中文文本聚类的研究与实现*

张宝艳　王庆辉

(北京邮电大学信息工程学院智能研究中心, 北京 100876);

E-mail: zhangby@bt-t.com

摘要: 在文本聚类中，聚类的最终结果应该是一棵树的形式。然而，随着互联网的普及，面对海量的电子文献，学科分枝的越来越细化，树的分枝粒度越来越小，逐层聚类必然会花费巨大的时间。本文讨论并提出了针对特定领域扁平聚类和分层聚类相结合的思想，并且对于文本预处理和具有较强实用性的 ISODATA 扁平算法给出了 VC++的实现.

关键词: 自然语言理解; 向量空间模型; ISODATA; 文本聚类

引言

随着互联网的出现，大量的文本信息如潮水般不断涌现，网络已经成为一个庞大而杂乱无章的桌面图书馆。对海量的文献人们迫切需要能够自动实现文本的分类处理，在节省时间的同时更好的定位查找自己需要的文献。

文本分类又分为有监督的文本分类和无监督的文本分类。有监督的文本分类是已知一批训练文本的标签，通过机器学习得到文本分类器。当新的文本出现时，根据分类器确定新文本的类标签; 无监督的分类是从待分类文本中提出特征，然后将提出的全部特征进行比较，再根据一定的原则将具有相同或相近特征的文本定义为一类，并设法使各类中包含的对象大致相等。通常我们称无监督的文本分类为文本聚类。

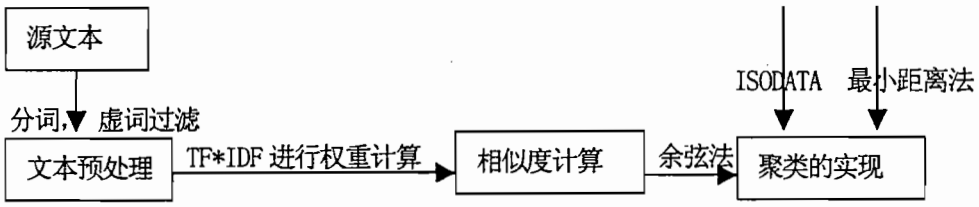
本论文给出根据领域的不同分别生成分层聚类的算法，该算法先根据大的领域对文本进行扁平聚类——即聚成不同的堆，再根据类内文献间的距离（相似性）自下而上类别精化。既缓解了自下而上类别精化法的次序独立性问题，又减少了重叠度。

1. 理论与算法

1.1 总体流程

基于文本的聚类系统分为三个模块（如下图所示）：文本的向量表示，文本的相似度计算，聚类方案的选择与实现。

* 本项目受到国家自然科学基金资助(69982001)。



1.1.1 文本的向量表示

对原文本进行分词和词性标注之后，过滤虚词和禁用词，生成每篇文章的有效词列表，最后对所有的有效词进行汇总，得到了一张总词表，把总词表里的每个词都作为向量的分量，根据每篇文章的有效词列表，就生成了所有文章的向量。

简单的说，首先统计文档集里出现过的所有词，得到一部词典，对词典里的每个词一个编号。最简单的编号方法是该词在词典里的位置，第一个词编号为 1，如此类推。

这样我们就可以根据该词典对每篇文章进行向量化：

词典(总词数为 n)	Word(1)	Word(2)	Word(3)	Word(n)
文章 1 中每个词的出现次数：	f1(1)	f1(2)	f1(3) ...	f1(n)
文章 2 中每个词的出现次数：	f2(1)	f2(2)	f2(3) ...	f2(n)

。。

对于传统空间向量模型，向量化的结果如下：

	文章 1	文章 2	文章 3……文章 n
词 1 在每篇文章中的权重：	w11	w12	w13 … w1n
词 2 在每篇文章中的权重：	w21	w22	w23 … w2n

1.1.2 文本的相似度计算

在文本被向量化之后，就可以用数学方式来对文本进行处理，我们采用余弦法来计算相似度，以词频反篇章概率法计算出权重，计算文章 Di 和 Dj 的相似度如下：

$$Sim(d_i,d_j)=Cos\theta=\frac{\sum_{k=1}^{M} w_{ik}*w_{jk}}{\sqrt{(\sum_{k=1}^{M} w_{ik}^{2})(\sum_{k=1}^{M} w_{jk}^{2})}}$$

1.1.3 文本聚类的实现

文本聚类分为两个步骤，第一步为扁平聚类，我们选用叠代自组织算法，优点在于叠代自组织算法的次序独立性，没有重叠度。

具体算法如下：

思路：给定一些大致参数（根据目的）。

原则：①样本数太少的类 - 取消；②类内离散太大的类 - 分裂；③距离近的类 - 合并。

1) 给一些参数

K：期望分类个数的大致范围

θ_K：一个类内的最少样本数

θ_S：关于类内分散程度的参数

θ_C：关于类间距离（最小）的参数

L：每次叠代允许合并的类数

I：允许叠代的最大次数

2) 适当选取类中心$\{Z_1, Z_2, \ldots, Z_{N_c}\}$，$N_c$：类数

2)' 分配样本。如果有$\{i=1,2,\ldots,N_c\}$

$\| Z - Z_j \| \leq \| Z - Z_i \|$，则$Z \in S_j, j = 1,2,\ldots,N_c$

3) 如果S_j类样本数$N_j < \theta_k$，则取消S_j类。$N_c=N_c-1$，goto 2)'

4) 重新计算各类中心 $Z_j = \frac{1}{N_j} \sum_{Z \in S_j} Z, j = 1,2,\ldots,N_c$

5) 计算类S_j内平均距离 $\overline{D_j} = \frac{1}{N_j} \sum_{Z \in S_j} \| Z - Z_j \|, j = 1,2,\ldots,N_c$

6) 对全体样本求类内距离平均值 $\overline{D} = \frac{1}{N} \sum_{j=1}^{N_c} N_j \cdot \overline{D_j}, N = \sum_{j=1}^{N_c} N_j$

7) [a]如果叠代次数≥I，则转向 11)（合并）

[b]若$N_c \leq K/2$，则转向 8)（分裂）

[c]若偶数次叠代或$N_c \geq 2K$，则转向 11)（合并）

8) 计算各类中各分量的标准差

$\sigma_{ij} = \sqrt{\frac{1}{N_j} \sum_{Z \in S_j} (x_{ik} - z_{ij})^2}$，i=1, 2, . . . , n，j=1, 2, . . . , N_c，k=1, 2, . . . , N_j

x_{ik}为$Z \in S_j$的第 i 个分量，z_{ij}为Z_j的第 i 个分量。

σ_{ij}为第 j 类第 i 个分量标准差

9) 找到各类的标准差最大的分量

$\sigma_{j\max} = \max\{\sigma_{1j}, \sigma_{2j}, \ldots, \sigma_{nj}\}, j = 1,2,\ldots N_c$

10) 分裂：条件 1. $\sigma_{j\max} > \theta_S$且$\overline{D_j} > \overline{D}$且$N_j > 2(\theta_k + 1)$

条件 2. $\sigma_{j\max} > \theta_S$且$N_c \leq K/2$

若满足两条件之一，则分裂S_j

(a) 建立Z_j^+和Z_j^-，2 个新的类中心，$N_c=N_c+1$

其中Z_j^+和Z_j^-是沿着$\sigma_{j\max}$轴，在原来的Z_j位置上，分别加上和减去一个数$k\sigma_{j\max}(0 < k \leq 1)$。k 是经验值。

(b) goto 2)'（分配样本）。

11) 计算所有各类中心的相互距离 $D_{ij} = \| Z_i - Z_j \|, i = 1,2,..., N_{c-1}, j = i+1,..., N_c$

12) 对于比 θ_c 小的 D_{ij} 从小到大排队。假定为

$$D_{i_1 j_1} \le D_{i_2 j_2} \le ... \le D_{i_L j_L}$$

13) 按 l=1, 2, . . . , L 的顺序，把 $D_{i_l j_l}$ 对应的 Z_{i_l} 和 Z_{j_l} 合并

$$Z_l^* = \frac{1}{N_{i_l} + N_{j_l}} \left[N_{i_l} Z_{i_l} + N_{j_l} Z_{j_l} \right], N_c = N_c - 1$$

计算 $D_{i_l j_l}$ 时的 Z_i，Z_j，若至少其中一个是在本次叠代中合并取得类中心，则越过此项。

14) 若叠代次数≥I，或参数无改变，则终止。

否则 goto 2)'，需要时可返回 1) 修改参数

至此，文本的扁平聚类结束，得到的是若干类文本的聚类；这样的聚类使得每一个类中的文本数还是很大，要想得到功能更强的分类器，对于特定的领域，还需要更进一步的细化分类，还需要有一个逐层分类的过程，那就是分层聚类.具体算法如下：

思路：寻找“距离”最近的两个样本结合

1. 有 N 个样本的集合 $Z_s=\{Z_1, Z_2, ..., Z_N\}$

2. 若想要聚成 K 个类 （事先给定 K）

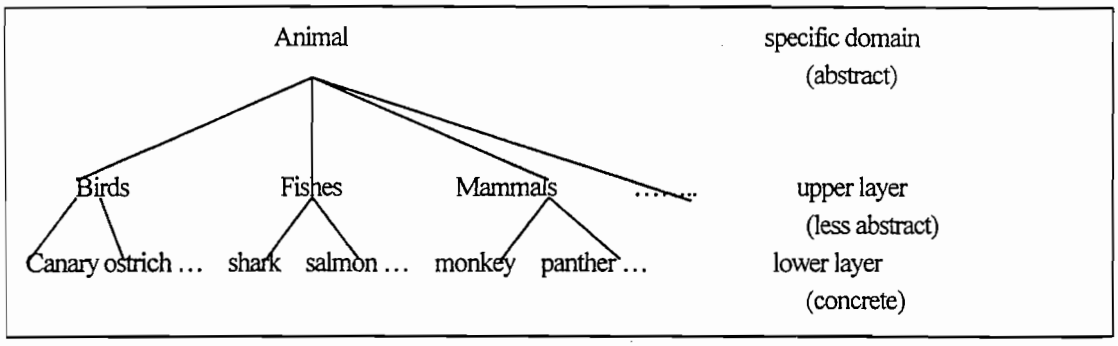
[1] k=N, $C_i=\{Z_i\}$, i=1,2,...,N

[2] if k=K then END

[3] 找到 C_i 与 C_j 之间的距离 $d(C_i, C_j)$ 最小的一对

[4] C_i 和 C_j 合成一个类 C_i，并计算新的 C_i 的中心

[5] 去除 C_j, k=k-1. goto [2]



经过两次聚类处理，就得到了一棵棵树，在每棵树中都是一个分层的扁平聚类。在在线文本聚类、搜索引擎中都有较强的实用价值。

2. 实现与结语

本文对人工挑选的 300 篇文档进行了叠代自组织文本聚类训练，得到了和预期一致的结果。由于在不同的类内所挑选的文本区分度不大，所以没有进一步地进行分层聚类，但是面对海量的桌面图书馆文献，

仅仅是扁平聚类很难满足查询的需要。笔者也深深感到：离开了具体的应用环境，再华美的理论也只能是纸上谈兵而已。

目前对于聚类的算法或是次序不独立，或是重叠度不满足要求，又由于文本的多样性，复杂性以及待分类对象的广泛性使得文本聚类大多仅限于理论上的讨论，很少有投入实际应用。本文试图在文本聚类的具体应用环境下，提出一种能够减缓文本的次序独立性又能够尽量避免重叠度的聚类方式，给出了相应的算法。

参考文献：

[1] http://www.cs.cmu.edu/~dunja/KDDpapers/Steinbach_IR.pdf

[2] Dik L.LEE, Comp336 Spring 2000, Department of Computer Science, hkust

[3] http://202.112.116.44/documents/分类

[4] Yiming Yang, Xin Liu, A re-examination of text categorization methods, School of Computer Science, Carneigie Mellon University, Pittsburgh, PA, USA, http://www.cs.cmu.edu/yiming

[5]《文本自动分类系统的研究与实现》张莉 硕士论文，北京邮电大学 2001.

[6] "自然语言理解与人机合作"部分，《中国人工智能进展》论文集（2001）中国人工智能学会、北京邮电大学出版社

致谢： 北京邮电大学智能研究中心王枞教授、钟义信教授在本文的写作中给予了悉心的指导，美国 Rutgers 大学计算机工程系张莉博士在成文过程中给予了很大的帮助，在此一并表示衷心的感谢。

作者简介： 张宝艳（1977— ），女，河北唐山人，硕士生，主要研究领域为文本分类与聚类；王庆辉（1975— ），男，河北滦南人，硕士研究生，主要研究领域为模式识别、语音合成、自然语言处理。

Research and realizations of Chinese document clustering*

Zhang baoyan[1] Wang qinghui[1]

[1](Beijing University of Posts & Telecommunications, Beijing 100876, China);

E-mail: zhangby@bt-t.com

Abstract: The ideal mode of document clustering is a tree. But, as the development of internet,the document is more and more, the branch of subject and the granularity is smaller and samller, the hiberarchy cluster will spend too much time, at the same time, it is useless to cluster absolutely different material documents into a tree. Based on these,this paper finds a two-step cluster solution, the first is a non- hiberarchy cluster, categorizate the related documents into several cluster, then a hiberarchy one makes the clusters form a tree, thus the orientation of the reader is more effective.

Key words: **Natural language processing, Document clustering, ISODATA, Vector Space Model**

* Supported by the National Natural Science Foundation of China under Grant No.69982001