第 36 卷第 12 期
2008 年 12 月
同济大学学报(自然科学版)
JOURNAL OF TONGJI UNIVERSITY(NATURAL SCIENCE)
Vol. 36 No. 12
Dec. 2008

# 一种提高文本聚类算法质量的方法

冯少荣

(厦门大学 信息科学与技术学院, 福建 厦门 361005)

**摘要**:针对基于 VSM(vector space model)的文本聚类算法存在的主要问题,即忽略了词之间的语义信息、忽略了各维度之间的联系而导致文本的相似度计算不够精确,提出基于语义距离计算文档间相似度及两阶段聚类方案来提高文本聚类算法的质量. 首先,从语义上分析文档,采用最近邻算法进行第一次聚类;其次,根据相似度权重,对类特征词进行优胜劣汰;然后进行类合并;最后,进行第二次聚类,解决最近邻算法对输入次序敏感的问题. 实验结果表明,提出的方法在聚类精度和召回率上均有显著的提高,较好解决了基于 VSM 的文本聚类算法存在的问题.

**关键词**: 文本聚类; 语义距离; 最近邻聚类; 相似度; 聚类算法
**中图分类号**: TP 312　　**文献标识码**: A　　**文章编号**: 0253 - 374X(2008)12 - 1712 - 07

# A Method to Improve Text Clustering Algorithm Quality

*FENG Shaorong*

(School of Information Science and Technology, Xiamen University, Xiamen 361005, China)

Abstract: The main problem with the text clustering algorithm based on vector space model (VSM) is that semantic information between words and the link between the various dimensions are overlooked, resulting in inaccuracy in the text similarity calculation. A method based on computing the text similarity using semantic distance and two-phrase clustering is proposed to improve the text clustering algorithm. First, the text analyzed according to its semantic, with nearest neighbor algorithm used for the first cluster. Some feature words are chosen according to the similarity weight to represent the cluster with the remaining feature words similar to the main themes of the cluster, and then class combination is carried out. Finally, the second clustering is carried out to improve the nearest neighbor clustering which is sensitive to the input order of the document. Simulation experiments indicate that the proposed algorithm can solve these problems and performs better than the text clustering algorithm based on VSM in the clustering precision and recall rate.

Key words: text clustering; semantic distance; nearest neighbor clustering; similarity; clustering algorithm

文本聚类是一种典型的无指导的机器学习问题. 它将一个文档集分成若干称为集簇的子集,每个集簇的成员之间有较大的相似性,而集簇之间文档具有较小的相似性. 与分类方法有所不同,聚类没有

**收稿日期**: 2007 - 07 - 06
**基金项目**: 国家自然科学基金资助项目(50474033)
**作者简介**: 冯少荣(1964—),男,副教授,工学博士,主要研究方向为并行分布数据库、数据仓库、数据挖掘. E-mail: shaorong@xmu.edu.cn

预先定义好主题类别标记,需要由聚类学习算法来自动确定.

目前,人们已提出很多文本聚类算法[1-2],大部分基于 VSM(vector space model)来聚类,该模型假设词语间是独立的,并没有从语义上去分析文档内容,因而不能准确计算文档间的相似度,影响了聚类的精度. 笔者着重从语义[3-5]上分析文本,先根据词语频率 TF(term frequency)和词语倒排文档频率 IDF(inverse document frequency)选择文档的关键词,将文档表示为一组关键词集合,再利用语义距离计算关键词间的相似度,通过相似关键词对数目最后衡量文本间的相似度,这样真正从语义上具体分析文本之间的相似值,使结果更接近人的主观衡量.

## 1　语义距离计算的基本思想

### 1.1　改进的相似度计算方法

将文档间语义距离具体转化为词语间语义距离、义原间语义距离. 达到利用语义距离计算文档间相似度的目的. 计算以《知网》[6-7]作为语义的本体,通过对《知网》的数据、结构、知识描述语言以及文献[7]的分析、研究,提出改进的相似度计算方法.

#### 1.1.1　改进的词语相似度计算方法

2 个词 $W_1$, $W_2$的相似度 Sim($W_1$, $W_2$)具体计算步骤如下:

(1) 若 2 个词都只有 1 个义项,则根据文献[7]中式(1)计算 $W_1$, $W_2$第一义原的相似度 Sim($W_1$, $W_2$);若有词有 2 个以上的义项,按照文献[7]中式(2)判断哪个义项为义原描述符,并计算 Sim($W_1$, $W_2$).

(2) 若 $W_1$有义原与 $W_2$同或 $W_2$有义原与 $W_1$同,且这个相似的义原不是弱义原,则 Sim($W_1$, $W_2$) = 1,即 2 个词同义.

(3) 若存在 $W_1$只有 1 个基本义原, $W_2$有 2 个以上义原,则继续判断 $W_1$基本义原是否和 $W_2$的其他义原相等且不是弱义原,若是,则最后 2 个词的相似度为 Sim($W_1$, $W_2$) = 0.8,值 0.8 说明 2 个词是相关的;否则 2 个词相似度为它们的第一义原相似度值.

(4) 若 $W_1$, $W_2$都有 2 个以上义原,则两两比较 2 个词其他的义原,若有 2 个义原相等且不是弱义原,则 2 个词的相似度 Sim($W_1$, $W_2$) = 0.8,结束计算;若没有义原对相等或者相等但为弱义原,则按照文献[7]中式(4)计算 2 个词的相似度,若是弱义原配对,赋予较低权重.

有许多词汇的第一义原往往是很抽象的义原,而且与此相同的第一义原重复很多,对获取这些词汇的语义信息帮助不大,所以通过比较 2 个词的非弱义原是否相等来判断 2 个词是否相关,进而计算 2 个义原距离,可以大幅提高计算义原距离的效率.

#### 1.1.2　改进的文档相似度计算方法

若 2 个词相似度值大于等于 0.8,则 2 个词是相关的. 文档相似度计算步骤如下:

(1) 取出文档 1 中的 1 个关键词.

(2) 计算与文档 2 中所有未匹配关键词两两之间的相似度,选择最大一个记为最匹配相似度,对应的词为最匹配词.

(3) 若最匹配相似度值大于 0.8,说明 2 个词语相关,相关词对数量加 1,并置文档 2 对应匹配词已匹配.

(4) 重复步骤(1),直到文档 1 所有关键词扫描完毕.

(5) 若相关词对数量超过规定值 2,则说明 2 篇文档相似,属于同类.

步骤(3)规定最匹配相似度值大于 0.8,这样能更有效地发现尽量多的相关词对. 因为若 2 个词相似度值太小,说明 2 个词毫无关系,此时若置词已经匹配,则后面的词无法继续和这个词比较,但是后面词有可能和这个词是相关的.

若取有 2 对相关词对,则 2 篇文档相似,但这只能保证大部分文档相似. 有些文档间相关词对有 2 个以上且这 2 词不是主题词,可以这样处理:文档和当前所有类比较,找最相似的类,这个最相似的类表现在它所有的关键词两两匹配后的相似度加权之和最大,而且和这个最相似的类有 2 对以上相关词对才能说明 2 篇文档相似,否则不相似. 这样计算文档相似度既用到了所有词的加权相似度,又考虑到词的相关词对数量,更能准确说明 2 篇文档的相似与否.

### 1.2　相关概念

(1) 文档列表(ArtistList):主要用于存放文档.

(2) 文档关键词数组(Dword[rowcount][colcount]):rowcount 为文档维, colcount 为关键词维.

(3) 预类特征词矢量(VPreCenter):词语两两相似计算后先放入预类特征词矢量中,经过筛选后将相似权重较大词放入类特征词矢量中.

(4) 预类特征词相似权重(VPreSimCenter):每个预类特征词对应的相似权重值.

(5) 类特征词矢量(VVCenter):每个类的特征

描述词,这是个矢量二维,一维代表类,二维为每个类的特征词矢量.

(6) 类特征词相似权重(VVSimCenter):每个特征词对应的相似权重值.

(7) 待聚类文档矢量(WaitCluster):需要重新聚类的文档.

(8) SumSim: 文档与某类的总相似度值,为全局变量.

### 1.3 判定文档与类相似的方法

使用 SumSim 值大小来判断文档是否与类相似,它的值是文档关键词与类特征词两两匹配后相似度值之和. SumSim 值越大,说明文档与类越相似. 但是不等于文档属于使 SumSim 最大那个类,按照 1.1.2 节文档相似度计算方法,至少有 2 对相关词对,文档才能相似,所以在找到 SumSim 最大类后还要再判断是否这个类与文档有 2 个相关词对,若有,文档属于该类,否则,文档与当前所有类相似度都很低,需要新增一类. 通过 SumSim 变量可以找到与文档最相似的类,减少一些非主题相关词对的干扰,更好判断文档与类的相似性,加强类内的紧凑性.

### 1.4 文档与类特征词的相似度权重计算方法

(1) 取出文档 I 的关键词 $j$,Dword[I][$j$]; CountMax = 0;

(2) 按照 1.1.1 节计算方法计算关键词 $j$ 与类 $K$ 未匹配特征词的相似度,根据这个相似度值两两比较找出与关键词最匹配的类特征词,MaxSim 记为它们的相似度值. 最后文档与类的总相似度值 SumSim = SumSim + MaxSim.

(3) 若 MaxSim ≥0.8, CountMax = CountMax +1;

若 2 个词不相等,说明它们相关,则将它们存入矢量 VPreCenter 中,并将它们的相似度加上类特征词的相似权重,存入相应的预类特征词相似权重 VPreSimCenter 中,并将该类特征词标记为已匹配.

若 2 个词相等,则取 1 个词存入矢量 VPreCenter 中,类特征词的相似权重加上 1,存入相应的 VPreSimCenter 中,并将该类特征词标记为已匹配.

若 MaxSim < 0.8 且 MaxSim ≥0.5,说明 2 个词基本义原相同,则 2 个词存入矢量 VPreCenter 中,文档关键词相似权重为相似度值,类特征词新的相似权重为类特征词的相似度加上类特征词的相似权重,存入相应的 VPreSimCenter 中.

若 MaxSim < 0.5, 说明 2 个词毫无关系,则文档关键词不作为候选词加入预类特征词矢量中,而类特征词则保留,加入预类特征词矢量中,其相似权重不变.

(4) 重复执行步骤(1),直到文档 I 所有关键词扫描完毕,返回 CountMax 值.

步骤(3)中的 4 个假设主要根据词对相似度大小来相应更新预类特征词权重,经过这样条件筛选后,逼近主题的类特征词权重不断增加,而干扰词权重不断减少,直至最后被淘汰.

### 1.5 更新类中心

更新类中心(UpdateCenter(k))不能同数据聚类中那样按照均值大小来更新,因为文档和类之间,词没有一一对应,而类中心的特征词选择又必须尽量逼近主题,所以,根据上述相似权重的概念,按照相似权重大小来选择. 初始时,对选入的每个类特征词都置权重为 1,当有新文档加入类,按照 1.1.1 节中提出的词语相似度计算方法计算文档中关键词和类中特征词的相似度,再根据相似情况按照 1.4 节方法更新相应相似权重. 然后,通过如下 3 步骤完成类中心的更新.

(1) 删除 VVcenter 和 VVSimCenter 所有元素,准备加入新的特征词和权重.

(2) 按照相似权重 VPreSimCenter 排序预类特征词矢量 VPreCenter.

(3) 将权重较大的侯选词存入 VpreSimCenter 中,同时处理 VVSimCenter. 若侯选词已在类中心中,则不加入. 最后,根据相似权重选择权重较大的前 6 个词,而淘汰权重较小的候选词.

以上处理过程中尽量加大了相关词权重,因为这类词描述主题概率最大,也即尽量减少了相似度小的词的权重. 随着同类内相似的文档越来越多,描述主题的特征词的相似权重会越来越大,因为文档与类的相似主要体现在描述主题词的相关上,所以这类主题词会越来越多,它们对应相似权重值也会越来越大,而与主题无关词可能开始时被选作类特征词,但会随着类内文档增多相似权重逐渐减少而被淘汰. 这样根据权重优胜劣汰,可以不断选出有效类特征词来逼近主题.

### 1.6 初始化类中心

(1) 取当前文档的每个关键词.

(2) 依次加入类特征矢量 VVCenter 第 $K$ 个矢量中,作为第 $K$ 类的类特征词.

(3) 初始化相似权重矢量 VVSimCenter,并将第 $K$ 个矢量中每个对应的特征词权重赋值为 1.

## 2　基于语义距离的文本聚类算法

基于语义距离的文本聚类算法用到最近邻聚类算法. 首先将第一篇文档作为第一类,然后依次扫描剩下文档,若文档与当前类最相似,则属于当前类,并更新类中心; 否则增加一类,新类中心为当前文档. 而判断文档与类是否相似可利用 1.3 节中的计算方法.

### 2.1　算法流程

算法流程为: ①初始化第一类中心; ②第一次聚类; ③整理类; ④第二次聚类.

### 2.2　相应算法的形式化说明

#### 2.2.1　初始化第一类中心

```
{ obtain first documentin in ArtistList;
    K = 1;      // K: 当前类总数
    InitCenter(K);}
```

#### 2.2.2　第一次聚类

第一次聚类基于最近邻聚类算法.

```
FirstCluster(ArtistList, W, WaitCluster)//W: 聚类产生的分类集合; K:当前类总数.
{ WaitCluste = {}; W = {初始化第一类}; K = 1;
  For each document i in ArtistList Do
    { Max = 0; Flag = 1; SumSim = 0;
      For each class k in W Do
      { 调用函数 ComputeSim(i, k);
        计算使得 SumSim 值最大的类 t 与 i 相关词对 CountMax;
        If CountMax >= 2 Then {i 加入 t; UpdateCenter(t); }
        Else { K = K + 1; InitCenter(K); 新增一类加入 W 中}
  } }
For each class j in W Do    //m:j 中包含的文档数
{ If m = 1 then j 中文档加入 WaitCluste 中; } }
```

#### 2.2.3　整理类

```
CleanCluster(W) //当前类总数为 K,类集合为 W.
{ For each class i and j in W Do    //Size(i): 类 i 内文档数量
  {If Size(i) > 3 and Size(j) > 3 and VVSimCenter(i) > 0.8 and
    VVSimCenter(j) > 0.8
  Then { 合并 i 和 j 为 m; UpdateCenter(m); }
  Else { Ifi 和 j 相关词对数 > Size/2 Then {合并 i 和 j 为 m;
      UpdateCenter(m); } }
} }
```

#### 2.2.4　第二次聚类

```
SecondCluster(WaitCluster, VVCenter, W, K)
{W = {第一次聚类产生的类集合}; K = count(W);
  For each document i in WaitCluster Do
  { Max = 0; Flag = 0;
    For each class k in VVCenter Do
    { ComputeSim(i, k);
      If CountMax >= 2
      Then { i 加入 k 中; UpdateCenter(k); Flag = 1; exit;}
      //CountMax: 相似度大于 0.8 的相似词对数量
      If Flag = 0
      Then { K = K + 1; InitCenter(K); 新增一类加入 W 中;
        exit;}
      }
    }
}
```

## 3　实验结果及评价

从中国科学院的中文自然语言处理开放平台[8] CNLP 网站上下载 100 篇文档作为测试数据,根据语料主题分为 10 类,即军事(10 篇)、体育(6 篇)、政治(14 篇)、环境(9 篇)、交通(10 篇)、艺术(10 篇)、医药(11 篇)、经济(10 篇)、教育(10 篇)、经济(10 篇).

### 3.1　基于 VSM 的 K-Means 聚类算法[1-2]实验结果

文本聚类前需将文本表示为文本向量的形式. 根据 TF(term frequency) & IDF(inverse document frequency)计算每个特征词权重,并存入数据库.

每篇文章选用了 10 个高频词作为特征项,文档采用文本向量模型,对样本数据进行聚类,取聚类数为 15,多于实际的分类数 10,这样能使相似度大的几篇文档汇聚一类,而相似度小的文档归入类中或者单独成类,这样可以在一定程度上提高聚类精度,最后簇的类别标识主要提取权重较大词,从而实现自动类别标识,得到表 1 结果.

表1 基于VSM的K-Means算法聚类结果

Tab.1 K-Means clustering results based on vector space model

| 类别 | 类标识 | 文档数量 |
|---|---|---|
| 1 | 卡,公司,欧亚大陆桥,学生,中学生,森林,开幕式,医院,厂,彩虹,音乐会,委员长,群众 | 13 |
| 2 | 巡航导弹,自行车,人口,土壤,艾滋病,糖尿病 | 6 |
| 3 | 军队,议院,亚运会,手机,欧亚大陆桥,学员,学生,校舍,企业,信息,经济,锗,工会,关系 | 14 |
| 4 | 家长,石油,工会 | 3 |
| 5 | 战争,总公司,航运,台胞,角膜,重点,邮票 | 8 |
| 6 | 战争,铁路,旋风,歌舞团 | 5 |
| 7 | 生物,家长,艾滋病,马戏团 | 4 |
| 8 | 政府,诗,足球队,大学生,会计师,纪录片,会议 | 7 |
| 9 | 游击队,微机,信息,策略,航道,会县,垃圾,木材 | 8 |
| 10 | 列策,藏医 | 2 |
| 11 | 废物,林业,胃癌 | 3 |
| 12 | 核,选手,端口,间谍,课程,降雨量,企业,精子,粉,委员长,国际 | 11 |
| 13 | 年画,画廊 | 2 |
| 14 | 超级大国,汽车,旋风,旋律,国家,群众,总统 | 7 |
| 15 | 彩带,体育,电脑,森林,会议,工会,条约 | 7 |

## 3.2 基于语义距离的文本聚类算法实验结果

每篇文章选用了10个关键词作为特征项,文档采用关键词集合表示,类中心用6个特征词表示,不用预先指定类数,经过第一轮和第二轮的聚类,结果如表2、表3.

表2 第一轮聚类结果

Tab.2 First round clustering results

| 类别 | 类标识 | 文档数量 |
|---|---|---|
| 1 | 大战,战争,国,议院,大会,党员 | 3 |
| 2 | 核武器,核弹头,巡航导弹,核导弹,军备,武器 | 4 |
| 3 | 美军,军队,侵略者,军事,战区,军 | 4 |
| 4 | 亚运会,记录,选拔赛,冠军,参赛者,锦标赛 | 4 |
| 5 | 亚运会,体育,体育馆,组委会,老人,彩带 | 2 |
| 6 | 小学,中学,学院,校舍,校园,体育 | 5 |
| 7 | 电脑,网络,键盘,计算机,终端,接口 | 5 |
| 8 | 硬盘,计算机,电脑,微机,权威,市场 | 2 |
| 9 | 相声,摄像机,小品,纪录片,神,文化 | 3 |
| 10 | 公司,芯片,网路,网络,触角,风险 | 2 |
| 11 | 经济,外商,成本,银行,商务,商人 | 8 |
| 12 | 航运,业务,工程,两岸,港口,江水 | 3 |
| 13 | 地区,区域,流域,桥,大动脉,欧亚大陆桥 | 3 |
| 14 | 小费,公司,交通,列车,汽车,车 | 3 |
| 15 | 癌,艾滋病,近视眼,秘方,患者,医院 | 7 |
| 16 | 学生,小学,中学生,中学,师范大学,课 | 4 |
| 17 | 学校,思想,学员,思潮,资产阶级,党委 | 2 |
| 18 | 问题,环境,世界,人口,和平,协会 | 2 |
| 19 | 森林,林区,林业,生态学家,生态,效益 | 2 |
| 20 | 旋风,牲畜,家畜,灾民,突击队,直升机 | 2 |
| 21 | 佳作,报告,首都,青年,锗,照片 | 2 |
| 22 | 小调,歌舞团,冠军,观众,马戏团,剧场 | 4 |
| 23 | 国际,国家,文艺,团长,共产党,会议 | 4 |
| 24 | 议会,议长,团长,委员,市长,总理 | 6 |

表3 第二轮聚类结果

Tab.3 Second round clustering results

| 类别 | 类标识 | 文档数量 |
|---|---|---|
| 1 | 大战,战争,国,议院,大会,党员 | 3 |
| 2 | 核武器,核弹头,巡航导弹,核导弹,军备,武器 | 4 |
| 3 | 美军,军队,侵略者,军事,战区,军 | 4 |
| 4 | 亚运会,体育,选拔赛,体育馆,参赛者,锦标赛 | 6 |
| 5 | 小学,中学,学院,校舍,校园,学生 | 10 |
| 6 | 电脑,键盘,计算机,终端 | 7 |
| 7 | 相声,小品,纪录片,神,文化 | 5 |
| 8 | 公司,芯片,网路,网络,触角,风险 | 2 |
| 9 | 经济,外商,成本,银行,商务,商人 | 12 |
| 10 | 航运,业务,工程,两岸,港口,江水 | 3 |
| 11 | 地区,区域,流域,桥,大动脉,欧亚大陆桥 | 4 |
| 12 | 小费,公司,交通,列车,汽车,车 | 3 |
| 13 | 癌,艾滋病,近视眼,秘方,患者,医院 | 9 |
| 14 | 问题,环境,世界,人口,和平,协会 | 2 |
| 15 | 森林,林区,林业,生态学家,生态,效益 | 2 |
| 16 | 旋风,牲畜,家畜,灾民,突击队,直升机 | 2 |
| 17 | 佳作,报告,首都,青年,锗,照片 | 2 |
| 18 | 小调,歌舞团,观众,冠军,音乐家,音乐会 | 5 |
| 19 | 议会,议长,国际,国家,团长,委员 | 12 |
| 20 | 土壤,渗透法,污染物,液体,粒子,负电荷,理工学院,科学家,土,原理 | 1 |
| 21 | 精子,山羊,异体,胚胎,细胞,通信员,首例,农业,大学,家畜 | 1 |
| 22 | 废物,承运人,货,规定,机构,人,规章,代理人,舱,国家环保局 | 1 |

对比表1、表2、表3中的聚类结果,得出基于语义的文本聚类结果明显较好,且第二轮聚类结果好于第一轮.

## 3.3 性能比较

在文本聚类中将查全率(又称聚类精度)和查准率(又称聚类召回率)2个指标结合起来评价聚类结果.聚类精度反映了将相似文本单元和不相似文本单元合并到同一类的程度,反映了对不同主题的区分能力,聚类精度越高,每个类中的内容越集中.聚类召回率反映了将同一主题相似文本单元集合合并到一个类中的程度,反映了对相同主题的识别能力,聚类召回率越高,相似的文本单元越集中,即被拆分到不同类中的情况就越少.一个聚类 $j$ 及与此相关的分类 $i$ 的聚类精度 $P$ 和聚类召回率 $R$ 定义为

$$P = \frac{|N_{ij}|}{|N_j|}, \quad R = \frac{|N_{ij}|}{|N_i|}$$

式中: $N_{ij}$ 表示聚类 $j$ 中分类 $i$ 的文档数目; $N_j$ 表示聚类 $j$ 中所有对象数目; $N_i$ 表示分类 $i$ 中所有对象数目.

由于基于VSM的K-Means聚类算法[5]同主题文档在每次聚类后不一定能全部聚在一个类中,可能有几篇文档分散在其他类里,故本文采取计算

一个主题的主类的聚类精度和聚类召回率的方法. 主类即为一个主题的大部分文档所在的类. 基于语义距离文档聚类尽管将有些类大主题细分为小主题,但是在计算聚类精度和召回率时,还是将它们归到大主题计算,这样更方便和 K-Means 算法比较.

图 1 给出了基于 VSM 的 K-Means 算法和基于语义的文本聚类算法性能比较,可见,采用基于语义文本聚类方法无论是聚类精度还是聚类召回率上,在每个主题下都远高于基于 VSM 模型的 K-Means 聚类算法,尤其是体育、教育、军事等主题聚类精度和召回率都达到 90 %以上,具有良好的聚类效果.

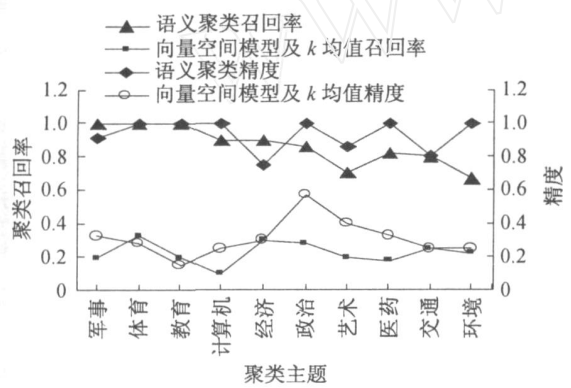


**图 1　2 种聚类算法的性能比较**

**Fig. 1　Comparison of performance for two clustering algorithms**

### 3.4　评价说明

基于语义距离的文本聚类算法将对文本的分析具体到文本的内容中,从每篇文档中抽取最重要的 10 个词代表文档,然后再从这 10 个关键词计算文本的相似度. 而关键词间的相似度以《知网》作为语义的本体,将相似度计算转为词间的距离计算,从而量化了关键词间的相似度. 这样处理后,文本间相似度计算可通过 2 个关键词集合相似词对数目和两两配对加权求和值来判断,该方法比基于 VSM 模型的余弦求值更加精确地判断 2 个文本相似与否,为文本聚类的成功提供了重要的一步.

文本聚类还有一个难点是如何描述更新类中心,基于 VSM 的 K-Means 聚类算法采取的是用类中权重较大几个词描述,但是这样类特征词无法精确描述类的主题. 而基于语义聚类算法不存在这个难题,因为它是基于语义分析文档,它可以分析出怎样的词才是主题特征词. 它引入相似权重的概念,以词语间的相似度作为权重计算基础,选择权重较大的词代表类,这样随着类内文档增多,那些体现类特征词的候选词权重会越来越大,并最终被选择,而那些无法代表类特征词的候选词也会渐渐被淘汰. 最后每个类特征词都可由一些体现主题的候选词来代表,实现了文本聚类的目的.

目前存在许多聚类算法,笔者选择一次遍历聚类算法,主要利用该算法不用预先指定类数、运行速度快的优势,但是该算法也存在对输入次序敏感的问题,所以分 2 次聚类,在第一次扫描后,第二次重新聚类那些未找到类的文档,并合并相似的类,加强类的耦合度和紧凑度.

可见基于语义距离的文本聚类算法不需要预先指定类数,不仅改进了基于 VSM 的文本聚类算法在文本相似度计算方面的弱势,而且利用相似权重选择的类特征词更能代表类的主题特征,从而更符合文本聚类的具体要求.

## 4　结论

文本聚类是文本挖掘领域一个重要的技术,当前应用的文本聚类算法都是基于 VSM 模型的利用关键词权重匹配的方法,这种方法忽略了词语间的联系,忽略了语义信息,导致聚类结果主题特征不明显,失去聚类原有的目的. 从语义上具体分析文档内容,引入语义距离计算文本相似度,将文档间语义距离计算依次分解为关键词语义距离计算、义原间的语义距离计算,又考虑文本聚类的具体应用,改进了现有的词语相似度算法,更侧重利用词语的相关性计算词语间的相似度. 实践证明,改进的词语相似度计算方法更适合文本聚类的要求.

本文的聚类算法先利用最近邻聚类算法找出类特征词,然后整理、合并相似的类,再将未找到类的文档重新聚类,大幅提高了聚类精度. 引入词语的相似权重概念来优胜劣汰候选的类特征词,从而使代表类的特征词更加逼近主题.

**参考文献**:

[1] Likas A, Vlassis N, Verbeek J. The global k-means clustering algorithm[J]. Pattern Recognition, 2003, 36(2): 451.

[2] 李凡,林爱武,陈国社. 一种基于 VSM 文本分类系统的设计与实现[J]. 华中科技大学学报:自然科学版, 2005, 33(3): 53.
LI Fan, LIN Aiwu, CHEN Guoshe. A Chinese text categorization system based on the improved VSM[J]. Journal of the Huazhong University of Science and Technology: Nature Science, 2005, 33(3): 53.

[3] LI Sujian, ZHANG Jian, HUANG Xiong, et al. Semantic computation in Chinese question-answering system[J]. Journal of Computer Science and Technology, 2002, 17(6): 933.

[4] Fragos K, Maistros Y, Skourlas C. Word sense disambiguation using WordNet relations[C]//Proceeding of the 1st Balkan Conference in Informatics. Thessaloniki: Greek Computer Society, Aristotle University, University of Macedonia, Technological Institution of Thessaloniki, 2003, 633 - 643.

[5] WANG Yong, Hodges Julia. Document clustering with semantic analysis[C]//Proc of the 39th Annual Hawaii International Conference on System Sciences. Hawaii: University of Hawaii. Department of Information Technology Management, 2006.

[6] 董振东. 知网[EB/OL]. [2007 - 04 - 16]. http://www.keenage.com./zhiwang/c_zhiwang.html

DONG Zhendong. HowNet[EB/OL]. [2007 - 04 - 16]. http://www.keenage.com./zhiwang/c_zhiwang.html

[7] 刘群, 李素建. 基于《知网》的词汇语义相似度计算[J]. 计算语言学及中文信息处理, 2002, 7(2): 59.
LIU Qun, LI Sujian. Words meaning similarity computing based on HowNet[J]. Computational Linguistics and Chinese Information Processing, 2002, 7(2): 59.

[8] 中国科学院计算技术研究所数字化室和软件室. 中文自然语言处理开放平台[CD/DL]. [2007 - 03 - 06]. http://www.nlp.org.cn/
The Institute of Computing Technology of the Chinese Academy of Sciences Digital Section and Software Section. Chinese nature language procersing open platform[CD/OL]. [2007 - 03 - 06]. http://www.nlp.org.cn/

(上接第 1673 页)

JIANG Feng. Study on the technology treating slightly polluted water with PAC adsorption[D]. Shanghai: Tongji University. School of Environmental Science & Technology, 1994.

[3] Amy G L, Sierka R A, Bedessem J, et al. Molecular size distributions of dissolved organic matter[J]. Jour AWWA, 1992, 84(6): 67.

[4] Bouwer E J, Crowe P B. Biological processes in drinking water treatment[J]. Jour AWWA, 1988, 80(9): 82.

[5] 黄君礼, 鲍治宇. 紫外吸收光谱法及其应用[M]. 北京: 中国科学技术出版社, 1992.
HUANG Junli, BAO Zhiyu. UV absorption spectrum and application[M]. Beijing: China Scientific and Technical Publishers, 1992.

[6] Andrew Eaton. Measuring UV-absorbing organics: a standard method[J]. Jour AWWA, 1995, 87(2): 86.

[7] McCarty P L, Aieta E M. Chemical indicators and surrogate parameters in water treatment[J]. Jour AWWA, 1984, 56(10): 98.

[8] 董秉直, 曹达文, 范瑾初, 等. 黄浦江水源的溶解性有机物分子量分布变化的特点[J]. 环境科学学报, 2001, 21(5): 553.
DONG Binzhi, CAO Dawen, FAN Jinchu, et al. Characteristics of changes in distribution of molecular weight of dissolved organics in Huangpu River water source[J]. Acta Scientiae Circumstantiae, 2001, 21(5): 553.

[9] 董秉直, 李伟英, 陈燕, 等. 用有机物分子量变化评价不同处理方法去除有机物的效果[J]. 水处理技术, 2003, 29(3): 155.
DONG Binzhi, LI Weiying, CHEN Yan, et al. Evaluation with the change of organics MW distribution for effect of organics removal by different treatment method[J]. Water Treatment Technology, 2003, 29(3): 155.