# 基于 PLSI 的标签聚类研究*

吴志媛，钱雪忠
（江南大学 物联网工程学院，江苏 无锡 214122）

**摘　要**：针对现有的大众分类中标签模糊导致影响用户搜索效率的问题，使用概率潜在语义索引（probabilistic latent semantic indexing，PLSI）模型对标签进行潜在语义分析，经回火期望最大化（tempered exception maximization，TEM）算法训练得到在潜在语义下的条件概率，生成概率向量；在此基础上，提出凝聚式层次 $k$ 中心点（hierarchical agglomerative K-mediods，HAK-mediods）聚类算法对概率向量进行聚类，并进行了相关对比实验。实验结果表明，HAK-mediods 算法的聚类效果要好于传统的聚类算法，从而验证了该算法的可行性和有效性。

**关键词**：大众分类；概率潜在语义索引；语义标签；回火期望最大化算法；凝聚式层次 $k$ 中心点聚类
**中图分类号**：TP391　　**文献标志码**：A　　**文章编号**：1001-3695(2013)05-1316-04
doi:10.3969/j.issn.1001-3695.2013.05.009

# Tag clustering research based on PLSI

WU Zhi-yuan，QIAN Xue-zhong
（*School of Internet of Things Engineering*，*Jiangnan University*，*Wuxi Jiangsu* 214122，*China*）

**Abstract**：Ambiguity of tag may influence users' efficiency in Folksonomy systems. PLSI model was used to analyze semantic tags，through the training of TEM algorithm to get the conditional probability of latent variables，and to generate probability vectors. On that basis，this paper proposed HAK-mediods clustering algorithm to cluster probability vector. The experimental results turn out that HAK-mediods clustering algorithm enhances the clustering performance than traditional clustering algorithm.

**Key words**：Folksonomy；PLSI；semantic tags；TEM algorithm；HAK-mediods clustering

## 0　引言

随着 Internet 技术的不断发展，互联网经历了以信息提供商为中心的 Web 1.0 向以用户为中心的 Web 2.0 转变。在 Web 2.0 时代，信息技术的发展使得网络用户广泛参与到信息资源描述和组织中成为可能。大众分类（Folksonomy）[1] 是典型的 Web 2.0 系统，允许所有互联网用户为网络资源添加标签。Folksonomy 是 VanderWal 和 Smith 于 2004 年首先提出，其含义是由大众的一致意见而产生的基于用户的分类体系。此分类法根据用户个人的使用习惯，以自定义的词对网络资源进行标注和分类。这些自定义的词称为标签（tag），也就是指描述信息资源的字、词或者短语。但是，由于用户的文化程度和兴趣爱好存在很大差异，所添加的标签也不受系统的控制，所以不同的用户会使用不同的标签进行标注，导致对描述信息资源的标签在一定程度上存在着描述精确度不高和组织混乱等问题，从而影响大众标注系统中对网络信息资源的分类、组织和检索。

目前，国外对于大众分类中的标签聚类问题的研究已从理论研究向实际应用过渡，且更加注重在潜在语义层面上的聚类研究。Abbasi 等人[2] 建立了一个 T-ORG（tag-organizer）系统，用来在语义层面上将标签按层次结构进行组织。Begeman 等人[3] 提出了自动标签聚类的方法来改善自由分类法的检索和浏览。国内也有专家提出了一些有关标签聚类的算法。武汉大学的曹高辉等人[4] 利用凝聚式层次聚类算法对标签聚类进行研究，利用相关标签的权重计算标签之间的相关度，从而实现标签的聚类。大众标注与生俱有的不足是自由、不受控制，非专业路线的结果就是标准、规范缺乏，这将引发共享、查找以及浏览上的问题。因此，人们提出基于语义标注的大众标注，但是，由于标签过于个性化而产生的歧义，平面化的、没有等级层次的划分，相对传统分类法不够严谨、缺乏准确度，都可能制约 Folksonomy 的发展。

本文在上述研究的基础上，引入 PLSI 模型来挖掘页面资源与标签间的潜在语义关系，并结合提出的 HAK-mediods 聚类算法对潜在语义下的标签进行相似度聚类，最后得到潜在语义下的聚合标签集，在一定程度上使得形式不同、意义相同的标签根据用户的需要被其中的一个典型标签所替代，从而实现对用户标签的重新组织，为用户提供更好的标签导航、浏览机制。

## 1　相关概念

### 1.1　大众分类与标签

大众分类是一个创造词，是由 Folk（或 Folks）与 Taxonomy 组合而成。与传统结构严谨的登记体系分类法、庞大的文献分类法以及网站预设的信息分类法不同，大众分类法让用户参与

**收稿日期**：2012-08-27；**修回日期**：2012-10-19　　**基金项目**：国家自然科学基金资助项目（61103129）；江苏省科技支撑计划资助项目（BE2009009）

**作者简介**：吴志媛（1989-），女，江苏淮安人，硕士研究生，主要研究方向为数据挖掘（wuzhiyuan0613@163.com）；钱雪忠（1967-），男，江苏无锡人，副教授，硕导，主要研究方向为数据库技术、数据挖掘、网络安全等.

信息资源的描述和组织,允许用户自由使用标签对文章、图片、视频、声音等各种对象进行描述,并利用这些用户标签完成信息资源的分类、组织、检索。标签是一种由用户产生的元数据,区别于以往由专家或网站作者产生的元数据,它能直接、迅速地反映用户的词汇和需求及其变化。

### 1.2　PLSI 模型

为了挖掘大众分类系统中标签间的潜在语义关系,本文引用概率潜在语义索引(PLSI)[5]模型来实现。PLSI 模型是 Hofmann 于 1999 年提出的,从概率统计的角度对潜在语义索引(latent semantic index, LSI)[6,7]进行了全新的诠释。PLSI 模型的变量包括给定 $m$ 个标签的模型 $D=\{d_1,d_2,\cdots,d_m\}$、$n$ 个页面 $T=\{t_1,t_2,\cdots,t_n\}$ 和潜在语义变量 $Z=\{z_1,z_2,\cdots,z_k\}$。对于给定的页面 $t_i$ 和标签 $d_j$,使用联合概率来表示页面与标签之间的潜在关系。

$$p(t_i,d_j)=p(t_i)p(d_j|t_i) \tag{1}$$

$$p(d_j|t_i)=\sum_{k=1}^{k}p(d_j|z_k)p(z_k|t_i) \tag{2}$$

式(2)中的 $p(z_k|t_i)$ 是潜在语义在页面上的分布概率,也可以解释为页面对潜在语义的贡献度,通过对 $p(z_k|t_i)$ 排序可以得到潜在语义的一个直观的页面表示;$p(d_i|z_k)$ 表示标签在潜在语义条件下的条件概率。PLSI 模型使用期望最大化(exception maximization, EM)[5]算法对模型进行拟合。使用随机数初始化之后,交替实施 E 和 M 步骤进行迭代计算。在 E 步骤中,计算任何一个 $(t_i,d_j)$ 对产生潜在语义 $z_k$ 的先验概率:

$$p(t_i,d_j)=\frac{p(d_j|z_k)p(z_k|t_i)}{\sum_{k=1}^{K}p(d_j|z_k)p(z_k|t_i)} \tag{3}$$

在 M 步骤中,利用 E 步中得到的期望,使用式(4)和(5)来最大化当前的参数估计。

$$p(z_k|t_i)=\frac{\sum_{j=1}^{M}n(t_i,d_j)p(z_k|t_i,d_j)}{n(t_i)} \tag{4}$$

$$p(d_j|z_k)=\frac{\sum_{i=1}^{N}n(t_i,d_j)p(z_k|t_i,d_j)}{\sum_{m=1}^{M}\sum_{i=1}^{N}n(t_i,d_m)p(z_k|t_i,d_m)} \tag{5}$$

算法不停迭代在式(4)与(5)之间,直到满足收敛条件时停止。收敛条件为似然函数 $L$ 的期望值 $E(L)$ 增加量取最小值。

$$E(L)=\sum_{j=1}^{M}\sum_{i=1}^{N}[n(t_i,d_j)\sum_{k=1}^{K}p(z_k|t_i,d_j)\log(p(d_j|z_k)p(z_k|t_i))] \tag{6}$$

为了避免过度拟合,PLSI 模型利用回火期望最大化(TEM)[5]算法最大化被观测数据的概率,即保持 M 步骤不变,引入控制参数 $\beta$,修改 E 步骤。

$$p_\beta(z_k|t_i)=\frac{[p(d_j|z_k)p(z_k|t_i)]^\beta}{\sum_{k=1}^{K}[p(d_j|z_k)p(z_k|t_i)]^\beta} \tag{7}$$

式中:$\beta(0<\beta<1)$ 为控制参数。该算法的复杂度为 $O(mnk)$,$n$ 为网页总数,$m$ 为标签总数,$k$ 表示潜在语义个数。

### 1.3　“页面—标签”矩阵 *TD* 的权重

在向量空间模型中,一个词的权重表现着这个词在文档中的重要性,通过统计得到每个标签在各个页面中的出现频率和包含某个标签的文档个数,然后利用(term fequency-inverse document frequency, TF-IDF)[7]公式得到“页面—标签”矩阵 *TD* 的每个权重。

$$a_{ij}=\frac{\log(f_{ij}+0.1d)\times\log(n/n_i)}{\sqrt{\sum_{p=1}^{m}[\log(f_{pj}+0.1)\times\log(n/n_p)]}} \tag{8}$$

其中:$a_{ij}$ 表示标签 $i$ 在页面 $j$ 中的权重;$f_{ij}$ 表示标签 $i$ 在页面 $j$ 中的出现频率;$n$ 表示页面集的页面总数;$m$ 表示页面集的标签总数,$n_i$ 表示标签 $i$ 的页面频数。此 TF-IDF 公式将最能表达文档内容的关键词提取出来,除了能够降低矩阵的维数外,还能够避免非关键词对统计模型的干扰。

## 2　基于 PLSI 的文本标签聚类

### 2.1　标签聚类的思想

基于 PLSI 的标签聚类的基本思想是:针对大众分类系统中的标签模糊导致影响用户搜索效率的问题,运用概率潜在语义索引技术来消除标签的同一性及一词多义现象,同时生成标签及页面的潜在语义,然后通过两两标签间的语义相似度比较找出相似度很高的语义标签;在此基础上对相关的语义标签加以 HAK-mediods 聚类分析,得到潜在语义下的聚类标签集,更好地满足用户检索的要求。

### 2.2　标签的潜在语义分析

许多研究发现,标签通常是语义相关的。如果它们多次用来标记相同或相关的页面资源,那么使用这些语义相关标签的用户可能有相似的兴趣;如果某些页面资源被许多有相似兴趣的用户标注,那么这些页面资源在语义上也是相关的,所以标签和网页资源在用户兴趣的语义上是相互关联的。这种语义关联关系隐含在社会化标注服务的网页资源和标签共同出现的频率中。

受 PLSI 模型研究的启发,通过概率关系可以发现网页与标签间隐含的语义关系。利用 PLSI 模型,得到标签 $d_i$ 在潜在语义 $z_k$ 已知条件下的概率 $p(d_i|z_k)$,构建标签的概率向量 $dz_i=\{p_{i,1},p_{i,2},\cdots,p_{i,k}\}$,其中,$i=\{1,2,\cdots,m\}$,$p_{i,k}$ 为标签 $d_j$ 在潜在语义 $z_k$ 已知条件下的条件概率 $p(d_i|z_k)$。此条件概率反映了标签和潜在语义的相关程度。计算同一语义下两标签的相似度,如式(9)所示。

$$\mathrm{sim}(dz_i,dz_j)=\frac{\sum_{m=1}^{k}p_{i,m}p_{j,m}}{\sqrt{\sum_{m=1}^{k}p_{i,m}^2}\times\sqrt{\sum_{m=1}^{k}p_{j,m}^2}} \tag{9}$$

### 2.3　聚类算法描述

#### 2.3.1　K-mediods[8] 算法

K-mediods 算法选用聚类中位置最靠近中心的点作为参考点,从而消除了 K-means 算法因采用质心作为参考点而导致对孤立点敏感的缺点。K-mediods 算法步骤如下:

a)从 $n$ 个数据对象中任意选择 $k$ 个对象($O_1,O_2,\cdots,O_i,\cdots,O_k$)作为初始聚类中心(medoids)。

b)将余下的对象划分到各个类中(根据与 medoids 距离最小或相似度最大的原则)。

c)对于每个类 $O_i$ 中,顺序选取一个非中心对象 $O_r$,计算用 $O_r$ 替代 $O_i$ 后的消耗 $E(E=\sum_1^t\sum_{O_r\in O_i}|O_r-O_i|)$。选择 $E$ 最小的那个 $O_r$ 来代替 $O_i$ 成为新的聚类中心,然后再转到 b)。

d)循环直到 $k$ 个聚类中心(medoids)都固定下来不再变化。

在各种不同的 K-mediods 算法中,较为常见的有 PAM

(partitioning around medoids)、CLARA(clustering large application)、CLARANS (clustering large application based upon randomized search)等算法。

### 2.3.2 HAK-mediods 聚类算法

K-mediods 算法虽能有效地处理数据集,但是该算法的限制在于:必须制定参数 $k$,即初始聚类中心(medoids)数。另外,初始的 $k$ 个中心点的选择也对聚类的结果有比较大的影响,很容易陷入局部最优解。

本文在分析了现有聚类算法和实际应用环境后,提出了将基于距离的平面划分法 K-mediods 聚类算法与凝聚式层次(hierarchical agglomerative, HA)[9,10]聚类算法相结合的 HAK-mediods 聚类算法。凝聚式层次聚类是一种常用的文本层次聚类方法,它自底向上分析目标文档,每个目标文档最初被看成一个最小的聚类,两个相似度最大的聚类被合并为一个最大的聚类,这个过程一直持续到所有文档合并为一个聚类或者达到一个终止条件为止。HAK-mediods 算法首先利用凝聚式层次聚类算法进行初始聚类,确定初始聚类中心和相应的值,然后用 K-mediods 算法进行聚类分析。HAK-mediods 聚类算法大大提高了 K-mediods 聚类算法的性能。

算法 HAK-mediods 聚类

输入:"页面—标签"矩阵 $TD$、相似度极限 $\mu$ 和潜在语义数 $k$。

输出:聚类结果 $C'=\{C_1',C_2',\cdots,C_t'\}$,其中 $t$ 表示聚类数,$C_i'$ 包含访问页面的标签总数。

a)对原始的页面资源进行数据预处理,构造出"页面—标签"矩阵 $TD$,计算矩阵中各个权值。

b)利用 PLSI 模型训练矩阵 $TD$,得到标签向量 $dz_i$。

c)每一行对象 $dz_i$(即标签)看做是一个具有单个成员的聚类 $C_i=\{dz_i\}$,这些聚类构成了向量集合的一个聚类中心 $C=\{C_1,C_2,\cdots,C_n\}$。

d)计算每对类$(C_i,C_j)$之间的相似度 $\mathrm{sim}(C_i,C_j)$。

e)选取具有最大相似度的值 $\max=\max\{\mathrm{sim}(C_i,C_j)\}$。如果 $\max\geqslant\mu$,则将 $C_i$ 和 $C_j$ 合并为一个新的类 $C_k=C_i\cup C_j$,从而得到一个新的聚类中心 $C=\{C_1,C_2,\cdots,C_{n-1}\}$,然后重复步骤 d)e);如果 $\max<\mu$,则凝聚算法结束,得到具有 $t$ 个子类的聚类中心 $C'=\{C_1',C_2',\cdots,C_t'\}$。

f)对于每一个向量对象 $dz_i$,依次计算它与各个聚类中心 $C'=\{C_1',C_2',\cdots,C_t'\}$ 的相似度 $\mathrm{sim}(dz_i,C_i')$。

g)选择具有最大相似度的聚类中心 $C_i'$,将 $dz_i$ 归入以 $C_i'$ 为聚类中心的类中。

h)计算新的聚类中心。对于每个类$(C_i)$中,顺序选取一个非中心对象 $C_r$,计算用 $C_r$ 代替 $C_i'$ 后的 $E$ 值$(E=\sum_1^t\sum_{C_r\in C_I'}|C_r-C_i'|)$,选择 $E$ 值最小的那个 $C_r$ 来代替 $C_i'$ 作为新的聚类中心。

i)重复步骤 f)~h),直到所有的数据对象计算完毕,所有的聚类中心点均不再变化。

## 3 实验及结果评价

### 3.1 实验数据

本文使用了两个 Web 页面数据集和两个搜索引擎结果数据集。表 1 描述了这些数据集的特点。数据集 DBS1 是人工收集美味网[1]并标注的 300 篇 Web 页面;数据集 DBS2 是来自于 reuters21578 数据集的 340 篇 Web 网页新闻;数据集 DBS3 和 DBS4 是从 Web 搜索引擎获取的、专门用于评估信息获取技术相关算法的数据集,数据集 DBS3 包含 500 篇文档,DBS4 包含 500 篇文档,通过人工标注均分为 5 个类,每类 100 个文档,平均词数比较少。

表 1 实验数据集

| 数据集 | 名称 | 类型 | 文档数 | 分类数 | 词数 |
|---|---|---|---|---|---|
| DBS1 | delicious | HTML | 300 | 4 | 850 |
| DBS2 | reuters21578 | HTML | 340 | 5 | 1 240 |
| DBS3 | AMBIENT | TEXT | 500 | 5 | 1 800 |
| DBS4 | AMBIENT | TEXT | 500 | 5 | 2 540 |

### 3.2 实验结果分析

#### 3.2.1 聚类评价指标

F-measure 是最常用的评价聚类算法效果的指标。它综合了信息检索领域常用的 precision(准确率)[9]和 recall(召回率)[11]指标。设 $C=\{C_1,C_2,\cdots,C_k\}$ 是数据集 $D$ 的 $N$ 个文档的一个聚类结果,设 $C^*=\{C_1^*,C_2^*,\cdots,C_l^*\}$ 表示数据集 $D$ 的正确分类,则第 $j$ 个聚类对于第 $i$ 个分类的召回率和准确度分别定义为

$$\mathrm{precision}(i,j)=\frac{|C_j\cap C_i^*|}{|C_j|},\mathrm{recall}(i,j)=\frac{|C_j\cap C_i^*|}{|C_i^*|}$$

F-measure 集成了准确率和召回率,定义为

$$F(i,j)=\frac{2\times\mathrm{precision}(i,j)\times\mathrm{recall}(i,j)}{\mathrm{precision}(i,j)+\mathrm{recall}(i,j)}$$

$F$ 值越大,则聚类效果越好。

#### 3.2.2 聚类效果分析

为了得到标签矩阵 $p(d|z)$,实验中使用 PLSI 模型对数据集在 MATLAB 数学软件上进行训练,首先利用 TF-IDF 方法得到"页面—标签"矩阵作为算法输入初始值。在实验中以数据集 DBS1 为例,选取四个类作为训练数据,然后对于每个选取的数据,使用不同参数控制的 PLSI 模型进行训练,得到 $p(d|z)$ 和 $p(z)$。表 2 为 TEM 算法第一次迭代后的结果(只列举了结果的部分数据)。表 3 为目标函数取最优值时(即迭代结束时,此时迭代次数为 53)的结果(只列举了结果的部分数据)。观察表 2 和 3 中的每行数据可以看出,表 3 的数据更容易进行类别划分。

表 2 第一次迭代下的矩阵 $p(d|z)$ 和 $p(z)$

| iteration 1: | | | | |
|---|---|---|---|---|
| $p(z)$ = | 0.2430 | 0.2266 | 0.2807 | 0.2497 |
| $p(d\|z)$ = | 0.0222 | 0.0418 | 0.0576 | 0.0711 |
| | 0.0834 | 0.1589 | 0.0882 | 0.0662 |
| | 0.0856 | 0.0596 | 0.0774 | 0.0686 |
| | 0.1061 | 0.0854 | 0.0696 | 0.0341 |
| | 0.1042 | 0.1203 | 0.1238 | 0.1386 |
| | 0.0630 | 0.0665 | 0.0969 | 0.0624 |
| | 0.0724 | 0.1096 | 0.1166 | 0.0898 |
| | 0.0863 | 0.0345 | 0.0629 | 0.1070 |
| | 0.0762 | 0.0977 | 0.0924 | 0.0264 |
| | 0.1212 | 0.1222 | 0.0800 | 0.0720 |
| | 0.0442 | 0.0412 | 0.0785 | 0.1244 |

表 3 目标函数最优下的矩阵 $p(d|z)$ 和 $p(z)$

| iteration 53: | | | | |
|---|---|---|---|---|
| $p(z)$ = | 0.2839 | 0.1703 | 0.2811 | 0.2647 |
| $p(d\|z)$ = | 0 | 0.0000 | 0.0868 | 0.0921 |
| | 0.0000 | 0.5702 | 0.0017 | 0.0000 |
| | 0.1284 | 0.0000 | 0.1306 | 0.0000 |
| | 0.2577 | 0.0000 | 0.0000 | 0.0000 |
| | 0.0985 | 0.0000 | 0 | 0.3551 |
| | 0.0000 | 0.0000 | 0.2603 | 0 |
| | 0.0000 | 0.0000 | 0.3471 | 0.0000 |
| | 0.0000 | 0.0000 | 0 | 0.2764 |
| | 0.0859 | 0.2865 | 0.0000 | 0.0000 |
| | 0.1718 | 0.1433 | 0.0000 | 0.0921 |
| | 0.0000 | 0.0000 | 0.1735 | 0.0921 |

将 PLSI 模型与 K-mediods 聚类、HA 聚类及提出的 HAK-mediods 聚类三种算法结合,根据模型训练得到的最优 $p(d|z)$ 值进行聚类,然后通过比较聚类得到的标签和原始标签的最大匹配来计算准确率和召回率。对于每次选取的数据,分别重复五次实验,取平均值记录,所得数据如表 4 所示。表中加粗标记的数据分别为准确率和召回率取的最大值。

表 4　不同聚类算法的准确率和召回率

| $\mu$ 值 | precision | | | recall | | |
|---|---|---|---|---|---|---|
| | K-mediods | HA | HAK-mediods | K-mediods | HA | HAK-mediods |
| 0.35 | 59.51 | 62.94 | 67.83 | 63.48 | 65.35 | 70.13 |
| 0.45 | 66.81 | 70.53 | 74.36 | 67.81 | 71.83 | 76.89 |
| 0.55 | 75.67 | 79.64 | 83.61 | 76.61 | 79.67 | 84.63 |
| 0.65 | 79.58 | 83.51 | 88.96 | 80.72 | 85.14 | 90.17 |
| 0.75 | 82.14 | 78.12 | 80.22 | 83.22 | 82.32 | 85.63 |
| 0.85 | 72.12 | 73.68 | 71.61 | 73.17 | 75.78 | 77.11 |
| 0.95 | 62.48 | 60.31 | 64.33 | 64.28 | 63.38 | 68.64 |

图 1 和 2 展示了随着控制参数 $\mu$ 变化的三个聚类算法的准确率和召回率。阈值对聚类准确率和召回率有一定的影响,因为阈值大小直接影响着页面中标签的类别归属。从图 1 和 2 可以看出,HAK-mediods 聚类算法在阈值控制得当的情况下有着比 K-mediods 聚类和 HA 聚类更好的表现。图中的准确率和召回率随着阈值 $\mu$ 的增加首先上升,到达极值后开始逐渐下降,因此参数 $\mu$ 需要小心地选择。表 4 的数据显示,HAK-mediods聚类取得的最优结果与 K-mediods、HA 聚类相比,其准确率分别提高了 6.38% 和 5.45%,因而得到了更稳定的聚类效果。

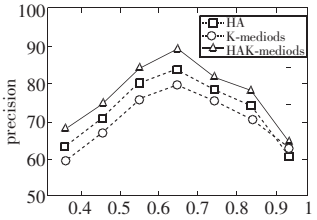

图 1　三种聚类算法的准确率

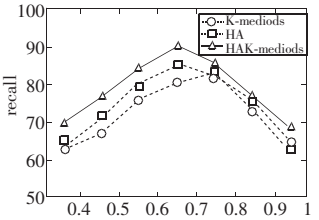

图 2　三种聚类算法的召回率

图 3 显示了三种聚类算法在四个数据集上的 F-measure 指标。可以看出,HA 聚类算法速度快,聚类质量不高;K-mediods 聚类算法的初始聚类中心是随机选取的,聚类结果不太稳定;HAK-mediods 聚类算法利用 HA 聚类获得初始聚类中心,能够获得稳定的、较高的聚类质量。

### 3.2.3　时间消耗分析

本文提出的聚类方法的时间消耗主要包括 PLSI 模型训练时间和聚类算法执行时间两个部分。聚类算法的时间消耗如表 5 所示,分别列出了四个数据集上 HAK-mediods 算法总时间和各算法执行的时间。计算每个数据集上运行的总时间和 HAK-mediods 聚类算法时间的差,发现 PLSI 模型训练需要的时间很少,主要时间消耗为聚类算法;而且 HAK-mediods 聚类比 K-mediods、HA 聚类需要更多的时间,约为后两者之和。

表 5　三种聚类算法在四个数据集上的运行时间

| 数据集 | 总时间/ms | K-mediods | HA | HAK-mediods |
|---|---|---|---|---|
| DBS1 | 1 344 | 539 | 784 | 1 319 |
| DBS2 | 1 543 | 686 | 845 | 1 471 |
| DBS3 | 2 513 | 872 | 1 441 | 2 387 |
| DBS4 | 2 556 | 841 | 1 529 | 2 468 |

为了进一步说明聚类时间与标签数目的关系,在数据集 DBS4 中做了实验,结果如图 4 所示。实验结果表明,随着标签数目的增加,聚类算法的时间呈线性增长。

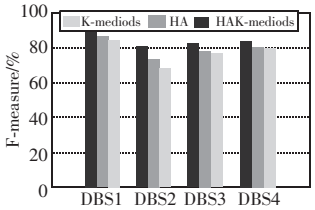

图 3　三种聚类算法在四个数据集的 $F$ 指标

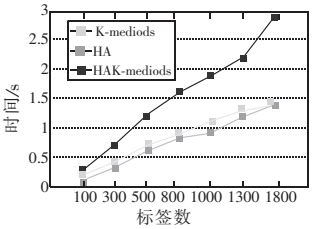

图 4　聚类时间与标签数的关系

## 4　结束语

概率潜在语义索引模型具有坚实的数学理论基础,较好地模拟了一个概率潜在语义空间。本文将概率潜在语义索引模型应用于大众分类中的标签聚类,首先对页面资源进行数据预处理,构建“页面—标签”矩阵;然后利用概率潜在语义索引(PLSI)算法对标签进行潜在语义分析,得到潜在语义下标签的条件概率,生成概率向量;最后利用本文提出的 HAK-mediods聚类算法对标签的概率向量进行聚类,得到聚类的标签集,并在实验中与 K-mediods 聚类、HA 聚类算法的聚类效果作比较。

实验结果表明,HAK-mediods 聚类算法较好地提高了聚类性能,能够更加有效地为网络用户提供服务。但 HAK-mediods 算法的聚类效果受到参数的一定影响,因此,下一步研究的重点是细化和协调各个参数值并进行更详细的实验分析,以进一步提高该算法的聚类性能。

**参考文献:**

[1] 高彧,郭利伟. Web 2.0 环境下走向语义标注的 Folksonomy 研究[J]. 中国科技信息,2009(14):112-123.

[2] ABBASI R, STAAB S, CIMIANO P. Organizing resource on tagging systems using T-ORG [C]//Proc of Workshop on Bridging the Gap between Semantic Web and Web 2.0. 2007:97-100.

[3] BEGEMAN G, KELLER P, SMADJIA F. Automated tag clustering: improving search and exploration in the tag space[C]//Proc of Collaborative Web Tagging Workshop at World Wide Web. 2006:22-26.

[4] 曹高辉,焦玉英,成全. 基于凝聚式层次聚类算法的标签聚类研究[J]. 现代图书情报技术,2008,51(4):23-27.

[5] 王奕. 基于概率潜在语义分析的中文文本分类研究[J]. 甘肃联合大学学报:自然科学版,2011,25(4):75-78.

[6] 曾广平. 贝叶斯概率 LSA 模型权重更新算法[J]. 计算机工程与应用,2009, 45(21):88-90.

[7] 熊忠阳,暴自强,李智星,等. 结合 LSA 的中文谱聚类算法研究[J]. 计算机应用研究,2010,27(3):917-918.

[8] 孟颖,罗可,刘建华,等. 一种基于差分演化的 K-medoids 聚类算法[J]. 计算机应用研究,2012,29(5):1651-1653.

[9] 刘一鸣,张化祥. 引入信息增益的层次聚类算法[J]. 计算机工程与应用,2012,48(1):142-144.

[10] 贾瑞玉,耿锦威,宁再早,等. 基于代表点的快速聚类算法[J]. 计算机工程与应用,2010,46(33):121-123,126.

[11] 李良俊,张斌,杨明. 基于 LSA 降维的 KNN 文本分类算法[J]. 东北师范大学学报:自然科学版,2007,39(2):33-36.