中 文 信 息 学 报
第 20 卷 第 1 期 JOURNAL OF CHINESE INFORMATION PROCESSING Vol. 20 No. 1

文章编号:1003 - 0077(2006)01 - 0029 - 08

基于多策略优化的分治多层聚类算法的话题发现研究*

骆卫华[1,2],于满泉[1,2],许洪波[1],王 斌[1],程学旗[1]

(1. 中国科学院计算技术研究所,北京 100080; 2. 中国科学院研究生院,北京 100039)

摘要:话题发现与跟踪是一项评测驱动的研究,旨在依据事件对语言文本信息流进行组织利用。自 1996 年提出以来,该研究得到了越来越广泛的关注。本文在研究已有成熟算法的基础上,提出了基于分治多层聚类的话题发现算法,其核心思想是把全部数据分割成具有一定相关性的分组,对各个分组分别进行聚类,得到各个分组内部的话题(微类),然后对所有的微类再进行聚类,得到最终的话题,在聚类的过程中采用多种策略进行优化,以保证聚类的效果。基于该算法的系统在 TDT4 中文语料上进行了测试,结果表明该算法属于目前结果最好的算法之一。

关键词:计算机应用;中文信息处理;话题发现与跟踪;分治多层聚类;系统聚类

中图分类号:TP391 **文献标识码**:A

The Study of Topic Detection Based on Algorithm of Division and Multi-level Clustering with Multi-strategy Optimization

LUO Wei-hua[1,2], YU Man-quan[1,2], XU Hong-bo[1], WANG Bin[1], CHENG Xue-qi[1]

(1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China;
2. Graduate School of Chinese Academy of Sciences, Beijing 100039, China)

Abstract: Topic Detection and Tracking is a research driven by evaluation, which intends to organize and utilize information stream of texts according to event. Since being brought forward in 1996, it comes under more and more attention. This paper proposes an algorithm of division and multi-level clustering with multi-strategy optimization, which bases on study of today's mature algorithms. The core thought of the algorithm is to divide all data into groups (each group has intrinsic relevance), and cluster in each group to produce micro-clusters, and then cluster on all micro-clusters to result in final topics. During the process, various strategies are employed to improve the effect of clustering. The system implemented with the algorithm has been tested on TDT4 corpus. The test indicates the algorithm is one present best algorithm.

Key words: computer application; Chinese information processing; topic detection and tracking; division and multi-level clustering; hierarchical clustering

1 应用背景

在目前信息爆炸的情况下,信息的来源已不再是问题,而如何快捷准确的获取感兴趣的信息才是人们关注的主要问题。目前各种信息检索、过滤、提取技术都是围绕这个目的展开的。但一般的检索工具都基于关键词,返回的信息冗余度过高,因此人们迫切地希望拥有一种工

* **收稿日期**:2005 - 05 - 20 **定稿日期**:2005 - 10 - 17
基金项目:国家 973 资助项目(2004CB318109)
作者简介:骆卫华(1977 —),男,助理研究员,主要研究方向为文本分类与聚类,信息检索等.

具,能够自动把相关话题的信息汇总供人查阅。话题发现与跟踪(Topic Detection and Tracking,简称 TDT)技术就是在这种情况下应运而生的。TDT 是一项旨在依据事件对语言文本信息流进行组织利用的研究,也是为应对信息过载问题而提出的一项应用研究。

TDT 的概念最早产生于 1996 年,当时美国国防高级研究计划署(DARPA)提出要开发一种新技术,能在没有人工干预的情况下自动判断新闻数据流的主题。从 1998 年开始,在 DARPA 支持下,美国标准技术研究所(NIST)每年都要举办 TDT 国际会议并进行相应的系统评测,参加者包括 IBM Watson 研究中心、BBN 公司、卡耐基梅隆大学、马萨诸塞大学等著名大学和研究机构。最近中科院计算所、北京大学和东北大学的研究人员也开始进行跟踪和研究。

本文首先介绍了 TDT 的相关定义以及作为 TDT 研究平台的 TDT 评测会议的发展历程,并分析了 TDT 的研究现状,针对现有算法的不足,提出了多策略优化的分治多层聚类算法来解决话题发现问题,然后介绍了基于此算法实现的系统,给出了此系统在 TDT4 语料上的测试结果,并对实验结果进行了详细分析。

2 研究现状

在《TDT 评测计划》中,话题定义为"一个核心事件或活动以及与之直接相关的事件或活动"。"话题发现与跟踪"就是"在新闻电讯和广播等来源的数据流中自动发现话题,并且把话题相关的内容联系在一起的技术"。例如,"俄州爆炸案"这个话题包括 1995 年美国联邦大楼被炸、悼念仪式、联邦政府的一系列调查等等,但同一天发生在别处的爆炸不属于这个话题。

TDT 包括五项子任务,即:报道切分、话题跟踪、话题发现、新事件发现和报道关联发现,话题跟踪和话题发现是 TDT 的核心技术。话题发现本质上是一个将文本集分组的全自动处理过程。如果把文本内容作为聚类的基础,不同的组就与文本集不同的主题相对应。

总的来看,目前的话题发现算法主要存在如下问题:(1)大部分沿用原有的信息检索方法,对于"什么是话题"、"话题与报道之间的关系"等问题,目前的研究还不够深入;(2)传统的聚类算法系统开销很大,实用性不高;(3)虽然评测要求话题发现算法尽量语种无关,但实际上如果不利用各个语种的特征,算法的效果往往差强人意,现有的算法主要是在英语语料上测试与调整,在中文及跨语言语料上的效果要大打折扣。

3 基于多策略优化的分治多层聚类算法的话题发现

3.1 算法基本思想与流程

我们的目的是开发面向实际应用的话题发现系统,为此,我们对现有的成熟算法加以总结,针对其中的一些问题加以改进,提出了多策略优化的分治多层聚类算法(Algorithm of Division and Multilevel Clustering with Multistrategy Optimization,简写为 DMCMO),针对语种特点加入了一些优化策略,使之适合处理大规模语料,而且能更好地处理跨语言问题。为了验证算法的性能,我们在 TDT 语料上对该算法进行测试,取得了比较理想的结果。

算法的基本思想是:通过一定的策略把全部数据分割成具有一定相关性的分组,对各个分组分别进行聚类,得到各个分组内部的话题,称之为微类(microcluster);然后对所有的微类再进行聚类,得到最终的话题;在聚类的过程中采用多种策略进行优化,以保证聚类的效果。

3.2 数据分治策略

对数据分治两个好处:首先是降低系统开销。系统聚类的时间复杂度达到 $O(n^2)$,当数据规模很大时,普通的 PC 机将无法正常运行得出结果,而把大的数据集切分成数据规模较小

的组,单个数据集的运算开销大大降低,从而保证算法在大规模数据上的有效性。其次,通过一定的分治策略,可以事先就把具有内在相关性的数据放在一起进行计算,而把不太相关的数据分隔开来,从而避免把内容相似但实际上讨论两个话题的报道聚在一起,比如第一次海湾战争和第二次海湾战争。那些时间跨度较大的话题,还有机会通过组间聚类再合并成一个话题。

时间是话题的一个重要特征。一般来说,讨论同一话题的报道在时间上往往是相近的,时间间隔越远越倾向于描述不同的话题。按照出现的模式,话题可分为"突发性话题"和"持久性话题"。突发性话题的特点是对该话题的报道集中出现在某个很短的时间段内;持久性话题的特点是持续时间长,但报道的数量在各个时间段内呈现不均匀分布。这两种模式并无本质区别,持久性话题也可以看作多个突发性话题的集合。按时间顺序切分语料,每组内的报道就具有一定的时间相关性。

3.3 话题的特征空间与文档空间

对于内容这个难以表示的特征,我们采用向量空间模型,每个文档 d 表示成一个范化特征向量 $V(d)=(t_1,w_1(d);\cdots,t_i,w_i(d);\cdots;t_n,w_n(d))$,其中 t_i 为特征项,$w_i(d)$ 为 t_i 在 d 中的权值。为了更准确地表示文档内容,我们取词及其标注的词性为一个特征项,称为一个语义单元。每个语义单元 t_i 的权重为:$Wt(t_i)=\log(TF(t_i,d)+1)*\log((N/DF(t_i,d))+1)$。其中词频 $TF(t_i,d)$ 为特征 t_i 在文档 d 中的出现频度,文档频率 $DF(t_i,d)$ 为 t_i 在其中至少出现一次的文档的数目,N 为总文档数。$Wt(t_i)$ 刻画了特征 t_i 区分文档属性的能力。

通常情况下,话题包含不止一个文档,因此话题实际上由两部分构成:$T=<F,D>$,其中 F 是话题的特征空间,D 是属于该话题的文档空间。话题模型采用中心向量来表示,即

$F=<t_1,w_1;t_2,w_2;\cdots;t_m,w_m>$,其中 $w_i=(w_{id1}+w_{id2}+\cdots+w_{idn})/n$。

3.4 基于分组的(组内)系统聚类

算法的基本流程是:

对于每一组数据 $S_K=\{d'_{k+1},d'_{k+2},\cdots,d'_{k+w}\}$,执行如下操作:

(1)将 S_K 中的每个文档 d_i' 转化为面向话题的 VSM;

(2)将 S_K 中每个文档 d_i' 看成有一个成员的微类 $c_i=\{d_i'\}$,这些微类构成了 S_K 的聚类 $M_k=\{m_1,\cdots,m_i,\cdots,m_l\}$;

(3)计算 C_k 中每对微类 (m_i,m_j) 之间的相似度 $sim(m_i,m_j)$;

(4)若有满足 argmax $sim(m_i,m_j)$ 且 $sim(m_i,m_j)>\Theta$ 的类对 (m_i,m_j),将其合并为新微类 $m_k=m_i\cup m_j$(Θ 为相似度阈值),构成 D 的新聚类 $M_k=\{m_i,\cdots,m_{l-1}\}$,然后重复(1)到(4)步;否则,转向(5);

(5)把 S_K 的聚类 M_k 中的元素(即每个微类)逐一添加到 M 中。

第一层聚类的目的是尽可能准确地把各个组内的文档聚合成微类。由于文本包含的信息较少,而且微类还要用于后续处理,因此采用系统聚类来计算。其基本思想是:开始时各个样本自成一类,然后选择距离最近的一对合并成一个新类,计算新类和其他类的聚类,再将距离最近的两类合并,直到满足停止条件。算法本身的思想简单有效,但具体操作时还需要解决以下两个问题:(1)类和类之间的相关性度量;(2)合并操作。

要判断文档 d_i 和 d_j 的相关度,就要计算两者的相似度,然后把结果和阈值进行比较。向量空间模型通常采用 cosine 公式计算相似度,即求两者的内积。由于文档 VSM 采用语义单元作为特征,因此这个相似度就是两者在内容上的相似度。

和以往的信息检索任务不同,话题本身除了拥有语义特征之外,还具有时间特征,属于同一个话题的文档除了内容相关之外,在时间分布上也有一定联系。为此,我们在计算相似度的时候还考虑了时间因素,加入了时间衰减函数 T,其形式为:

$$T(d_i, d_j) = \begin{cases} 1 - t/m & \text{如果 } t < m \\ 0 & \text{否则} \end{cases}$$

其中 $t = |S(d_i) - S(d_j)|$,$S(d_i)$为文档 d_i 的时间戳,即文档 d_i 描述的事件发生的时间,可以简单地认为就是 d_i 的创建时间,m 是时间衰减因子,用于控制衰减速度和最大允许的时间间隔。

最终得到的相似度为加入了时间衰减因子修正之后的结果,即 $sim(d_i, d_j) = D(d_i, d_j) * T(d_i, d_j)$。

话题和文档的比较要复杂一些,因为话题通常包含多个文档,这涉及到集合和元素的比较。我们直接用话题的特征空间来表示话题,计算话题的中心向量和文档向量之间的相似度,并通过阈值策略来决定文档与话题是否相关。我们采用阈值策略来控制系统聚类迭代的次数,设置合并的相似度阈值 Θ,只有相似度大于 Θ时才能进行合并。

文档和话题、话题和话题之间的合并不是简单的集合并操作。虽然话题的文档空间进行的是并运算,但话题特征空间在合并的同时还需要更新,对此我们采用权值平均法。具体做法是:设某个特征 t_i 在话题 C_1 中的权重为 w_1,$DF(t_i)$为 m,在话题 C_2 中权重为 w_2,$DF(t_i)$为 n,则取算术平均之后的权重 w_i'作为合并后的权重,即 $w_i' = (w_1 * m + w_2 * n)/(m + n)$。

通过系统聚类,系统得到各个组内的话题,这些话题作为中间结果,即微类。当所有的组都完成聚类,就以微类作为待处理的数据进行第二层聚类。

3.5 基于微类的(组间)增量式聚类

这部分算法的基本流程是:

令聚类集合 V 为空,对微类集合 $M = \{m_1, \cdots, m_p\}$执行如下操作:

(1)如果 V 为空,创建一个新类 $T_1 = \{m_1\}$,把 m_1 从 M 中删除;否则转向(2);

(2)如果 V 为空,转向(4);

(3)按编号顺序取 M 中的第一个元素 m_i,逐一计算 m_i 与 $T_j \in V$ 的相似度 $sim(m_i, T_j)$;

(4)如果能找到满足 argmax $sim(m_i, T_j)$ 且 $sim(m_i, T_j) > \Theta$的类对(m_i, T_j),则把 m_i 合并到 T_j 中;否则创建一个新类 $T_k = \{m_i\}$,把 T_k 添加到 V 中;

(5)把 m_i 从 M 中删除,转向(2);

(6)把 V 中的元素作为结果(即新话题)输出。

通常,如果测试数据规模很大,通过相似度阈值来控制系统聚类的生成结果,得到的微类仍然很多。为降低算法的时间和空间复杂度,系统采用单步增量式聚类方法来处理微类。微类包含的信息比文档丰富,因此我们认为直接比较微类之间的相似度就可以判断两者的关系,而不会显著影响系统性能。这是一个增量聚类过程,即系统以已经存在的话题作为过滤条件,如果后续微类与已有话题相关,则把该微类添加到此话题中,否则,就创建一个只包含该微类的新话题。

在这个阶段,话题和微类之间采用单步比较,它包含两个关键操作:聚类比较和合并。这里同样也有多种策略可供选择:一种是和第一层聚类相同的方法,计算话题和微类的中心向量,并更新合并后的向量;另一种是单链法(single link),系统先计算任意两个微类之间的相似

度 $sim(c_i, c_j)$,设话题 C_i 中包含 n 个微类,微类 $m_i \in T_i$, $i = 1, \cdots, n$,则 T_i 和待处理微类 m 的相似度 $sim(T_i, m') = MAX(sim(m_i, m'))$。在这个过程中,话题的特征空间不参与运算,因此在微类和话题合并的时候,话题的中心向量也不需要更新。后续比较时只需要找到话题所含微类与待处理微类之间的最高相似度。两种方法都需要比较相似度和阈值 Θ,高于阈值就认为两者相关。系统同时实现了这两种策略,我们将在试验中对两者进行比较。

这一步完成之后,最终得到的话题即为新发现的话题。

3.6 多策略优化技术

除了提出面向任务的通用算法之外,我们还针对任务的特点提出一些有针对性的策略以优化性能,主要包括以下几个方面:

(1) 面向语种的优化策略

话题与报道的关系是一种语义关系,报道是否开始一个新话题,主要是判断它和以前的话题有没有内容上的关联。因此,文档和话题模型应能够表示各自的内容。我们认为,词包含的信息比字丰富而比组块少,但是目前组块分析的性能和分词有较大差距,因此用词表示文档更合适。对于中英两个语种,我们做了两种合理性假设:(1) 同一个词如果被标注成不同词性,那么它实际上拥有不同的语义,比如"展览/vn 活动/vn 向/p 公众/n 开放/v"和"潜艇/n 主要/b 活动/v 于/p 水下/s"中的"活动"词性不同,意义也不同,因此预处理阶段还需要进行词性标注;(2) 英文单词形态变化丰富,单复数名词、动词的不同时态实际上表示同一个意义,应作为相同的词处理,因此还应把名词和动词还原成词典形式,比如"goes"、"went"、"gone"都应还原为"go"。

(2) 特征选择

VSM 是一种通用的模型,但话题不仅仅是内容相关文档的简单集合,还需要对话题的内在特征进行分析。系统采用的是面向话题的 VSM,这主要体现在以下几个方面:(1) 除了内容词之外,话题通常还具有人物、时间、地点等命名实体,这些要素对于区分不同的话题起着很重要的作用,因此,算法对命名实体进行加权;(2) 对于向量中没有区分或表达能力的特征,系统将其排除,包括两层策略:一是用禁用词表过滤禁用词,并用禁用词性表把虚词(如连词、叹词、标点等)过滤掉;二是使用文档频率信息,把 $DF(w_i) \leq M$ 或 $DF(w_i) \geq N$ 的特征过滤掉(M, N 均为阈值);(3) 篇幅很短的文档通常视为数据噪音,系统将维数小于 MIN_DIM 的文档过滤掉。

(3) 命名实体归一化

经过加权的命名实体在权重上一般比其它词大,对效果的影响也就很大。然而命名实体的用词极为灵活,同一个实体在文字表述上可能有好多种,这样就造成命名实体不匹配的现象。例如,"李光耀"与"李资政","江苏省"与"江苏","U. S."与"U. S. A."等。为此,我们对常见的命名实体进行归一化,主要方法是对具有别称的命名实体建立一张别称表,遇到表中的命名实体就把它换成统一的称谓。对于地名,我们建立了一张后缀表,遇到地名时就把表中的后缀去掉。这些后缀如:"省"、"市"、"县"、"镇"、"村"、"特别行政区"等。对于没有出现人物全名的情况,例如,"李/nr1 资政/n",简单地把姓氏与前文中出现最近的全名关联起来。

3.7 话题发现系统

我们在这个思想的指导下实现了一个完整的系统,大体上分成三个模块:(1) 预处理模块执行如下操作:对于中文语料,系统对文本进行分词、词性标注和命名实体识别;对英文语料,系统对文本进行词性标注和词形还原。(2) 核心算法模块首先将文本分组,然后分别对各组执行系统聚类,得到的微类再进行增量式聚类。(3) 结果输出模块按照 TDT 评测要求的格式把结果输出到文件。为便于进行实验,算法中的大多数参数均可通过配置文件进行修改。

4 试验结果与分析

4.1 测试语料

已经完成的试验都针对 TDT4 语料进行处理。TDT4 共有 98,245 篇报道,其中中文语料 27,142 篇,分别来自于新华社、联合早报等新闻机构,英文语料有 28,390 篇,分别来自 NYT、CNN、VOA 等新闻机构,其余为非英文报道经机器翻译成英文之后的结果。时间跨度为 2000 年 10 月至 2001 年 1 月。语料形式基本是原始的新闻文本或语音转录得到的电视或广播新闻文本。

4.2 评测标准

为了对不同的系统进行量化比较,TDT 会议制订了一套评测规范。每一个参评系统的性能是由误报率和漏报率加权求和的结果进行衡量的,其计算公式是:

$$C_{Det} = C_{Miss} \cdot P_{Miss} \cdot P_{target} + C_{FA} \cdot P_{FA} \cdot P_{norr\,target}$$

其中 C_{Det} 是系统的性能评测指标,称为检测错误代价,这个值越低越好。C_{Miss} 和 C_{FA} 分别是漏报和误报的代价;P_{Miss} 和 P_{FA} 分别是漏报和误报的条件概率;P_{target} 是目标话题的先验概率。C_{Miss}、C_{FA} 和 P_{target} 都是预设值,用来调节漏报率和误报率在评测结果中所占比重。检测代价通常被归一化为 0 和 1 之间的一个值:$(C_{Det})_{Norm} = C_{Det}/\min(C_{Miss} \cdot P_{target}, C_{FA} \cdot P_{norr\,target})$。

TDT 会议采用话题加权分数作为系统最终的评测结果。目前跨语言话题发现的最好检测代价约为 0.3,单独对英文测试约为 0.2。我们利用 NIST 在网上公开的 TDT 评分工具对系统的输出结果进行评测,评测公式的参数取值为:$C_{miss} = 1$;$C_{false} = 0.1$;$P_{target} = 0.02$;$P_{norr\,target} = 0.98$。

4.3 比较结果

我们设计了一个简单算法作为基准,把它与系统在同一个语料上进行对比测试。该基准算法在其他方面均与系统算法一致,但聚类时只采用了单层法。系统中很多参数对于最终的性能起着重要作用。我们在 TDT4 语料上进行多次测试,通过反复比较,系统中的一些经验参数设置如下:时间衰减因子 $m = 180$;命名实体加权系数 $P_{NE} = 3$;短文档过滤阈值 $MIN\_DIM = 10$。

目前的试验结果表明阈值在区间[0.2,0.3]内性能最佳,如表 1 所示。

从表中的测试结果可看出:

(1) 采用分治多层聚类且聚类比较采用中心向量的算法在阈值取 0.23 时性能最佳;

(2) 采用分治多层聚类的效果一般都比单层聚类有较大提高;

(3) 聚类比较采用中心向量法时性能一般都比单链法好。

表 1 测试结果(表中的值为 $(C_{Det})_{Norm}$)

阈值	0.20	0.21	0.22	0.23	0.24	0.25	0.26	0.27	0.28	0.29
基准	0.3129	0.3154	0.3097	0.3048	0.3006	0.2980	0.2996	0.2960	0.2980	0.3015
中心向量	0.3026	0.3005	0.3013	0.2906	0.2926	0.2968	0.3051	0.3051	0.3171	0.3324
DMCMO + 单链	0.2172	0.2210	0.2150	0.2103	0.2094	0.2095	0.2083	0.2126	0.2162	0.2175
DMCMO + 中心向量	0.2098	0.2087	0.2068	0.2028	0.2117	0.2148	0.2133	0.2133	0.2221	0.2220

4.4 分析

我们考察了不分组、按时序分组与不按时序分组对检测代价的影响,结果如图 2 所示。图 2 中,按照从上到下的次序,曲线 1 表示不对文档分组,一开始就对全部语料进行单遍聚类的结果,横坐标是不同的聚类相似度阈值 θ。曲线 2 是采用时序分组的结果,分组时间间隔为 7 天。在曲线 3 的实验中,我们仍旧把文档进行分组,组内优先聚类,与曲线 2 不同的是,文档按

时间均匀打散,即每组内的文档在时间上并不相邻。结果表明,基于时序分组的策略使检测代价降低了近 30 %;当组内文档时间不相邻时,分组策略对效果提高不明显。我们又考察了在时序分组的条件下,不同的分组时间间隔对检测代价的影响,结果分布如图 3 所示。通过实验可以看出,时序分组策略能有效降低系统的检测代价,主要原因是它把时间相近的文档优先聚合在一起,提高了属于同一话题的文档的合并机会。

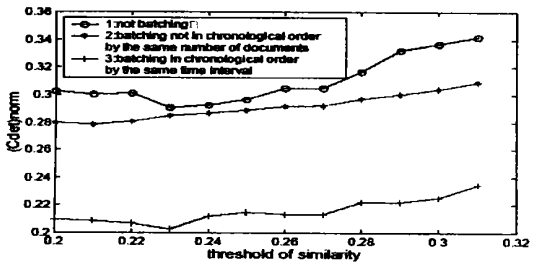

图 1　分组策略的影响

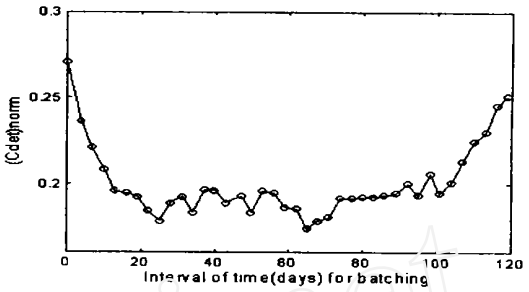

图 2　不同时间间隔的影响

单独对中文语料测试取得了 $(C_{Det})_{Norm} \approx 0.2$ 的成绩,已追平目前对英文语料单独测试取得的最好成绩,表明针对语种的优化策略取得了一定效果。而先把文档聚成微类,再对微类进行聚类的方法则在效率和效果上都比直接对文档聚类要好,因为基于时间的数据分组保证了同一组内文档的时间相关性,同时微类包含的信息比文档更丰富,使得基于内容的聚类结果更加准确。用中心向量来表示话题从效率而言显然比用话题的文档空间更有效,而准确率并没有明显差别。

5　结论和进一步工作

本文系统地介绍了话题发现与跟踪技术的由来与发展,总结了目前主要的话题发现算法框架,针对已有方法的不足,我们提出了多策略优化的分治多层聚类算法,针对中英文语料分别进行预处理,并依据时间间隔把文档进行分组,在每一组内进行基于中心向量的系统聚类法,得到的微类再进行增量式聚类。该算法结合话题发现任务的特点,把话题的一些关键性特征(如命名实体、时间等)抽取出来进行处理。通过在 TDT4 中文语料上进行测试,该算法取得了很好的效果,从测试成绩上看,其性能已追平目前最好的话题发现算法在英文语料上的测试成绩。

但总体来看,分治多层聚类算法仍源于传统的信息检索技术,对于话题特征的发掘仍远远不够。此外,该算法在层次话题发现任务中的性能仍有待检验。我们下一步的研究工作是在文档建模中借助 WordNet 等知识库引入语义扩展,并通过分析文档内容改进聚类策略,从而进一步提高算法的效率和性能。

参　考　文　献:

[1]　骆卫华,刘群,程学旗. 话题检测与跟踪技术的发展与研究[A]. 孙茂松,陈群秀. 全国计算语言学联合学术会议(JSCL - 2003)论文集[C]. 北京:清华大学出版社,2003,560 - 566.

[2]　Jonathan G. Fiscus, George R. Doddington. Topic Detection and Tracking Evaluation Overview[A]. In: James Allan. Topic Detection and Tracking, Event-based Information Organization[C]. Norwell: Kluwer Academic Publishers, 2002, 17 - 31.

[3] Y. Yang, T. Pierce, J. Carbonell. A Study on Retrospective and Online Event Detection[A]. In: W. Bruce Croft, Alistair Moffat, C.J. van Rijsbergen, et al. Proceedings of the 21th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)[C]. New York: ACM Press, 1998, 28 - 36.

[4] Brants, T., Chen, F. R., Farahat, A. O. A system for new event detection[A]. In: Charles Clarke, et al. Proceedings of SIGIR 2003, the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval[C]. New York: ACM Press, 2003, 330 - 337.

[5] Juha Makkonen, Helena Ahonen-Myka, and Marko Salmenkivi. Simple Semantics in Topic Detection and Tracking [J]. Information Retrieval, 2004, 7 (3 - 4): 347 - 368.

[6] Y. Yang, J. Carbonell, C. Jin. Topic-conditioned novelty detection[A]. In: Hand D, et al. Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining[C]. New York: ACM Press, 2002, 688 - 693.

[消息]

中文语言资源联盟

在国家高科技研究规划发展项目(863)和国家重点基础研究发展规划项目(973)以及其他项目的支持下,由中国中文信息学会语言资源建设和管理工作委员会发起,由中文语言(包括文本、语音、文字等)资源建设和管理领域的科技工作者自愿组成了中文语言资源联盟(英文译名 Chinese Linguistic Data Consortium,缩写为 CLDC),该联盟是学术性、公益性、非盈利性的社会团体。本团体隶属于中国中文信息学会,接受中国中文信息学会语音资源建设和管理工作委员会的业务指导和监督管理。

CLDC 以代表中文信息处理国际水平的、通用的汉语语言语音资源库为目标,建设和收集了具有完整性、权威性、系统性的开放式中文语言资源,涵盖中文信息处理各个层面上所需要的语言语音资源,包括词典、各种语音语言语料库、工具等。为汉语语言信息处理等基础研究和应用开发提供支持,并服务于教育、科研、政府研究部门和工业技术开发等领域。

目前,联盟已经拥有的资源库超过四十余种。本资源主要用于:机器翻译、句法语义语法、语音识别、语音合成、手写识别、文本分类等方面的技术研究和开发。在每一种资源中,将包含有丰富的数据信息,如,资源简介、标注规范、技术文档、资源用途、样例下载和数据库的基本结构等。详情请登陆我们的网站:http://www.chineseldc.org,在提供丰富多样的资源数据的同时,我们诚恳的邀请您加入中文语言资源联盟,您会成为会员(在我们网站的"会员申请"中申请),并享受我们为您提供的优质会员服务和购买资源折扣。

我们期待着您的加入,并希望您一如既往的支持我们,使我们不断进步!

地　　址:北京市海淀区中关村东路 95 号,自动化大厦 1013 室,P.O. X2728,
　　中文语言资源联盟　　邮编:100080
电　　话:010 - 82614519　戴　瑛　　010 - 62565533 - 9601　雷　俊
传　　真:010 - 62551993　　E - mail:service@chineseldc.org