

WRM: 一种基于单词相关度的文档聚类新方法

伍赛* 杨冬青* 韩近强* 张铭* 王文清⁺ 冯英⁺

(*北京大学信息与科学技术学院 北京 100871)

(⁺北京大学图书馆中国高等教育文献保障系统管理中心 北京 100871)

(wsai@db.pku.edu.cn)

摘要 目前大多数的搜索引擎如 Google、百度等，查询的结果都是按照重要度排序然后分页地显示给用户。但是有时候这样显示并不能很好地服务于用户，用户经常要浏览了很多页面才找到自己所需要的内容。如果将返回的结果再进行分类，就可以很好的解决这一问题。不同于传统的向量空间模型的方法，本文提出了一种基于单词相关度的聚类方法。实验的结果表明该方法具有较高的准确性和很高的效率。

关键字 文档聚类，单词相关度，单词向量空间模型 WVM，向量空间模型 VSM，TF/IDF，聚类引擎

中图法分类号 TP311

WRM: A Novel Document Clustering Method Based on Word Relation

Wu Sai* Yang Dong-Qing* Han Jin-Qiang* Zhang Ming* Wang Wen-Qing⁺ Feng Ying⁺

(* School of Electronics Engineering and Computer Science, Peking University, Beijing, China, 100871)

(⁺Administrative Center for China Academic Library & Information System Room 607, Peking University Library Beijing, China, 100871)

Abstract The most popular search engines, such as Google and Baidu, answer users' queries as lists of ranked results according to importance. But in some cases the most "important" is not the most useful for the user. A user has to look through several pages to get what he wants. Trying to classify the results is a good idea to solve this problem. In this paper, we propose a novel clustering method based on the word relation WRM, which is different from the traditional VSM method. Experiment results show that our method WRM is not only very effective but also efficient.

Keywords Document Clustering, Word Relation, Word Vector Model (WVM), Vector Space Model (VSM), TF/IDF, Clustering Engine

1. 引言*

面对网络资源爆炸式的激增，越来越多的人选择使用搜索引擎来帮助他们找到所需资源。当前比较著名的搜索引擎如 Google、Yahoo、百度等，都保持着大量而稳定的用户群。虽然这些搜索引擎在技术细节上各有独到的地方，但是在使用方式上始终保持着“用

户输入查询关键词，搜索引擎返回相关的结果及其链接”的传统方式。搜索引擎一般都对返回的结果进行排序，它将重要的网页排在前面。用户在浏览前一两页的时候一般都能找到他所需要的结果。这样的方式虽然简单有效，但有的时候也无法满足用户的要求。比如用户把“世界杯”作为关键字进行查询，返回的结果有传统的男足世界杯的，有女足世界杯的，也有电子游戏世界杯的，甚至还有机器人足球的世界杯。用户想要寻找关于机器人世界杯足球赛的情况，他就要在 Google 中浏览到第十页才找到他所想要的。用户难以从大量的结果网页中找到相关的资源，对他来说结果集合仍然过于庞大和杂乱。为了解决这样的问题，结果自动聚类是一个很好解决方法。当结果被分类显

*本项目由中国高等教育文献保障系统管理中心“网上科研服务原型系统”(200309DEV01)和国家自然科学基金“基于 WebGIS 的拓片检索与导航系统的关键技术研究”(国科金外资助字第 60221120144)的支持。

示给用户后，他便可以选择自己需要的类进行浏览，对他来说，结果集就变成了该类的文档而不是搜索引擎返回的全部文档。当某几个类文档数量很多而其他类相对较少的时候，该方法可以帮助用户浏览不容易找到的小类，这些小类的文章在搜索引擎的结果中一般都很靠后，不容易被找到。

为了对结果进行聚类，需要选择一种合适的聚类方法。该方法必须首先是高速的，因为就搜索引擎的应用来说，用户对响应时间一般都有很高的要求；其次，该方法也应该尽量准确，能够给用户带来检索的帮助。作者认为速度上的要求要比准确性上的要求要高。

当前存在的大部分聚类算法都是基于文档向量空间 VSM 的，这类算法的优点是易于理解，但是也有很显著的不足：算法并不知道每个类所代表的含义，仅仅是因为元素的相对距离较近所以将它们归为一类，导致结果的不准确，而且算法的速度比较慢，不能实用搜索引擎的应用。

本文提出了一种基于单词相关度的聚类方法 WRM (Document Clustering Method Based on Word Relation)。总体的思想首先充分利用单词之间的相关度分类来确定词的分类，然后计算每个文档向量和各个类之间的距离将其归入一个或者多个类别中。对于文档大小有限而数量较多的情况，这个算法可以很好的反映文档的分类情况。事实上，对于搜索引擎返回的结果（文档的一部分摘要）和它们本身所代表的原文档分别进行分类，比较结果发现误差为 20%[1]。对于追求速度的搜索引擎来说，这是可以忍受的。WRM 算法正好适用于这种情况，因为每个搜索结果都是一个小文档，但每次返回的结果却有上千篇之多。实验的结果表明，WRM 算法具有较高的准确性和很高的效率。

本算法在“网上科研服务原型系统”项目中加以实现。在该系统中我们实现了一个基于 SDARTS 协议的元搜索器，搜索器返回的结果使用 WRM 进行分类显示，大大方便了使用者的查询。

本文以下的部分将如下组织：第二节简单介绍相关的研究，第三节介绍单词向量空间模型，第四节介绍基于单词相关度的聚类 WRM，第五节介绍文档分类，第六节为实验情况简介，第七节为简单总结和展望。

2. 相关研究

关于文档聚类网址[2]给出许多综述性质的资料。目前在大多数系统中使用文档向量空间模型 VSM 来描述整个文档。对于文档集合 $D=\{D_1, D_2, D_3...D_n\}$ ，每一篇文档 D_i 对应一个向量 V_i ， V_i 定义为：

$$V_i = \{f(tf_1, idf_1), f(tf_2, idf_2) \dots f(tf_k, idf_k)\}$$

其中 tf_j 是单词 j 在文档 D_i 中出现的频率，而 idf_j 是单词 j 在文档集合 D 中出现的文档数， f 表示 tf_j 和 idf_j 的某种函数，比如常用的有 $f(tf_j, idf_j)=tf_j/idf_j$ ， $f(tf_j, idf_j)=tf_j(\log idf_j+1)$ 。

在使用文档向量模型的系统中，需要计算每个文档向量之间的距离，然后通过它们之间的距离使用 K-Means 或 DBScan 之类的聚类方法进行计算。通常这些方法的复杂性都是 $O(n^2)$ 。在计算文档向量的时候通常使用文档之间的余弦来代表它们的距离。向量 V_a ， V_b 的余弦定义如下：

$$\text{Cos } \theta = \frac{V_a * V_b}{|V_a| |V_b|}$$

然而试验证明基于文档空间模型的聚类算法，其分类效果很差，难以达到要求。为了改进这种情况，目前国外进行了一些研究，也出现了一些类似的商业系统，比较著名的有 vivisimo[3]、iboogie[4]。他们被人们称为聚类引擎。关于聚类引擎方面的研究工作，哥伦比亚大学的 Oren Zamir 和 Oren Etzioni 给出了一种聚类的方法[1][5]。他们采用后缀树 (Suffix Tree) 的数据结构进行快速聚类。这种方法能够将返回结果分类呈现给用户，更加方便他们对资源的查找。如果用户查找“以色列”，返回结果将按照：国家政治，宗教文化，旅游和地理等类分层显示给用户。用户可以选择他感兴趣的类别进行浏览，或者进行第二次查找。但是他们的方法由于使用后缀树的结构因此空间上的消耗比较大。

为了改进这种情况，Noam Slonim 和 Naftali Tishby 提出了使用单词聚类来指导文档聚类的方法[6]，并且实验证明他们的方法甚至可以和一些分类算法相媲美。在他们的方法中表示文档的单词被提取出来，然后计算它们属于一篇文档的概率，再通过贝叶斯公式来计算两个单词属于一个类的概率，进而通过此概率对单词进行聚类；然后再通过文档和单词类的关系对文档分类。受他们的启发，作者

放弃了传统的文档向量的方法，通过加入语义信息来构建一个类似文档向量的单词向量空间模型WVM (Word Vector Model)，单词向量 V_i 表示了一个单词 w_i 和所有其它单词之间的相关度。WRM根据单词之间的相关度进行分类，通过这样的分类可以抽取一些具体的代表类的概念和信息，用这些信息对下一步的文档分类做指引。和 Noam Slonim的方法最大的不同在于：我们考虑的不是单个词的属性，而是词之间的联系，从而使聚类更加准确。

此外还有一些文档聚类的使用是基于结构的方法，而不是基于内容的方法。基于结构的方法主要是用于信息提取的工作，无法解决前面提到的问题。

3. 单词向量空间

WRM的模型只考虑每两个单词之间的关系。如果两个单词分别在两篇文档中的某句话中同时出现，他们很可能就被作为分类的特征。按照这种思想作者采用一种单词向量空间模型。单词向量空间模型的数学模型是一个对称矩阵 M 。它的行列数相等，都为单词向量的维数也就是单词的个数。矩阵中元素的意义在于表示两个单词之间的相关度。

在WRM系统中，分词的粒度为文档中的句子。作者认为一句话中同时出现的单词才有相关意义的存在，并且它们之间距离不能太远。同时，多次出现在同一句话的两个单词具有更高相关度，基于这样的考虑，建立 M 的算法如下：

```

For 文档  $D_i \in$  文档集合  $D$ 
  For 句子  $S_j \in$  文档  $D_i$ 
     $M[w_x, w_y] = M[w_x, w_y] * 2 +$ 
       $1 / \text{Distance}(w_x, w_y) \forall$ 
       $w_x, w_y \in S_j \wedge w_x \neq w_y$ 
  
```

算法1 相关度矩阵的计算

$\text{Distance}(w_x, w_y)$ 表示单词 w_x 和 w_y 之间有几个单词的间隔。对两个单词来说，它们多次同时出现在一些句子中比它们在一句中多次出现要有更强的相关度。比如对于句子“新出的电脑赛车游戏《微软拉力赛车》比以往的电脑赛车类游戏都更加注重对赛车本身性能的模仿”和句子“电脑游戏已经慢慢走向了网络时代”，在第一句中“电脑”出现2次、“赛车”出现

4次而“游戏”出现2次，在第二句话中“电脑”和“游戏”都出现一次。对于都出现在第二句话中的“电脑”和“游戏”来说，它们具有较高的相关性；而“赛车”虽然与“电脑”和“游戏”在一个句子中同时出现了多次，但它与这两个词的相关性仍没有“电脑”和“游戏”这两个词之间的高。所以在对每个句子的单词建立相关度矩阵的时候，为了突出它们同时出现的频率越高越重要的特点，作者将原来的相关度乘上2再加上新值。而除以单词距离的平方值是为了加深两个单词之间的间距对它们的相关度的影响，这样在一句的两个词相距越近相关度越高。

另外作者认为名词在文档分类中的作用是最显著的，因此WRM仅统计了文档中的名词。此外，仅统计名词还可以起到降低维度、提高分类速度的作用。

WRM的这种办法要比传统方法文档向量空间法快，尤其是在文档个数非常多的情况下。作者假设单词集合 $W = \{w_1, w_2, \dots, w_n\}$ 包含了所有出现在文档集合 D 中的单词，在文档数量增加的情况下单词增

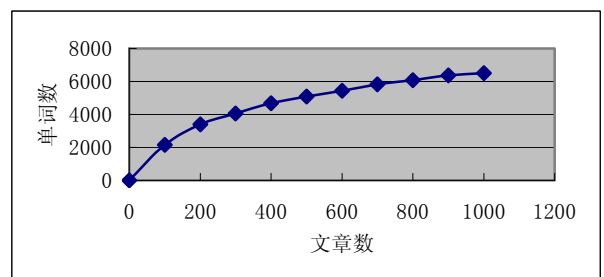


图1 文章数量和单词数量之间的关系

加的幅度是比较小的，图1表示了单词数量和文章数量之间的关系，其中的文章是从天网测试集中选取的1000篇文章，大小为10-30k之间。可以看到，当文档数达到一定程度时，单词数目的增加渐趋缓慢，从而也说明了单词向量的维度是可以控制在较小范围内的。

4. 单词聚类

设定一个阈值 C ，当单词 w_x 和 w_y 之间的相关度，即矩阵 M 第 x 行 y 列的值大于 C 的时候，作者认为它是高度相关的。单词的分类是基于这样一种思想：类别内部的单词是高度相关的，而它们与其他类之间的相关性应该尽可能地小。对于类别内部的单词的关系有两

种语义来描述[7]: 星型相关和完全相关。

星型相关指的是对于一个类的单词集合 $W=\{W_1, W_2... W_n\}$, 其中存在一个单词 W_i , 任何在 W 中的其他单词 W_j , 都和 W_i 是高度相关的。

完全相关是指对于一个类的单词集合 $W=\{W_1, W_2... W_n\}$, 任意两个不同的单词 W_i, W_j , 它们总是高度相关的。

使用星型相关可以快速地进行大量结点处理, 其代价为 $O(n)$, 但是得到聚类结果有些粗糙, 可能会出现因为一个中心点而将很多相关度很低的词归为一类的现象; 而使用完全相关得到的类是完美的, 但是运算代价也提升到 $O(n^2)$ 。因为单词分类是整个算法的核心部分, 所以作者采用完全相关以保证准确性, 而在速度上做出一定的牺牲。实验表明, 如果采用星型相关, 得到的结果的准确性是波动的。有时候接近完全相关的准确度, 有时候非常的差, 这可能和聚类起始点的选择有关。

单词分类开始随机选择单词来形成初始的类, 然后将剩余的单词加入这些类别或者生成新的类, 直到所有的单词都属于某个类为止。WRM允许一个单词属于多个类, 这也符合现实中一词多义的情况。聚类的算法如算法2所示。

该算法复杂度为 $O(n^2)$, n 为单词的个数。通过上面的严格提取, 作者得到了一些原子类, 这些类的元素两两之间互相相关。但是原子类别往往过于

```
while 还有单词没有被分类{
    随机取出一个单词W与现有的类一一比较;
    if W与某个类中的元素具有完全关联关系
    then 将W加入到该类中;
    if W与任何类都不具有完全关联关系
    then 将W加入新类C中;
        for 所有的单词Wj
            if Wj和W所在新类C构成完全
            关联关系
            then 将Wj加入新类C中;
}
```

细致和繁多, 不能作为一个独立的类别来对待; 并且有些原子类之间也存在很密切的联系, 所以应该将结合紧密的原子类合并为一个大类。两个类合并, 即将它们的元素合并到一起并且去处重复的元素。

算法2 单词聚类算法

如何定义类之间的是密切联系的? Oren Zamir 的策略是认为两个类共有二分之一的元素就应该合并。也就是, 说对类 $C1$ 和类 $C2$, 它们能合并的条件是: $|C1 \cap C2| > \text{Min}(|C1|, |C2|)/2$ 。

但是这样会导致个别类规模很大的现象。为了避免这种情况的发生, 作者规定两个类的相关度要除以它们的大小之和, 即:

$$R(C1, C2) = \frac{\sum M(c1, c2)}{|C1| + |C2|} \quad \forall c1 \in C1, c2 \in C2$$

如果这个值超过阈值两个类就可以应该合并。

类的合并过程类似于层次聚类算法, 可以用图2的二叉树表示。为了避免个别类规模很大的现象, 在合并的时候优先考虑层次较低的类之间的合并。图2表示将原子类1-7合并为两个类的过程, 由于类A、类B之间的相关度 $R(A, B)$ 没有超过阈值, 所以不能继续合并。可以看到通过控制阈值就可以控制二叉树的合并程度, 这是控制聚类粒度的一种手段。

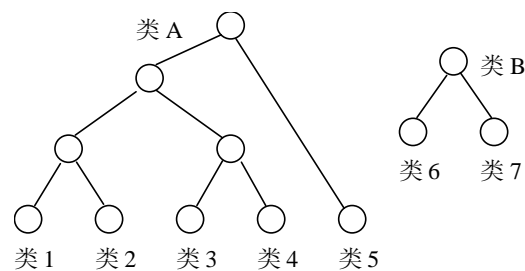


图2 原子类的合并

通过阈值的控制, 用户可以选择单词聚类的粒度。如果阈值过高, 那么将不会发生原子类合并的现象; 如果阈值过低, 那么所有的类别可能都被合并到一个大类里面。所以用户应该根据需求选择合适的阈值, 以控制所得到类的粒度, 初始的阈值设置为原子类中的平均相关度。单词分类的最后一步是将合并后的结果进一步过滤, 去除掉一些元素太少的类, 这些类别不能给文档的分类带来太大的帮助, 相当于一般聚类算法里面的孤立点。

5. 文档分类

分好类的单词被用来指导文档的分类, 每个文档向量与所有的单词类比较, 加入到合适的类中。这有点像一般的文档分类方法: 首先根据训练集合得到一些特征值, 然后将待分类文档向量与每个特征值进行

比较，归入最可能的类中。作者使用分类好的单词代表该文档类。

设文档 D_i 的向量为 $V_i=\{tf_1,tf_2...tf_n\}$ ， tf_j 表示的是单词 w_j 在文档中出现的次数。为了计算每一篇文档和每个单词类的关系，需要根据文档向量和类 C 中的元素（即单词）对文档打分。

$$M = \sum_{w \in C} I_w * tf_w$$

I_w 为单词 w 在该类 C 中的重要性。 I_w 的计算使用下面的公式：

$$I_w = \frac{\sum_{w1 \in C} M(w, w1)}{\sum_{w1 \in C, w2 \in C} M(w1, w2)}$$

不同于一般的系统，WRM允许一个文档属于多个类。这是符合现实情况的。比如有一篇关于电视剧的文档，该电视剧描写的是足球运动员的生活，这篇文档归于娱乐或体育都是可以接受的。所以只要文档属于某个类的可能性超过一个阈值，WRM将它归入这个类，如果它与所有类都不具有这样的关系，那么就将它归入可能性最高的一个类。

最后不同类别的文档被存储到不同的文件夹中，每个文件夹的名字为该类中出现频率最多的几个词。但是在某些情况下，比如关于体育报道和娱乐报道的类中，记者往往是出现频率很高的一个词，但它不能很好的表示本类的内容。所以对于那些同时出现在多个类中的高频率词，并不将其作为类名。

6. 实验结果

测试集是娱乐、财经、教育、军事、体育和科技的文档各100篇文章，文章的大小在1到3k之间。实验的机器为P4 2.2GHZ、512M DRAM，系统为Windows 2000，使用Java语言实现。同时为了比较性能，还实现了一个基于传统向量空间的聚类算法VSM和自动网页分类算法COMMIX-Classifier[8]（这里简称COMC）。所有的算法都允许文档属于多个类，并且对于COMC算法我们每个类提取了1000个特征项，以得到较准确的结果。

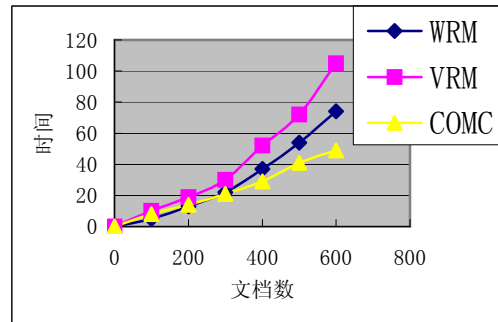


图3 三种方法的运行时间

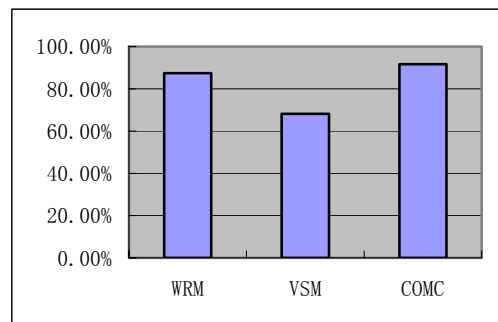


图4 三种方法的查准率

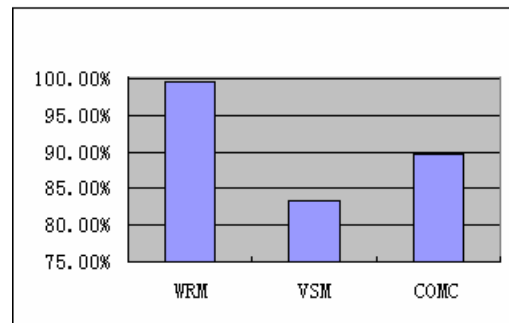


图5 三种方法的查全率

可以看到基于单词的聚类算法的查准率略低于基于训练的分类算法，而传统的基于向量空间的聚类算法效果却很差。事实上聚类算法一般要比分类算法的效果差很多，因为分类是在有指导信息有训练集的条件下进行的，而聚类算法没有任何的外部信息可以使用。基于单词的聚类算法之所以有这么高的准确性，因为它使用了单词分类作为文档聚类的指导，这一点类似于分类算法，不同之处是单词分类并不是外部给出的，而是通过对文章的分析而自动得到的。

在查全率方面，基于单词的WRM具有很高的查全率，基本上为100%，这是因为文档和单词之间的密切联系。而VSM方法和COMC方法只使用了文档特征，查全率不是很高。

一般的聚类算法大部分是 $O(n^2)$ 级[9]。传统的TF/IDF模型中 n 为文档数目，而本文的方法中 n 为单词的数目。一种语言中单词的数目是有限的。因此本文聚类算法在大量文档的分类中要比传统的算法快很多。

在我们的元搜索引擎上进行实验，如果用户输入“中国”，那么返回的结果将被分为：“位置，国土，气候”、“历史，人文，文化”、“社会主义，经济，人民币”和“教育，高校，汉语”等几个类，每个类使用该类中权重最大的3个词来做类名，并显示3篇分数最高的文档。用户可以方便的选择他所感兴趣的内容展开浏览，大大提高了其检索速度。

7. 进一步的工作

对于搜索引擎这种应用聚类的速度应该是越快越好。作者下一步的工作主要集中在进一步提高现有算法的速度。算法的时间开销主要是用来计算单词的相关矩阵。相关矩阵的维数等于单词的总数目，如果能够对单词进行有效的降低维度，算法的计算速度将大大加快。目前WRM采用比较简单的降低维度的方法。首先去除在所有文档中出现的和仅在一篇文档中出现的词，然后去除名词外的其他单词，因为那些单词会影响速度和准确性。但是其实名词也有很多与分类无关的词比如：消息、情况等；而非名词也有一些包含分类信息的，如：治疗、编程、扣篮等。考虑更多的情况建立一个完善的降低维度的选词方案也是下一步急需解决的。

由于缺乏一定的指导，在单词聚类的时候有些情

作者简介：



伍赛，男，1980年生，硕士研究生，主要研究方向为数字图书馆、数据库与信息系统。

况会不准确。比如，因为几篇文档都是描写电影和现代科技之间的关系，算法就会将电影和计算机归为一类。如果适当的加入一些Ontology的指导，这类现象完全可以避免，从而得到更好的结果。因此如何实现一个符合本文要求的简洁的Ontology也是要继续研究的问题。

参考文献

1. O. Zamir and O. Etzioni, Web document clustering: a feasibility demonstration, in: *Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*, 1998, p 46-54.
2. http://www.iturls.com/TechHotspot/TH_DocCluster.asp
3. <http://vivisimo.com/>
4. <http://iboogie.tv/>
5. Oren Zamir and Oren Etzioni, Grouper: A Dynamic Clustering Interface to Web Search Results, *Computer Networks*, 1999, p 1361—1374
6. Noam Slonim and Naftali Tishby, Document Clustering using Word Clusters via the Information Bottleneck Method, *Research and Development in Information Retrieval*, 2000, p 208-215
7. Sara Cohen and Jonathan Mamou and Yaron Kanza and Yehoshua Sagiv, XSEarch: A Semantic Search Engine for XML, VLDB2002
8. Li Liyu and Tang Shiwei and Yang Dongqing and Ye Hengqiang and Wang Tengjiao, COMMIX-Classifer- An Automatic Web Page Categorization System, NDBC 99
9. Jianwei Han and Micheline Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publisher



杨冬青，女，1945年生，教授，博士生导师，主要研究方向为数据库与信息系统。



韩近强，男，1979年生，硕士研究生，主要研究方向为海量信息处理、数据库与信息系统。



张铭，女，1966年生，在职博士研究生，副教授，主要研究方向为数字图书馆、数据库与信息系统。



冯英，女，1970年生，硕士，工程师，主要研究方向为数字图书馆。



王文清，男，1965年生，博士，高级工程师，主要研究方向为数字图书馆。