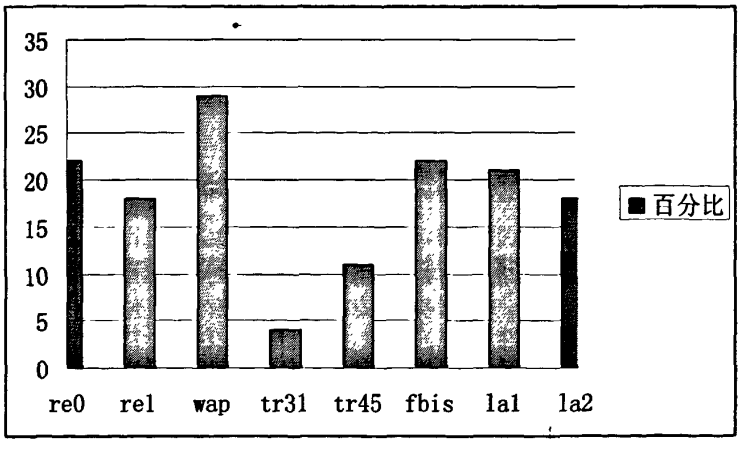


**图 4.1　文本与其最近邻不属于同一簇的比例**

大多数情况下(由图 4.1)，一个文本与其最近邻仍属于同一簇。因此，可以通过定义更合适的临近性度量来克服这种困难。JARVIS，Gowda 等引入了共享最近邻(SNN，shared nearest neighbor)的概念，间接的度量对象间的相似性，如果两个对象与相同的对

象中的大部分都相似，它们也相似。这种相似度度量方式可以解决高维数据相似度普遍比较低带来的问题，Gchat 在文献[44]中提出了基于共享最近邻的层次聚类算法(ROCK：Robust clustering using links)，成功的应用于事务数据聚类。

k-means 及其变种 bisecting k-means 在迭代过程中计算文本与簇中心点的相似度，这类似于计算了和簇中所有文本的平均相似度，克服了两个文本间相似度不可靠的缺点，一定程度上改善了聚类结果。k-means 还可以用来优化层次聚类方法的聚类结果。但是 k-means 受初始化影响，聚类结果容易出现大幅波动，有时候会得到不合适的结果；bisecting k-means 选择下一次分裂的簇时，如果选择的不好亦会影响最终的结果。

本文借鉴 Gchat 提出的基于共享最近邻的聚类方法(ROCK)中计算相似度的方法，对 k-means 提出一个新的选择初始聚类中心的方法，计算两文本的相似度时，与两文本都相似的文本越多，则两文本越相似。对 bisecting k-means 提出一个新的选择下一次分裂的簇的方法，引入邻居的概念度量每个簇的紧密性，并选择紧密性最差的簇作为下一次分裂的簇。

## 4.2 基于共享最近邻的层次聚类算法

共享最近邻(SNN)的概念最先由JARVIS提出，关键思想是：在定义相似度度量的时候考虑点的环境，SNN就是两个对象共享的近邻个数[43]。基本的临近性度量可以是任何有意义的相似性或相异性度量。Gchat针对事务数据的聚类问题提出了基于共享最近邻的凝聚的层次聚类算(ROCK：Robust clustering using links)。Gchat通过判断两个点的相似度是否满足一定条件，如果满足条件，则两点互为邻居。

### 4.2.1 邻居(Neighbors)

$sim(p_i, p_j)$ 表示 $p_i$ 和 $p_j$ 之间的相似度，$sim$ 可以是任何一种相似度度量甚至是非数值度量(eg：领域专家提供)。本文采用余弦相似度度量文本间的相似度，因此 $sim$ 的值大于 0 小于 1 且值越大相似度越高，给定一个阈值 $\theta(0<\theta<1)$，若满足(式 4.1)，则称 $p_i$ 和 $p_j$ 互为邻居[44]。

$$sim(p_i, p_j) \geq \theta, \tag{4.1}$$

$\theta$ 为用户指定的参数。假定当 $sim$ 为 1 是代表两点完全相同为 0 时代表两点完全不同，当 $\theta$ 定义为 1 时，每个点被限制只与其本身互为邻居，当 $\theta$ 定义为 0 时，任何一对

点可以定义为邻居。根据应用中对相似度的要求，用户可以选择适当的$\theta$值。

### 4.2.2 链接(link)

$link(p_i,p_j)$表示$p_i$和$p_j$的共同邻居的个数，也是一种共享近邻的概念[44]，如(式 4.2)所示：

$$link(p_i,p_j)=\left|\left\{x \mid sim(x,p_i)\geq\theta \wedge sim(x,p_j)\geq\theta\right\}\right| \tag{4.2}$$

$link$越大则$p_i$和$p_j$越相似，越有可能属于同一簇。在聚类的过程中引入$link$来度量点的相似度，与大部分聚类算法在聚类过程中只计算点与点之间的相似度不同，本方法在计算相似度时考虑了点邻域的信息，两个点之间$link$越大，两个点所包含的相同的信息越多，可以避免两个簇由于簇间几个孤立点的相似度比较大而被合并，这种度量方式可以有效的克服上文所提出的传统相似度在高维数据上的问题。

### 4.2.3 簇间相似度度量及评价准则

$link(C_i,C_j)$表示簇$C_i,C_j$内所有点的$link$值的总和[44]，如(式 4.3)所示：

$$link(C_i,C_j)=\sum_{p_p\in C_i,p_r\in C_j} link(p_p,p_r) \tag{4.3}$$

$g(C_i,C_j)$表示簇$C_i,C_j$之间的关联，值越大，簇$C_i,C_j$越相似[44]，如(式 4.4)所示：

$$g(C_i,C_j)=\frac{link(C_i,C_j)}{(n_i+n_j)^{1+2f(\theta)}-n_i^{1+2f(\theta)}-n_j^{1+2f(\theta)}} \tag{4.4}$$

在聚类的过程中，ROCK 选择$g(C_i,C_j)$值最大的两个簇进行合并。由于较大的簇比较容易有更多的邻居，较小的簇比较容易有较少的邻居，在合并的过程中，较大的簇比较容易与和其他簇的合并，为避免在聚类时小簇被大簇合并以及孤立点的影响，(式 4.4)用簇间实际的$link$值除以簇间$link$值可能的最大值。ROCK 的聚类过程属于凝聚的层次聚类方式，在聚簇合并后，同步修改簇间的$link$值，具体的算法流程参见文献[44]。ROCK 对聚簇的评价采用(式 4.5)。

$$E_l=\sum_{i=1}^{k} n_i\times\sum_{p_r,p_q\in C_i}\frac{link(p_r,p_q)}{n_i^{1+2f(\theta)}} \tag{4.5}$$

## 4.3 基于共享最近邻的改进的 k-means 算法

k-means 算法是基于原型的聚类算法，在基于原型的聚类中，簇是点的集合，其中

每个点到定义该簇的原型的距离(相似度)比到其它簇的原型的距离(相似度)更近(大)，对于具有连续属性的数据，簇的原型通常是质心，即簇中所有点的平均值。k-means[10]最初由 J.B.MaeQueen 于 1967 年提出的。由于它易于理解、效率较高，在科学研究以及工业界都得到了广泛的应用。

4.3.1 k-means 算法思想与流程

k-means算法的基本思想是，根据参数k，随机选取k个点作为初始聚类中心，根据每个点与初始聚类中心的距离将所有点划分为k个簇，以每个簇的质心作为新的聚类中心，不断地迭代以上的步骤，对簇进行调整，使簇内对象之间的距离尽可能小，而簇间对象之间的距离尽可能大，直至目标函数收敛。

k-means算法的流程如下：

1) 随机选取初始聚类中心 $Z_1^{(0)}$，$Z_2^{(0)}$，…，$Z_k^{(0)}$，迭代次数 $t=0$；

2) 对于数据集 $S_n=\{x_1,x_2,\cdots,x_n\}$ 中每个点 $x_i$，与聚类中心 $Z_j^{(t)}$ 比较,根据与聚类中心的相似程度将其分配到最相近的一类中,即：

$$j^*=\arg\max_j x_i^T Z_j^{(t)},1\le j\le k，\ t\text{为迭代次数。} \tag{4.6}$$

3) 由步骤2)得到新的簇分区集合 $C^{(t+1)}=nextKM(C^t)$ ,计算新的聚类中心

$$Z_j^{(t+1)}，\ 1\le j\le k \tag{4.7}$$

4) 如果 $Q(C^{(t+1)})-Q(C^{(t)})>\varepsilon$ ($\varepsilon>0$ 为判断终止的阈值),则 $t=t+1$,转到2)；否则终止算法,输出聚类最终簇的集合 $C^*$。

Dhillon将k-means用于文本聚类领域[18],利用余弦相似度来计算文本间的距离,成为目前常用的文本聚类算法。在文本聚类中采用的目标函数如(式4.8)所示：

$$Q(C)=\sum_{j=1}^{k}q(c_j)=\sum_{j=1}^{k}\sum_{d\in c_j}Z(C_j) \tag{4.8}$$

其中 $q(C_j)$ 为对应簇 $C_j$ 内部聚类(intra-cluster)的相似度，如(式4.9)所示：

$$q(C_j)=\sum_{d\in c_j}d^T Z(C_j)=\|S(C_j)\| \tag{4.9}$$

在该算法中，每次迭代中把每一个数据点分到离它最近的聚类中心所代表的簇，这个过程的时间复杂度为 $O(nkd)$，这里n指的是数据集中数据点的总数，k是指定的聚簇数，

d是数据点的维数；新的聚簇产生以后需要计算新的聚类中心，这个过程的时间复杂度为$O(nd)$。因此，这个算法一次迭代需要的总的时间复杂度为$O(nkd)$。

### 4.3.2 k-means 算法特点与缺点

k-means 算法采用两阶段反复循环，结束的条件是目标函数收敛。它具有复杂度低的优点，不仅效率高且可伸缩性强，可以用来处理大数据集。应用于文本聚类时，得到的结果通常是可以理解的。当结果簇是密集且簇间区别明显时，聚类结果较好。

k-means算法受初始聚类中心及聚簇个数的影响，并且受孤立点影响比较大，本文主要讨论如下两个缺点：

1) k-means算法的结果依赖于由初始聚类中心出发所遇到的第一个局部极值点，不同的初始聚类中心很可能导致截然不同的聚类结果。一旦初值选择不好，可能无法得到有效的聚类结果。

选择初始点一般有经验选择、随机选择、最大最小原则等方法。经验选择带有主观性，同时给用户增添了负担。随机选择可能选取“孤立点”、类边缘点，或者一个类中选取了两个以上的对象作为初始聚类中心，结果不理想。

Bradley提出RA方法[45]，通过多次抽样并对每个样本集用k-means聚类，再从中选择效果最好的作为最终的初始中心。该方法通过取样降低了算法的时间复杂度，然而，对样本集用k-means聚类时初始中心是随机选择的，如果选择不好，将会影响最终的聚类结果；且样本集的数目也会影响聚类的结果。

Zhuang 提出的 MFI k-means 算法[24]根据最大频繁词集生成 k-means 初始条件，首先得到最大频繁词集，将包含某一最大频繁词集的所有文档的中心作为 k-means 的一个初始中心。MFI k-means 克服了高维数据处理的复杂性。然而，该方法倾向于选择长文档，因此 MFI 只使用文档的前 300 个词计算初始中心点。但是对于长文档,仅利用前 300 个词无法很好地解释该文档的主题。

使用最大最小原则选择初始聚类中心[46],[47]，选择的聚类中心比较分散，具有较好的代表性，克服了随机选择的盲目性；减少了对经验知识的依赖；也不会受到文档长度的影响。因此本文选择最大最小原则作为初始聚类中心的选择方法。

2) 需要用户事先给出簇的个数k，而这个信息通常在聚类之前无法获得。这样就可能出现以下的情况，原本属于同类的对象被强行拆分到不同簇中，或是原本不属一类的

对象却被强行合并至一个簇中。

文献[48]通过dbscan算法确定簇的数目并给出k个簇的聚类中心，在对数据集进行多次抽样后使用dbscan方法进行聚类，再对聚类结果进行合并，得到k值并同时得到k个簇的聚类中心。本文使用最大最小原则在寻找初始聚类中心时同时确定聚簇数k[49]。

#### 4.3.3 基于最大最小原则的初始点选择

最大最小距离算法[50]是模式识别领域中一种比较简单的聚类分析方法，算法思想被用来为 k-means 选择初始中心点。最大最小原则依据待聚类对象的相似情况选择距离尽可能远的对象作为初始聚点，克服了随机选择的盲目性；减少了对经验知识的依赖；也不会受到文本长度的影响。最大最小原则选择聚类中心的基本原理是，假设要将样品分成 k 个类别，选取距离最远的两个点作为初始聚类中心后，其余聚类中心的选则用递推公式表达。具体流程如下：

1) 初始中心点集M初始化为空集，即 $M=\{\ \}$。在数据集 $S_n=\{x_1,x_2,\cdots,x_n\}$ 中，任选一点，例如 $x_1$ 作为第一个簇的中心点 $m_1$，$M=M\cup\{m_1\}$

2) 从集合 $S_n$ 中找出到 $m_1$ 相似度最小的点作为第二个类的聚类中心 $m_2$，$M=M\cup\{m_2\}$

3) 选择满足(式4.10)的点 $m_i$ 作为第 $i$ 个中心点，

$$d(m_i,q)=\min\{\max\{d(x,q),q\in M\},x\in S_n\setminus M\} \tag{4.10}$$

$m_i\in S_n\setminus M$，$d(x,q)$ 代表点 $x$ 与 $q$ 的相似度(余弦相似度)。

4) 将点 $m_i$ 并入中心点集，$M=M\cup m_i$

5) 重复3，4直到找到 $k$ 个中心点，即 $|M|=k$

本文结合最大最小原则的优点，提出一种新的初始点选择方法。

#### 4.3.4 基于共享最近邻的改进的初始点选择方法

最大最小原则选择初始中心点存在一定的问题，最大最小原则依据待聚类对象的相似情况选择距离较远的对象作为聚类中心，然而，第一个聚类中心为随机选取，聚类结果受第一个聚类中心选择的影响，会出现一定程度波动。且有可能将孤立点选为聚类中心。

基于最大最小原则选择初始聚类中心时，仅仅依靠相似度，这可能将孤立点选为初始聚类中心。对于初始聚类中心，不但希望分布得尽量散，以保证每类文本都能有相应的聚类中心，而且希望这些中心点具有一定的代表性，可以很好的表征该类别的文档，即与本类文档绝大多数文档的相似度要明显大于和其他类中文档的相似度，这要求所选点能具有一定的密度。

基于最大最小原则选择初始聚类中心，在选定第一个聚类中心后，后面的聚类中心都是根据与前面已得出的聚类中心的相似度(距离)递推得出，相似度的不可靠直接影响所选聚类中心的代表性。由 3.1 节所述，文本向量维数很高，传统的相似度度量在高维数据上存在一定的问题，与某文本相似度最大的文本可能与该文本不属于同一簇，相似度成为不可靠的指导。这导致在初始聚类中心的选择过程中，虽然选择的是与前面聚类中心的相似度较小的文本作为下一个聚类中心，但也可能选择了与已选出的聚类中心属于同一簇的文本。

现通过一个例子进行说明。在选择第 i 个聚类中心时，假设文本 a 和 b 是已选出的前 i-1 个聚类中心中的两个文本，文本 d 是文本集中除去已选聚类中心后的任一文本。假设文本 d 和文本 a 的相似度大于和文本 b 的相似度，然而，在这种情况下，文本 d 仍有可能与文本 b 属于同一簇。假设文本 d 与文本 a 的相似度恰好是各个点和所有 i-1 个聚类中心中最小的，文本 d 在被选为第 i 个聚类中心时，它所代表的类别与文本 b 相同，并不能代表一个新簇。

根据上文所提出的缺点，对初始聚类中心的选择做出如下改进：

1) 引入密度的概念[47]，首先，选取文本集中密度最大的点作为第一个初始聚类中心，避免中心点选择的随机性。该点密度最大，与较多文本的相似度较大，更有可能代表一个簇。由于第一个聚类中心为确定的点，其余聚类中心使用最大最小原则选择，也是确定的，这就消除了聚类中心选择的随机性。其次，从文本集中选择密度比较大的文本组成集合 $S_\mu$，后面的聚类中心在 $S_\mu$ 中选择，以避免将孤立点选为聚类中心[47]。

对文本集 $S_n=\{d_1,d_2,\cdots,d_n\}$，计算每个文本对象与其它文本对象的平均相似度，如(式 4.11)，再计算 n 个文本对象的平均值，如(式 4.12)。

$$s_i=\frac{1}{n}\sum_{j=1}^{n}sim(d_i,d_j) \tag{4.11}$$

$$s' = \frac{1}{n}\sum_{i=1}^{n} s_i \tag{4.12}$$

选出$s_i$值中最大的文本作为第一个聚类中心，选择满足条件$s_i \geq \alpha s'$的文本$d_i$组成文本集$S_\mu$，后面的聚类中心在$S_\mu$中选择。

2) 将4.2节基于共享最近邻的聚类算法(ROCK)的思想应用于中心点的选择过程中。引入其中邻居及共享最近邻的概念，对文本对象的相似度进行度量。由于计算每对文本共同的邻居时，直接对所有文本对计算共享最近邻的个数计算量过大，因此，首先通过最大最小原则选出一部分点作为备选聚类中心，再在备选点中使用共享最近邻度量文本的相似性。第一步，假设聚簇数为k，使用余弦相似度度量文本对象之间的相似度，基于最大最小原则从$S_\mu$中选择$m = k + n_+$个点作为备选初始聚类中心($n_+$的确定将在第五章实验分析中进行讨论)，记为$S_m$。第二步，使用共享最近邻度量文本对象间的相似度，在$m$个备选初始聚类中心中进一步选择出$k$个点作为最终的聚类中心点集，表示为$S_k$。

对备选初始聚类中心点集中的每对文本$p_i, p_j \in S_m$，计算在整个文本集$S_n$中的共同的邻居(共享最近邻)的个数$link(p_i, p_j)$，如(式 4.13)所示，并计算每个点$p_i$与其他点的$link$值的总和$link(p_i)$，如(式4.14)所示。

$$link(p_i, p_j) = \left| \{ x_j \mid sim(p_i, x_j) \geq \theta \wedge sim(p_j, x_j) \geq \theta, x_j \in S_n \} \right| \tag{4.13}$$

$$link(p_i) = \sum_{j=1}^{m} link(p_i, p_j) \tag{4.14}$$

选$S_m$中 $link$值最大的点$p_i$作为第一个簇的中心点$m_1$，在其他备选中心点中选出与$m_1$共同邻居数最少的点作为第二个簇的中心点$m_2$，其余聚点的选取用递推方式选择，选择第$i$个中心点时，对备选中心点集中剩余的每个点$p_j (p_j \neq m_1, m_2, \cdots, m_{i-1})$，计算其与前$i-1$个已选定的中心点$link$值的总和$L$：

$$L = \sum_{k=1}^{i-1} link(p_j, m_k) \tag{4.15}$$

选出$L$值中最小的点$p_j$作为第$i$个中心点$m_i$，直到选够$k$个点为止。

### 4.3.5 聚簇数目 k 的确定

由 4.3.2 节的讨论，选择合适的簇数是正确聚类的前提。很多情况种假设聚类数目 k 已知，或由用户根据经验知识指定 k。然而很多时候，用户对文本集了解有限，事先并不知道应该将文本集分为多少簇。采用文献[49]的方法，使用最小最大原则判断聚簇数 k。将 k 的确定作为一个可选的步骤加入到 k-means 的初始聚类中心的选择过程中。

设已选出$i-1$个聚类中心，$M$为已选出的中心点集，下一个聚类中心$m_i$的选择取决于以下指标：

$$D_{min}=\min\left\{\max\left\{d(x_i,q_j),q_j\in M\right\},x_i\in S_n\setminus M\right\} \tag{4.16}$$

$|M|=i-1$。使用最大最小原则选择中心点的过程中，在选择的点的个数小于真实簇数前，$D_{min}$所计算的相似度基于类间相似度，相似度较小；在大于真实簇数后，$D_{min}$所计算的相似度基于类内相似度，相似度较大。因此，在到达真实簇数后，$D_{min}$值会出现较大的波动。图4.2为实验中的一组数据，横坐标代表所选的点的个数，纵坐标为$D_{min}$随着聚类点的增加出现变化。可以看出，变化幅度最大的点与前一点组成的线段的斜率最大，因此，可以使用最大最小原则找出L个点，再计算这L个点中每个点与它前后两点$D_{min}$值的差的和深度$D_L$，该值越大，说明在该点处$D_{min}$的波动越大。$D_L$值最大的点即为最终的聚簇数k。聚类的个数k一般小于样本个数的平方根[51]，因此，一般可以取L为文本总数的平

方根,算出每个点的$D_{min}$值，再从中找出$D_L$值最大的点。

### 4.3.6 基于共享最近邻的改进的 k-means 算法的流程

根据上述选取初始聚类中心的方法，得到基于共享最近邻的改进的 k-means 算法。由于方法首先计算了文本对的相似度矩阵，使用最大最小原则选择初始聚类中心时不需要重新计算，只需在相似度矩阵中搜索即可，算法流程如图 4.3 所示。

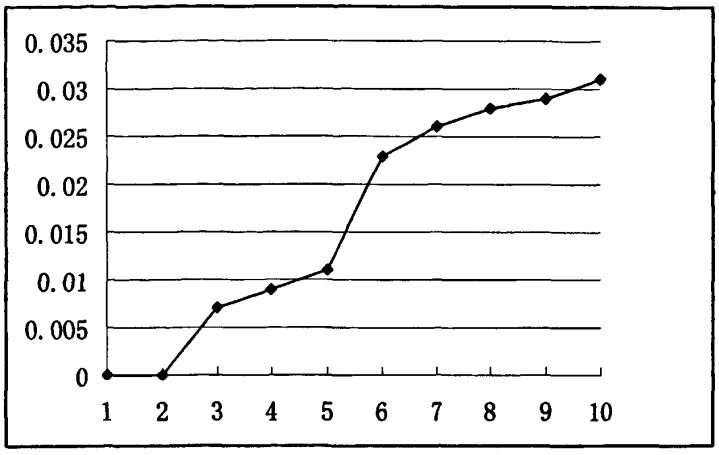


**图4.2 $D_{min}$ 随聚类点数变化的曲线**

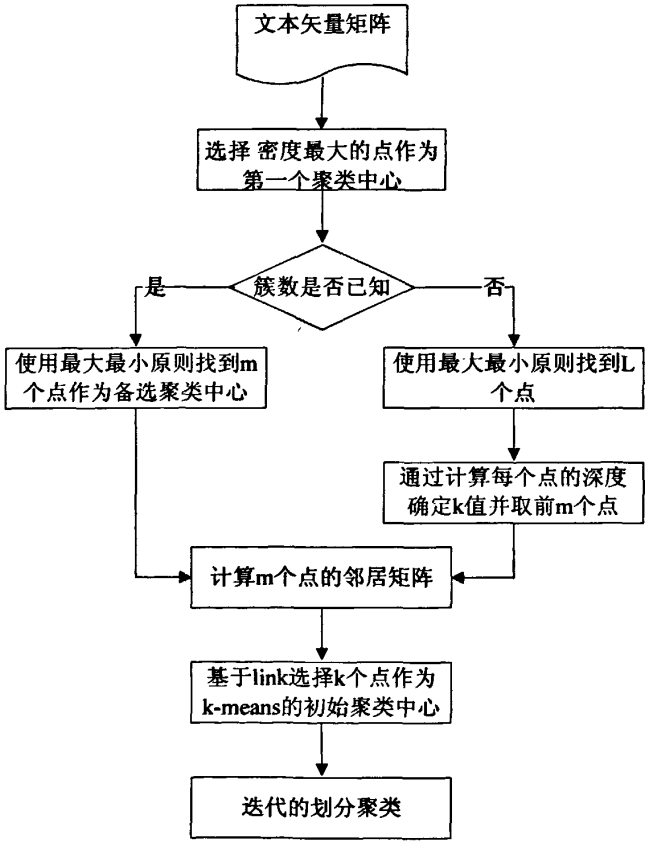


**图4.3：改进的k-means聚类流程**

## 4.4 基于共享最近邻的改进的 bisecting k-means 算法

### 4.4.1 bisecting k-means 算法的思想及流程

bisecting k-means[16]算法是 k-means 算法的直接扩展，它基于一种简单的想法，为了得到 k 个簇，首先将所有点的集合分裂成两个簇，从这些簇中选择一个继续分裂，如此下去，直到产生 k 个簇，与 k-means 不同，bisecting k-means 严格说来是一种分裂的层次聚类算法，可以得到嵌套的文档类别结构，以树的形式表示，这种层级结构比较符合文本集本身的特性，且该算法不会受到初始化的影响。算法不但时间复杂度低，同时也能得到良好的聚类结果。文献[16][17]指出，使用 bisecting k-means 得到的文本的层次划分优于传统的 UPGMA。本文对 bisecting k-means 进行研究，得到文本集的层次划分。

对分裂式层次聚类算法的研究主要集中在，如何选择一个簇作为下一次分裂的对象，以及如何对所选的簇进行划分。本文主要研究第一个问题。针对文本数据的特点，将共享最近邻的概念用于选择下一次分裂簇的过程中。

bisecting k-means 算法的流程。

1) 初始化簇表，使之包含由所有的点组成的簇。

2) 从簇表中取出一个簇(对选定的簇进行多次二分“试验” 试验次数定为(M)，设定试验次数 T=0。

3) 如果 T<M，转到 4)，否则转到 5)。

4) 使用 k-means 算法，将选定的簇分为 2 个簇，T=T+1，转到 3)。

5) 从二分试验中选择具有最优划分的两个簇，将这两个簇添加到簇表中。

6) 如果簇表中包含 k 个簇，停止，否则转到 2)。

### 4.4.2 改进的 bisecting k-means 算法

对下一次需要分裂的簇进行选择的准则有以下 3 种。

1) 选择包含对象最多的簇

2) 选择整体相似度最低的簇

3) 将以上 2 种准则结合

准则 1)实现简单，得出的结果中各个簇的大小相差不大。准则 2)应用整体相似度选择下一个需要分裂的簇，考虑了簇的质量，是应用最广泛的准则。在 bisecting k-means 每次分裂的过程中，依据某一准则从现有的簇中选择一个簇作为下一次分裂的对象，选择的目的是要从现有的簇中选出质量最差的簇，即该簇中的文本并不是与彼此相似度很

高，且它们之间的联系很小。文献[16]中在使用以上准则对文本进行试验时，发现各个准则得到的结果差别很小，因此作者建议使用准则 1)进行选择。然而用簇的大小作为选择的准则忽略了对簇质量的评价，当在两个簇中进行选择时，其中一个簇中文本之间相似度较小而另一个簇中文本之间相似度较大，即使第一个簇中对象较少我们也应该选择第一个簇作为下一次分裂的对象。

紧密性较差的簇中的文本极有可能属于不同的簇，对其进行分裂有利于提高聚类质量。因此，在选择下一次分裂的簇时，应该根据簇的紧密性进行选择，本文引入基于共享最近邻的聚类算法(ROCK)中邻居的概念[44]，对簇的紧密性进行度量，通过比较各个簇的中心向量的邻居数选择下一次分裂的簇。

度量一个簇的紧密性，只需考虑簇中心向量的局部邻居，即只考虑簇中心向量在它所代表的簇中所包含的邻居的数目。对于一个簇$C_j$的中心向量$c_j$，它的邻居数定义如(式 4.17)：

$$N(c_j)_{local}=\{sim(c_j,x_i)\geq\theta\,|\,x_i\in C_j,i=1,2,\cdots,|C_j|\} \tag{4.17}$$

对于同样的相似度阈值$\theta$，紧密性好的簇的中心向量的邻居数多于紧密性差的簇。各个簇的大小通常不同，文本比较多的簇的中心向量邻居数自然比较多，这会导致文本比较多的簇容易被首先划分，因此，需要对中心向量的邻居数进行归一化。用簇中心向量$c_j$的邻居数除以簇中文本的总数，即：

$$G(c_j)=N(c_j)_{local}/|C_j| \tag{4.18}$$

在选择待分裂簇的过程中，选择$G(c_j)$值最小的簇作为下一次分裂的簇。

改进后的算法流程如下：

1) 所有文本作为一个簇$c_1$，初始化簇表$C=\{c_1\}$

2) 根据$G(c_i)$值从簇表中取出一个簇$c_i$作为下一次分裂的簇

3) 使用 k-means 算法，将簇$c_i$分为 2 个簇$c_i$，$c_{i+1}$，T=T+1，使 T=0

4) 如果 T<M，到 3)，否则到 4)

5) 选择分裂效果最好的一次，将其并入簇表$C$

6) 计算簇$c_i$和$c_{i+1}$的 $G(c_i)$值，并保存

7) 如果簇表中包含 k 个簇，停止，否则，转到 2)

## 4.5 本章小节

由于文本数据具有高维性和稀疏性的特点，传统的相似度难以准确的度量文本对象的相近程度。对 k-means 及其变种 bisecting k-means 进行研究，结合最大最小原则与共享最近邻相的概念，在对 k-means 选择初始聚类中心时，首先基于最大最小原则选择多于聚簇数 k 的对象组成备选初始聚类中心点集，再对各个备选中心使用共享最近邻度量对象间的相似度，最后从中选择相似度较小的 k 个点作为聚类中心。在聚类数未知的情况下结合最大最小原则确定聚类数。在 bisecting k-means 中，用簇中心向量与簇中其它对象的邻居数度量该簇的紧密性，选择紧密性最差的簇作为下一次迭代时分裂的簇。算法的参数选择和实验结果在下章讨论。

# 第五章 基于 k-means 的中文文本聚类的实现与实验

前文对文本聚类中的相关技术进行了研究与分析。对特征选择方法进行了研究，提出了基于 DF 和 TC 的改进的特征选择方法。对聚类算法进行了研究，针对 k-means 的初始化问题进行了研究与改进，提出基于共享最近邻的改进的 k-means 算法；对 bisecting k-means 选择下一次分裂的簇的问题进行了研究与改进，提出基于共享最近邻的改进的 bisecting k-means 算法。

将上述技术和提出的算法应用到本文的基于 k-means 的中文文本聚类中，实现了文本聚类原型系统，并通过实验对上一章提出的改进算法与原算法进行比较。系统为研究文本聚类的相关技术提供了实验平台，包括文本解析和聚类两大部分。综合考虑研究和应用的需要，对各模块进行细致的模块化设计，使系统具有良好的扩展性和替换性。

## 5.1 软硬件环境

硬件环境：Pentium(R)CPU；256M 内存；40G 硬盘

软件环境：Windows XP 操作系统； JDK1.5

## 5.2 系统设计

### 5.2.1 语料库

语料库是指按照一定的语言学原则收集出的具有一定容量的大型电子文本库。语料库中存放的是在语言的实际使用中真实出现过的语言材料，真实语料需要经过加工才能成为有用的资源。

本文采用两个语料库，包括，复旦大学语料库，在“中文自然语言处理开放平台”可下载，包括环境、计算机、交通、教育、经济、军事、体育、医药、艺术、政治十个类别共 2816 个文本。

搜狗语料库：在 http://www.sogou.com/上下载的 sogou reduced 语料库，包括:新闻、科技、体育、军事、医药、文学、教育、旅游、杂类、艺术等十个类别，每类有近 2000 个文本。

语料库中的文本按照其所属类别以目录树的结构存储。实验将根据具体情况从语料中选择不同的类别和样本个数。经测试，所采用两个语料库中所有文本的平均相似度不

同。

### 5.2.2 功能模块设计

包括两大模块，第一个模块为文本解析模块，将文本表示为向量形式，第二个模块为聚类模块，对文本向量进行聚类分析。每个模块又包括几个子模块。文本解析模块包括预处理部分，特征处理部分。首先对文本进行分词及停用词过滤，再对文本的特征进行处理，包括词频统计，特征选择及权重计算，最终将文本表示成向量形式。聚类模块在对文本向量进行聚类分析的同时对每个簇生成聚类标识。

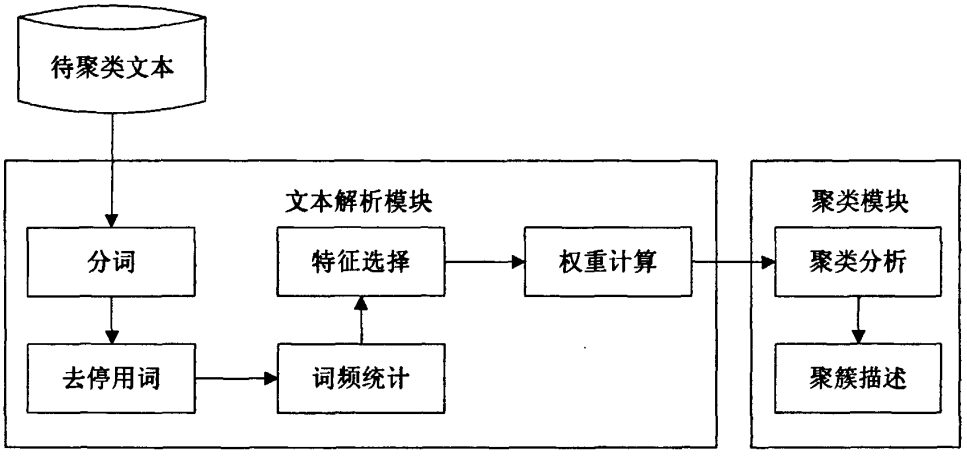


**图 5.1 文本聚类过程**

1) 文本解析模块

对文本进行分词，同时统计一些参数，包括文本的类别信息、文本总数、词频和文档频率，同时进行去停用词处理，进行粗降维。再利用特征评估函数对所有特征进行评价并选择满足条件的特征，降低向量空间的维数，避免噪音词对聚类效果的影响。

◇ 中文分词 本文采用中国科学院计算机技术研究所的 ICTCLAS 系统进行分词，系统对分好的词进行了词性标注。中科院分词系统的主要思想是利用基于层叠隐马模型的方法,将汉语分词、切分排歧、未登录词识别、词性标注等词法分析任务融合到一个相对统一的理论模型中。

◇ 去停用词 由于本文使用的中国科学院计算机技术研究所的 ICTCLAS 系统对于分好的词进行了词性标注，本文在分词后只保留名词和动词作为特征词。

◇ 频率统计 统计特征词的词频和文档频率。

◇ 特征选择 通过特征评估函数对特征词进行评分，选择分值较大的词构成特征

项。提供 DF、TC 和基于 DF 和 TC 的改进的特征选择方法三种特征选择方法。在使用 TC 和基于 DF 和 TC 的改进的特征选择方法时，都首先使用 DF 进行初次特征选择，由于后面还会进行特征的进一步选择，可以将参数适当放宽，仅过滤在 0.5%以下文本中和 90%以上文本中出现的特征词。由于 DF 方法具有速度快的优点，可以减少后续特征选择方法的运行时间，不会影响后续方法的质量。

◇ 权重计算 利用权重计算函数为选出的特征项赋予权重(重要程度系数)。采用 ltc 权重计算方法。

◇ 文本向量调整 把文本集中的每个文本表示成向量形式。其中特征词对应为 key，特征权重对应为 value，将文本表示为聚类模块要求的格式。

2) 聚类模块

本文在对聚类算法进行研究的同时，使用信息增益提出聚类标识方法，使得聚类呈现出的结果更易于用户理解。

◇ 聚类算法 采用基于 k-means 的聚类方法及基于 bisecting k-means 的聚类方法。具体包括 k-means，基于共享最近邻的改进 k-means，bisecting k-means 和基于共享最近邻的改进 bisecting k-means。

◇ 聚簇标识 使用信息增益方法生成聚簇描述，在第三章公式 3.6 中，

$$IG(t,c) = P(t,c)\log\frac{P(t,c)}{P(t)P(c)} + P(\bar{t},c)\log\frac{P(\bar{t},c)}{P(\bar{t})P(c)} \tag{5.1}$$

同时包含了正相关($P(t,c)\log\frac{P(t,c)}{P(t)P(c)}$)函数和负相关函数($P(\bar{t},c)\log\frac{P(\bar{t},c)}{P(\bar{t})P(c)}$),一个特征词在出现的情况下对文本的识别所起的作用要远大于该特征词在文本不出现的情况，我们利用它的正相关函数作为类中特征词标识提取：

$$IG_m(t,c)=P(t,c)\log\frac{P(t,c)}{P(t)P(c)} \tag{5.2}$$

根据每个类别的特征词的权重提取了候选词后，分别计算它们的 IGm，选取 IGm 最高的前几个特征词作为该类别的聚类标识。

### 5.2.3 评价标准

本文选择 F-measure 作为系统的评价准则，F-measure 是由 van Rijsbergen 于 1979 年提出的，是信息检索领域常用的一种系统性能测试指标。它综合考虑了信息检索中准确率(preeision)与召回率(recall)的思想。准确率和召回率反映了聚类质量的两个不同方

面，准确率考察的是精确度，而召回率考察的是完备性。

对于一个聚簇 $P_j$ 和人工分类 $C_i$。

$$\text{准确率}\quad presision(i,j)=\frac{n_{ij}}{n_j} \tag{5.3}$$

$$\text{召回率}\quad recall(i,j)=\frac{n_{ij}}{n_i} \tag{5.4}$$

$$F(i,j)=\frac{2*recall(i,j)*presision(i,j)}{recall(i,j)+presision(i,j)} \tag{5.5}$$

$$\text{最终 F 值: } FI=\sum_i \frac{n_i}{n}\max\{F(i,j)\} \tag{5.6}$$

*FI* 值越大，聚类结果质量越好。

## 5.3 系统实现

系统包括文本解析模块和聚类模块，实现了上述功能模块中除分词之外的其他模块。在前文理论研究的基础上，将改进的算法应用于本文基于 k-means 的中文文本聚类系统中。

### 5.3.1 系统主界面

文本聚类系统的主界面中，菜单包括：文本预处理和聚类分析。聚类分析里包括 k-means，bisecting k-means，改进的 k-means，改进的 bisecting k-means。上侧面板显示聚类结果，以树的形式标示。下侧面板显示当选中一个簇时该簇的聚类标识。如图 5.2 所示。

### 5.3.2 文本解析模块参数选择

在对文本进行解析的过程中，从效率考虑，将分词、去停用词处理、参数统计同时进行，这样只对文本集进行一次扫描就可以完成整个文本预处理过程。下一步根据所选特征评估函数进行特征选择。界面主要包括文档目录的选择，是否去停用词，特征评估函数的选择。特征选择可以根据需要选择文档频率，单词贡献度，基于 DF 和 TC 的改进方法，各方法在选中后可以对参数进行设置，可以使用默认的参数，也可以根据需要重新设置。如图 5.3 所示。

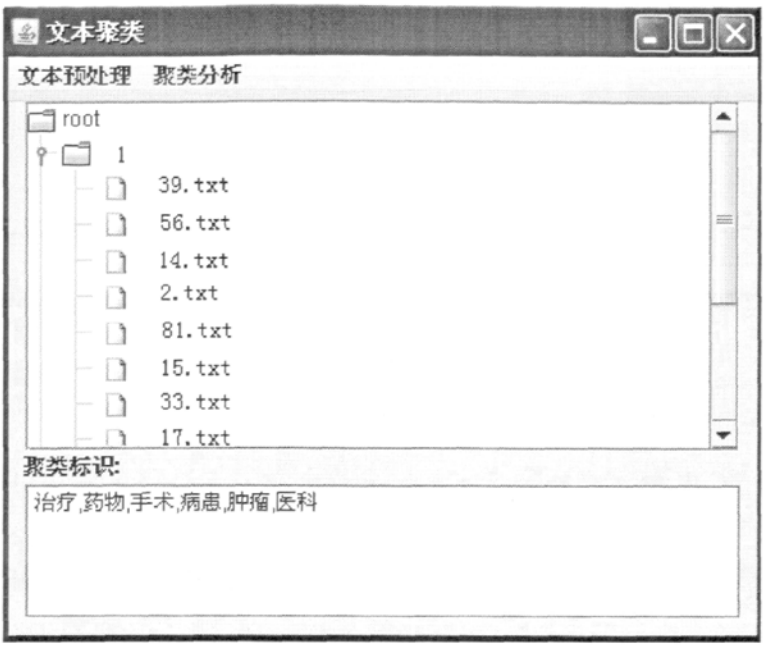


**图 5.2 系统主界面**

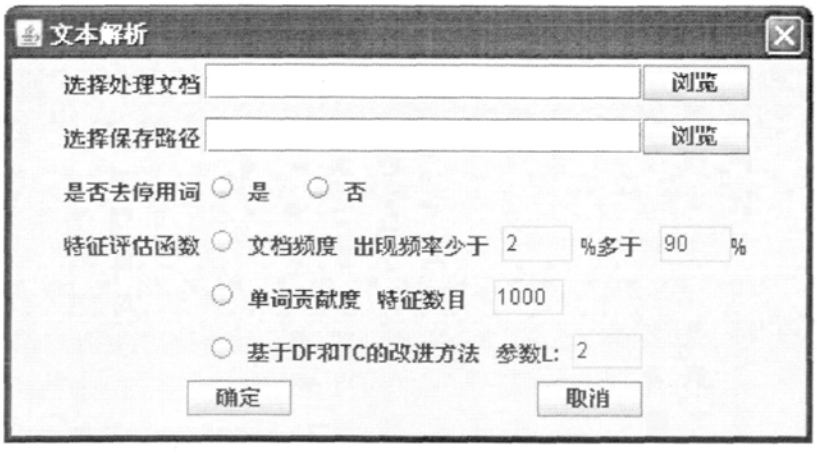


**图 5.3 文本解析模块界面**

### 5.3.3 聚类模块

聚类算法包括 k-means，改进的 k-means，bisecting k-means 和改进的 bisecting k-means。k-means 将簇划分为 k 个簇，bisecting k-means 可以将簇划分为 k 个簇，也可以对文本集生成树状结构的目录。k-means 根据 k 值是否已知分为两种，已知时输入参数 k，未知时不需要输入。各个参数可以使用默认值，也可以根据需要输入不同的值。如图 5.4 所示。

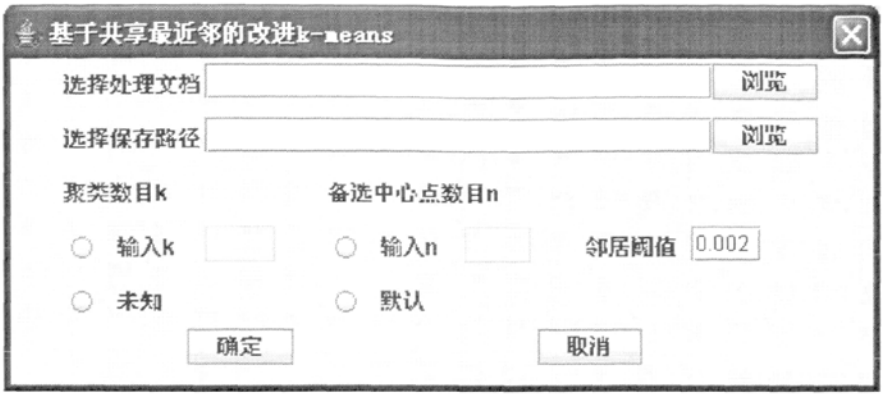


**图 5.4 基于共享最近邻的改进 k-means 聚类模块界面**

## 5.4 实验与分析

本节进行 3 组实验。实验 1 对余弦相似度和欧氏距离对 k-means 聚类结果的影响进行比较。实验 2 对 k-means 和本文的基于共享最近邻的改进 k-means 的聚类结果进行比较。验证在聚簇数 k 已知的情况下改进的初始聚类中心选择方法对聚类质量的影响。实验 3 对 bisecting k-means 和本文的基于共享最近邻的改进 bisecting k-means 的聚类结果进行比较。

### 5.4.1 实验样本集

从复旦大学提供的语料库中选择教育，计算机，政治，环境，医药 5 个类别文本进行实验。第一组样本中每个类别文本数相差不大，第二组样本中政治类明显多于其它类别。从搜狗语料库中选择计算机，旅游，医药，体育，经济 5 个类别文本组成样本三。详细情况如表 5.1——5.3 所示。

**表 5.1 样本一类别分布表**

| 类别 | 计算机 | 政治 | 教育 | 环境 | 医药 |
|---|---|---|---|---|---|
| 文本数(篇) | 132 | 156 | 146 | 134 | 136 |

**表 5.2 样本二类别分布表**

| 类别 | 计算机 | 政治 | 教育 | 环境 | 医药 |
|---|---|---|---|---|---|
| 文本数(篇) | 132 | 284 | 146 | 144 | 138 |

**表 5.3 样本三类别分布表**

| 类别 | 计算机 | 旅游 | 医药 | 体育 | 经济 |
|---|---|---|---|---|---|
| 文本数(篇) | 156 | 162 | 168 | 189 | 170 |

### 5.4.2 实验比较

实验 1：余弦相似度与欧氏距离的影响分析

在文本聚类中，余弦相似度或者欧氏距离都被用来度量文本之间的邻近度，本文研究它们对 k-means 聚类结果的影响。使用 F-measure 作为评价指标，采用样本一作为实验数据。由于 k-means 受初始聚类中心选择的影响，不同的聚类中心会得到不同的聚类结果，在实验中，将其运行 30 次，然后计算 F-measure 平均值。实验结果如表 5.4 所示。

**表 5.4 不同相似度度量方式对 k-means 聚类结果的影响**

| F-measure 区间 | 余弦相似度 | 欧氏距离 | 改进的欧氏距离 |
|---|---|---|---|
| 0.5 以下 | 3 | 10 | 6 |
| 0.5~0.6 | 8 | 12 | 12 |
| 0.6~0.7 | 12 | 8 | 9 |
| 0.7 以上 | 7 | 0 | 3 |
| 平均值 | 0.71 | 0.52 | 0.62 |

从实验结果可以看出，使用余弦相似度得到的结果优于欧氏距离。文本对象不但具有稀疏性、维灾难问题，还具有方向性的特征，余弦相似度计算文本对象之间的夹角，更适用于文本领域。文献[30]使用凝聚层次聚类算法对余弦相似度和欧氏距离对聚类质量的影响进行了比较，不同的类间相似度计算方法得到的结果不同，对于 k-means 不存在这种情况。

对于高维稀疏的向量，向量的非零项的重要性要高于值为零的项，然而，在欧氏距离的计算过程中，非零项与值为零的项的重要性没有区别，除非 2 个文本向量在该维值都为 0，否则在这一维上仍会计算属性间的距离，这样的结果使得文本间的距离差别不大。因此，欧氏距离没有很好的度量文本之间的关系。基于以上原因，我们改变欧氏距离的计算方式，当两篇文本中有一篇文本在某一维上的值为零时，在该维上不计算欧式距离，通过实验我们发现，改进后的欧氏距离得到的结果有很大提高。如表 5.4 所示。

实验 2：k-means 与基于共享最近邻的改进 k-means 的比较(在 k 值已知的情况下对两算法进行比较)

1) 参数的选择：首先对改进算法的参数进行讨论

(A)备选中心点集的文本数 $m = k + n_+$

在基于共享最近邻的改进 k-means 选择初始聚类中心的过程中，首先需要确定$n_+$的值，以选择$k+n_+$个点作为备选中心点集。$n_+$的确定非常重要，如果$n_+$的值太小，中心点的筛选被限制到比较小的备选中心点集，不能保证这些点可以很好的覆盖整个文本集的所有类别；比较大的$n_+$值可以帮助我们对备选中心点集进行二次筛选时能选择到更好的初始聚类中心，然而，如果$n_+$值过大，对于第二步的筛选来说，计算量过大。本文尝试了一系列的$n_+$值，以在聚类质量和计算量之间取得平衡，当$n_+$取 0 到 k 时，$n_+$值越大，聚类质量越好，因此，本文将$n_+$设置为 k，备选中心点集包括 2k 个文本。

(B)判断 2 文本是否为邻居的相似度阈值$\theta$

在$m$个备选初始聚类中确定两篇文本是否互为邻居时需要确定参数$\theta$，为了找到合适的参数$\theta$，我们首先尝试将$\theta$设置为文本集中所有文本对的平均相似度，然后通过在 0.01 到 0.25 之间尝试不同的值。采用样本一，样本二和样本三作为实验数据。实验结果如图 5.2 所示。

由实验结果，改进算法的聚类结果比较好。对于不同的$\theta$值，存在一定程度的波动但波动不大。在 0.06 处得到的结果较好。

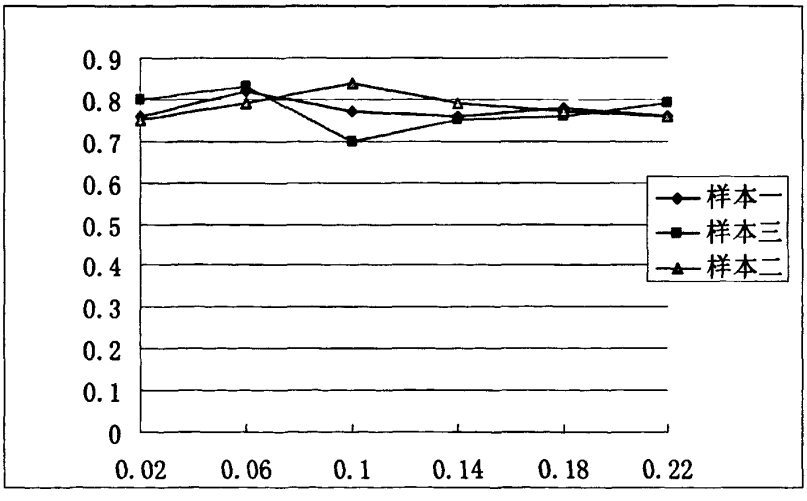


图 5.5 不同$\theta$值下聚类结果的比较

2) 实验比较：

将 k-means 运行 40 次，取平均值作为最终的 F-measure 值，将基于最大最小原则的改进 k-means 运行 40 次，取平均值作为最终的 F-measure 值，在基于共享最近邻的改进

k-means 中，将$\theta$设为 0.06，$n_+$设为 5。采用样本一，样本二，样本三作为实验语料。

**表 5.5 结果比较图**

| F-measure | k-means | 基于最大最小原则的 k-means | 基于共享最近邻的改进 k-means |
|---|---|---|---|
| 样本一 | 0.71 | 0.75 | 0.81 |
| 样本二 | 0.73 | 0.74 | 0.79 |
| 样本三 | 0.72 | 0.75 | 0.83 |

由实验结果可以看出，基于共享最近邻的改进 k-means 得到的结果优于其它两种方法。基于最大最小原则的改进 k-means 得到的结果优于 k-means。在最大最小原则方法中，由于第一个点的选择是随机选取，得到的结果也存在一定的波动性，当第一个点选择的不好时，影响后续聚类中心的选择，进而影响了算法的平均 F-measure 值。

在第二组实验中，基于最大最小原则的 k-means 得到的结果与 k-means 相差不大，这是因为样本二中各类别分布不平衡，政治类文本明显多于其他类别文本，在使用最大最小原则选择聚类中心时，在有些情况下容易将多个政治类的文本选为聚类中心，影响了所选聚类中心的代表性。基于共享最近邻的改进 k-means 由于在最大最小原则的基础上对聚类中心进行了二次筛选，得到的结果优于基于最大最小原则的 k-means。这进一步验证使用共享最近邻度量文本间的相似度时由于考虑了文本周围的情况，可以准确的度量文本间的相似度，选择到的聚类中心能更好的代表整个文本集。

实验 3：bisecting k-means 与基于共享最近邻的改进 bisecting k-means 的比较

实验中采用样本一和样本二作为实验数据。bisecting k-means 选择最大的簇进行分解，迭代次数定为 5。改进的 bisecting k-means 中$\theta$值取 0.06，迭代次数定为 5。实验结果如表 5.5 所示。

**表 5.6 bisecting k-means 与改进的 bisecting k-means 在不同样本下聚类结果比较**

| | bisecting k-means | 改进的 bisecting k-means |
|---|---|---|
| 样本一 | 0.75 | 0.77 |
| 样本二 | 0.68 | 0.74 |

通过实验结果可以看出，改进的 bisecting k-means 得到的结果优于 bisecting k-means。改进的 bisecting k-means 由于选择簇中心点邻居最少的簇进行分解，考虑到

了簇的紧密性，选择质量较差的簇进行分解，有利于提高聚类的质量。但是我们发现，改进的算法在第一组实验中对原算法改进的幅度小于第二组实验。bisecting k-means 选择文本数最大的簇进行分裂，容易得到比较平衡的分簇，得到的聚类结果中各个簇的大小相差不大，第一组实验所采用的样本分布较为平衡，因而两种方法得到的聚类结果相差不大。在第二组实验中，采用的样本各个类别的文本数量相差比较大，政治类文本的数量几乎是其它类别的两倍，在这种情况下，改进的 bisecting k-means 得到的聚类结果较原算法有了更多的提高。在样本分布不平衡时，仅仅使用文本数作为选择下一次分裂的簇的指标，样本多的簇容易被分解，容易得到错误的结果。因此，本文提出的方法对于分布不平衡的样本，可以得到更好的聚类结果。

使用邻居的概念从 m(m<k)个簇中选择一个簇的计算量很小，因此使用邻居概念选择下一次分裂的簇时，与原方法相差不大。

## 5.5 本章小节

将前文提出的改进算法应用于基于 k-means 的中文文本聚类中。通过实验对上一章提出的改进算法与原算法进行了比较，结果表明基于共享最近邻的改进的初始聚类中心选择方法选出的聚类中心比较分散且代表性好；改进的选择下一次分裂的簇的方法通过选择紧密性最差的簇，在对分布不平衡的样本进行实验时，得到了更好的聚类结果。

# 总结与展望

本文对文本聚类的主要技术：文本预处理、特征选择以及聚类算法进行了研究，主要做了如下工作。

1) 对中文文本聚类的相关技术和算法作了深入的探讨，包括：中文分词技术、文档的表示、特征选择算法、权重计算方法、相似度计算和聚类算法方面的内容做了研究。

2) 通过对已有的特征选择方法进行了研究。针对在聚类领域进行特征选择时由于不能使用类信息而很难选择到最具代表性的特征词。针对该缺点，在文档频率，单词贡献度两种特征选择方法的基础上，通过引入贪心算法的思想，提出了基于文档频率和单词贡献度的改进的特征选择方法。实验表明改进的算法可以在保证聚类质量的前提下过滤更多的特征词。

3) 针对 k-means 算法中的初始化问题，结合共享最近邻与最大最小原则对聚类中心分别进行选择，提出一个改进的初始聚类中心选择方法。实验表明改进的方法选择到初始聚类中心比较分散且代表性好。

4) 引入共享最近邻中邻居的概念对簇的紧密性进行刻画，每次选择紧密性最差的簇进行分裂。利用以上分裂簇的方法对 bisecting k-means 算法进行改进。实验结果表明改进的算法的聚类质量较原算法有一定的提高。

5) 在以上研究工作的基础上，实现了基于 k-means 的中文文本聚类原型系统。系统包括文本解析模块和聚类分析模块，文本解析模块实现了文档频率，单词贡献度以及基于文档频率和单词贡献度的改进的特征选择方法。聚类模块实现了基于共享最近邻的改进的 k-means 算法及基于共享最近邻的改进的 bisecting k-means 算法。实验表明，改进算法的使用提高了文本聚类的质量。

因为时间关系，本课题还有一些工作有待于继续深入和开展，主要有：

1) 文本数据高维稀疏的特点使得文本聚类的质量还较低，怎样选择更具有代表性的特征仍需进一步研究。

2) 本文所提出的算法复杂度较原算法有所提高，下一步将结合取样技术对改进算法进行研究，并在更大的数据集上进行实验。

3) 在现实中，文本所属的类别通常不唯一，下一步将对文本的软聚类进行研究。

由于本人水平有限，论文中不妥与错误之处在所难免，恳请各位专家、老师和同学

批评指正。

# 参考文献

[1] V. Hatzivassiloglou, J. L. Klavans, M. L. Holcombe, R. Barzilay, M.-Y. Kan, and K. R. McKeown. Simfinder: A Flexible Clustering Tool for Summarization[A]. Proc. of the Workshop on Summarization in NAACL-01[C]. Pittsburg, Pennsylvania,USA: 2001

[2] Oren Zamir, Oren Etzioni, Omid Madani, Richard M. Karp, Fast and Intuitive Clustering of Web Documents, KDD'97, 1997: 287-290.

[3] 林鸿飞, 马雅彬. 基于聚类的文本过滤模型[J]. 大连理工大学学报. 2003, 42(2): 249-252.

[4] Rauber, M.Frühwirth. Automatically Analyzing and Organizing Music Archives[A]. Proceedings of the 5th. European Conference on Research and Advanced Technology for Digital Libraries ( ECDL 2001 )[C]. Darmstadt. Germany, 2001: 402-414

[5] Cutting, D., Karger, D., and etc. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections [A]. SIGIR'92, 1992 [C]. 318 -329 .

[6] James Allan,Topic Detection and Tracking: Event-based Information Organization[M], Kluwer Academic Publishers, 2002: 1-16

[7] Daphe Koller and Mehran Sahami.Hierarchically classifying documents using veryfew words,Proceedings of the 14th International Conference on Machine Learning(ML), Nashville, Tennessee,July 1997: 170-178.

[8] Zheng Chen, WeiYing Ma, Jinwen Ma. Learning to Cluster Web Search Results[A]Proceedings of the 27th Annual International ACM SIGIR Conference [C]. Sheffield, South Yorkshire, UK, July 2004: 210 -217

[9] 苏芳仲.中文 Web 文本挖掘的若干关键技术研究及其实现[D]. 福州大学硕士学位论文, 2006.

[10]MacQueen J. Some methods for classification and analysis of multivariate observations[A]. proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability.[C] .1967: 281-297

[11]Huang Z. Extensions to the k-means algorithm for clustering large data sets with categorical values[J]. Data Mining and Knowledge Diseovery, 1998, 2(3): 283-304

[12]T Zhang, R Ramakrishnan, Mogihara.Birch:An effieient data clustering method for very

large databases. In Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data Montreal.Canada:June, 1996: 103-114

[13]Viswanath P,Pinkesh R. I-DBSCAN:a fast hybrid density based clustering method. 18th International Conference on Pattern Recognition, Hong Kong, 2006: 912-915

[14]Kouae L .Using self-organizing maps for information visualization and knowledge discovery in complex geospatial database.21th International Cartographic Conference. Durban. 2003: 1674-1701

[15]Bezdek J C. Pattern recognition with fuzzy objective function algorithms[M]. New York: Plenum Press 1981

[16]Steinbach M, KaryPis G, Kumar V . A comparison of document clustering techniques[A]. Proceedings of KDD 2000 Workshop on Text Mining[C],2000: 1-20

[17]Ying Zhao. KaryPis G. Hierarchical Clustering Algorithms for Document Datasets. Proceedings of Data Mining and Knowledge Discovery[C], 2005 10(2): 141-168.

[18]Dhillon I S, Modha D S. Concept decompositions for large sparse text data usingclustering[J]. Machine Learning, 2001, 42(1): 143-175

[19]Banerjee A,Dhillon I S, Ghosh J,S.Sra. Clustering on the Unit Hypersphere using Von Mises-Fisher Distributions Journal of Machine Learning Research (JMLR) 2005(6): 1345~1382.

[20]Beil F, Ester M, Xu X. Frequent term-based text clustering.In:Proc.8th Int. Conf. on Knowledge Discovery and Data Mining(KDD)' 2002[C], Edmonton. Alberta, Canada , 2002: 436-442

[21]Fung B C M , Wang K, Ester M. Hierarchical Document Clustering Using Frequent Itemsets. In: Proc. of the 2003 SIAM Intl. Conf. on Data Mining (SIAM'03)

[22]Hassan H . Malik and john R . Kender . High Quality, Efficient Hierarchical Document Clustering using Closed Interesting Itemsets Proceedings of the Sixth International Conference on Data Mining(ICDM'06) 2006

[23]龙昊, 冯剑琳, 李曲. R-means:以关联规则为簇中心的文本聚类[J], 计算机科学, 2005(9): 156-159

[24]Zhuang L, Dai Honghua. A Maximal Frequent Itemset Approach for Web Document Clustering[A] Proc of the 4th Int'l Conf on Computer and Information

Technology[C] , 2004: 970-977.

[25]顾益军，樊孝忠，王建华等. 中文停用词表的自动选取[J]. 北京理工大学学报, 2005, 25(4): 337-340.

[26]庞剑锋，卜东波，白硕. 基于向量空间模型的文本自动分类系统的研究与实[J].计算机应用研究， 2001, 18(9): 23-26.

[27]Gerald Salton, Wong A. , and Yang, C. S . A vector space model for automaticindexing[J]. Comm. ACM, 1975, 18(11): 613-620.

[28]M. Easter, H. P. Kriegel, J. Snade，X. Xu. A density-based algorithm for diseovering Clustering large spatial databases[A]. In proc. 1996 Int. Conf. Knowledge Dlseovery And Data Mining(KDD'96)[C]，226-231，Portland，OR，Aug.1996

[29]张猛,王大玲,于戈. 一种基于自动阈值发现的文本聚类算法，计算机研究与发展, 2004, 41(10): 1745-1753.

[30]周昭涛. 文本聚类分析效果评价及文本表示研究[D], 中国科学院研究生院硕士学位论文, 2005

[31]K. Van Rijsbergen. Information Retrieval[M]. Butterworths, London. 1979.

[32]F. Beil，M. Este, X. Xu. Frequent term-based text clustenng[J]. In Porc. 2002 Int. Conf knowledge Diseovery and DataMining(KDD'02). NewYork, 2002: 436-442

[33]Iwayama Makoto and Tokunaga Takenobu. Hierarchical Bayesian clustering for automatic text classification, Department of Computer Science Tokyo Institute of Technology, Tech Rep: TR95-0015, 1995.

[34]Y. Yang,O.Pedersen. A comparative study on feature selection in text categorization The ICML97[C], Nashville, 1997

[35]M. Rogati, Y. Yang. High performance feature selection for text categorization. The CIKM-02. Mclean. 2002

[36]L. Tao，L. Shengping, C. Zheng. et al. An evaluation on feature selection for text clustering. Proceedings of the twentieth international conference on machine learning The ICML-03, Washington DC, 2003

[37]姜宁，宫秀军，史忠植. 高维特征空间中文本聚类研究[J] 计算机工程与应用 2002, 10

[38]Wilbur,I.W.,&Sirotkin,K.The automatic identification of stop words[J]. Journal of

Information Science, 1992, 18(1): 45-55.

[39]Dash, M., & Liu, H. Feature Selection for Clustering[A]. Proc.of PAKDD[C], 2000: 110-121

[40]L. Liu, et al. A Comparative Study on Unsupervised Feature Selection Methods for Text Clustering[A]. Proceeding of Natural Language Processing and Knowledge Engineering' 05[C], October, November 2005: 597-601

[41]徐燕, 李锦涛, 王斌, 孙春明. 基于区分类别能力的高性能特征选择方法[J]. 软件学报, 2008, 1(1): 82-89.

[42]Nirmalie Wiratunga, Rob Lothian, and Stewart Massie. Unsupervised Feature Selection for Text Data[C]. In Proceedings of the 8th European Coference on Case-Based Reasoning, 340-354

[43]R. A. Jarvis and E. A. Patrick. Clustering Using a Similarity Measure Based on Shared Nearest Neighbors[J]. IEEE Transactions on Computers. 1973, C-22(11): 1025-1034

[44]Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. ROCK: A Robust Clustering Algorithm for Categorical Attributes. In Proc. of the 15th Int . conf . on Data Engineering[J]. IEEE Computer Society,March 1999 : 512-521

[45]Brandley P. S. , Fayyad U. M. Refining initial points for K-Means Clustering Proc. of 15th International Conferene on Maehine Learning, SanFraneiseo, 1998: 91-99.

[46]周涓，熊忠阳，张玉芳等. 基于最大最小距离法的多中心聚类算法. 计算机应用, 2006，26(6): 1425-1427

[47]任江涛, 施潇潇, 孙婧昊, 黄焕宇, 印鉴.一种改进的基于特征赋权的 k 均值聚类算法[J]. 计算机科学, 2006,33(7):186-187

[48]张猛. 文本聚类中参数自动设置技术的研究与实现[D]. 东北大学学位论文, 2005

[49]刘远超, 王晓龙, 刘秉权. 一种改进的k-means文档聚类初值选择算法[J]. 高技术通讯, 2006(1): 11-15

[50]李金宗. 模式识别导论. 北京. 高等教育出版社. 1994:294-356.

[51]Vesanto J，Alhoniemi E. Clustering of the self-organizing map. IEEE Transactions on Neural Networks, 2000, 11(3): 586-600

# 攻读学位期间发表的论文

[1]. 张睿，刘晓霞.《基于URN的特征冲突过滤方法的研究》.计算机工程.已录用.

# 致 谢

时光如梭，在毕业论文完成之际，不尽万分感慨。首先感谢我的导师刘晓霞教授，三年来，导师对于我的学习，工作，思想和生活给予了无微不至的关怀，刘老师渊博的知识、严谨求实的治学态度、平易近人的性格，让我获益颇多，从刘老师这儿我所学到的不仅仅是知识，还有堂堂正正做人的道理和从容面对困难的心态，更通过言传身教，培养了我严谨、求实的学习态度。本论文是在刘老师的指导下完成的，在论文的写作过程中刘老师给予了我很大的帮助。在论文的选题、撰写过程中，刘老师都给了我极大的关注和指导，论文的完善和定稿刘老师更是付出了很大的心血，在她及时的督促、认真的指导下才使我的论文在最后顺利的完成。

其次，我要感谢实验室的王平静，李亚军，李洪安，孙华斌等同学。在将近三年的学习期间内，她们给了我很大的帮助，使我在专业理论知识、实际操作技术上有了很大的进步。感谢我的舍友徐湘、马伟荣、陈麓屹，她们让我懂得了如何生活，如何调节自己，也因为有了他们，使我拥有了快乐、充实的研究生生活。我还要感谢我的父母和所有关心我的亲人和朋友，没有他们对我的关怀和支持，论文就不可能顺利完成。

请允许我再一次感谢所有关心、鼓励和支持我的老师、同学、朋友和亲人！他们的激励，是我在人生路上不断进取的动力。

# 基于k-means的中文文本聚类算法的研究与实现



作者： 张睿
学位授予单位： 西北大学
被引用次数： 2次

## 本文读者也读过(3条)

1. 孙爽 基于语义相似度的文本聚类算法的研究[学位论文]2007
2. 王钦平 基于改进K-means算法的Web文档聚类系统的研究与实现[学位论文]2007
3. 吴启纲 中文文本聚类算法的研究与实现[学位论文]2010

## 引证文献(2条)

1. 罗晖霞. 曲晓玲 基于网络舆情的K-Means算法的改进研究[期刊论文]-电脑开发与应用 2010(8)
2. 汤寒青. 王汉军 改进的K-means算法在网络舆情分析中的应用[期刊论文]-计算机系统应用 2011(3)