中图法分类号: TP391.3 **文献标志码**: A **文章编号**: 1006-8961(2010)11-1635-09
检索信息: 李晓燕, 陈刚, 寿黎但, 董金祥. 一种面向协作标签系统的图片检索聚类方法[J]. 中国图象图形学报, 2010, 15(11): 1635-1643

一种面向协作标签系统的图片检索聚类方法

李晓燕, 陈 刚, 寿黎但, 董金祥
(浙江大学计算机科学与技术学院, 杭州 310027)

摘 要: 为了更有效地进行图片检索, 提出了一种面向 Web2.0 协作标签系统的图片检索聚类方法。该算法首先针对标签空间由于标签表达多样性带来的不一致问题, 并通过挖掘标签间的词汇关系实现语义级查询扩展来得到语义可能相关的扩展图片结果集; 然后根据标签间的相关度度量选出图片结果集中与查询标签高相关的标签集, 接着采用一种自顶向下启发式的图划分算法来自动对次相关标签集进行分类。最后图片结果集即根据标签分类结果被聚类。为验证该方法的效果, 从标签图片共享网站 Flickr 上随机下载了大量真实图片集以及所含带的标签元数据, 在已实现的图片检索原型系统 PivotBrowser 上进行了大量实验, 结果证明, 该聚类算法能有效解决标签空间存在的标签表达不一致问题和标签查询歧义性问题, 能提供更满意的用户检索。

关键词: 标签; 不一致性; 歧义性; 关联度

An image clustering algorithm in collaborative tagging system

LI Xiaoyan, CHEN Gang, SHOU Lidan, DONG Jinxiang
(*Depertment of Computer Science and Technology, Zhejiang University, Hangzhou* 310027)

Abstract: In this paper, we propose a novel image clustering algorithm for effective image retrieval in Web2.0 tag-space. Different users may use different tags to describe the same object, causing inconsistency in tagging. Our algorithm capture the semantically similar tags to perform query expansion, and retrieve the candidate images which are possibly relevant to the query. The candidate tags can be shortlisted according to their tag relevances to the query tags. The shortlisted tags are then clustered on-the-fly using a graph partitioning algorithm. The candidate images are clustered based on the tag cluster results. The proposed algorithm is implemented in a prototype system called PivotBrowser. Experiment results performed on a large scale images that random downloaded from Flickr reveal that our proposal effectively address the inconsistency and ambiguity problems in tag-space image retrieval, and provide improved user satisfactory.

Keywords: tag; inconsistency; ambiguity; relevancy

0 引 言

基于 Web2.0 的社会化标签功能使得信息的分享和交流更为简单和方便。近年来, 互联网上出现了越来越多的针对文本和多媒体内容的标签应用系统。比如, Del.icio.us 提供的网页书签的联合标签, 同样的, 有针对学术论文发表领域的 CiteUlike, 针对图片标签共享的 Flickr, 以及针对视频标签共享的 YouTube 等。这些商业应用的成功也印证了用户使用自定义标签进行 Web 资源标示、共享查询的方式已经非常流行, 有很好的应用发展前景。社会化标签体现了一种很好的联合共享方式, 基于标签的检索已成为信息检索领域一种普遍而受欢迎的方法,

基金项目: 国家自然科学基金项目(60603044); 长江学者和创新团队发展计划资助项目(IRT0652); 国家高技术研究发展计划(863)项目(863-317-01-04-99)。
收稿日期: 2009-04-27; **改回日期**: 2009-08-27
第一作者简介: 李晓燕(1981—), 女, 计算机应用专业, 博士研究生。主要研究领域为图像数据库、图像语义检索技术、高维数据索引技术、文本语义挖掘。E-mail: kricel_lee@yahoo.com.cn。

但由于用户打标签具有明显的主观性,定义标签也相对随意,从而给标签共享系统的信息检索造成了如下困难:1)不同用户使用不同标签(比如同义词、词形变换等)来描述同一个事物,不同标签可能表示相同的含义,同义标签又导致标签系统的不一致性;2)由于缺乏统一的标准,致使同样的标签可能存在多种含义,而多义标签又导致查询语义的模糊性;3)用户定义的标签一旦离开上下文环境往往难以准确理解。相对于其他媒体资源,图片标签共享系统的信息表达和组织方式更加自由,图片语义内容更加丰富,理解主观性大,而且图片资源数据量通常非常庞大,这些都使得整个图片共享标签系统呈现高动态性,因而标签不一致性和查询模糊性等问题也更加严重。

在图片协作标签系统的检索中,最困扰用户检索的问题是同义标签和多义标签。这一方面原因是,比如 kid、kids、baby、babe 等表达同一意思但描述不同的标签,对于其中任意一个标签的查询,包含其他同义标签的图片可能都是用户想要的结果;另一方面,输入 tiger 进行检索时发现,对 tiger 的理解有多种,tiger 可以指老虎、也可以是指代猫科的动物或一种花 tiger lily,或者是著名的高尔夫球手泰格·伍兹,再或是苹果公司的一种产品标识。因此对同一标签的不同对象进行描述,要结合上下文,即要结合其他相关标签的描述内容。由于传统的基于关键词的查询绝对匹配的检索方法不能解决以上问题,从而导致目前在标签空间的检索效率低,用户查询满意度不高。如何快速、全面、准确地获得用户需要的信息,已成为学术界和产业界越来越关注的问题。

针对上述标签空间的图片检索存在和关注的主要问题,本文对图片协作标签系统的检索方法进行了研究和探索,提出了一种高效的图片标签检索聚类算法,旨在捕获标签描述的不一致性,快速自动地对检索结果进行聚类,有效区分标签的多义性,以解决查询模糊性问题,具体工作包括:1)挖掘标签语义联系,实现语义级查询扩展,从而在一定程度上解决了标签表达不一致的问题;2)挖掘标签间的关联关系,通过快速有效的聚类方法,一方面在一定程度上消除了查询标签的歧义性,另一方面以自动聚类结果的呈现方式来提高用户检索的满意度。

近年一些研究指出,尽管在标签空间缺乏规范的本体和语义结构,但仍可以通过标签数据中的关联信息的挖掘来发现一些模式和模型[1-2]。Schmitz 等人讨论了如何在社会化标签空间中通过关联分析来挖掘标签集的结构[3]。也有一些研究试图从标签集中提取出本体论结构[2,4];然而在高动态、大规模的标签空间中,创建、维护本体论结构不但代价很高,而且可扩展性差。由于标签代表的语义往往是模糊的和不全面的,因此对协作标签系统的信息检索离不开上下文环境。鉴于协作标签系统本身的结构化程度较低,为此,本文将通过标签系统本身的数据挖掘和语义关系来形成非结构化和半结构化知识库。

目前面向协作标签系统的信息浏览,通常要挖掘标签间的关联关系,标签间的关联关系的计算是发现标签结构化,进而对信息进行归类或提供标签推荐的基础。Leydesdorff 等人对关键词(标签)关联度分析在网络应用环境下的多种可能应用展开了讨论[5]。关键词关联度的计算则通常考虑关键词间共同出现的频率,以及关键词本身的使用频率,比如 Dice 距离、Jaccard 距离、余弦距离等。为此,本文将采用 Jaccard 距离来计算标签间的关联度。

依据图片的标签元数据即可以对图片进行聚类分析。目前已有许多图片检索系统研究都采用图片聚类的方法来改进网络图片检索,这些聚类算法基本上是基于设定的距离度量[6-8]。距离度量方法的选择决定了什么样的图片被聚集在一起,比如常见的欧拉距离,曼哈顿距离等。但是,这些方法都不能很好地适应于高动态性、海量的 Web2.0 标签系统。Flickr 通过挖掘标签信息来对图片检索结果进行聚类,然而,Flickr 的这种聚类应用,不但没有考虑所打标签本身的不一致性问题,而且该聚类应用仅适于对单个标签检索结果的聚类。标签间的关联度计算可以形成标签的关联度矩阵,Pothen 等人提出了一种图划分的算法[9]虽能有效产生最优的划分结果,但是由于该算法的计算复杂度很高,不适用网络检索时在线对检索结果进行聚类的实时要求。White 等人提出了两种光谱聚类算法[10],能有效找到最优的划分和划分数。鉴于图片检索结果的聚类应用,本文采用自顶向下的启发式图划分算法,能快速有效地对在线图片检索结果进行聚类。

1 预处理:标签数据结构

不失一般性,系统首先为整个图片数据库建立倒排索引,即每项均是一个与该项标签对应的所有图片 ID 列表。本文提出的检索聚类算法利用了标

签空间的以下两类结构信息:一是依据词汇知识得到的标签词汇关系结构(如同义词、各类拼写变换等同级关系联系);二是依据统计及关联信息分析得到的标签间的关联关系结构。在介绍图片检索聚类算法前,下面先给出算法依据的主要数据结构的分析和产生过程。

1.1　标签间词汇关系结构

标签空间的图片检索,用户需要得到的是与查询意思相同的查询结果。比如用户查询"movie",而那些被标注为"film"的图片也是用户感兴趣的。由于图片标签的空间标签描述,因存在随意性、主观性而呈现多元化,且普遍存在诸如同义词(flower、bloom、blossom)、各种词性变换(单复数形式、动名词形式、不同缩略写等)、语义高相关的词(food 和 cuisine)等词汇级关联,因此相对传统的关键词绝对匹配的查询,语义级相似的查询匹配更能为用户提供完整的查询结果。

本文通过标签之间的同级关系来帮助用户查找相关信息,即通过构建标签的词汇关系结构来完成语义级查询扩展。首先借助已有词汇的关联知识、词形变化知识等来构建包含同义词、词形变化、语义相近等词汇关系的标签词典。以下首先给出标签原子的概念,其定义如下:

定义 1 标签原子　根据标签词典获得的所有词汇关系的最小结构,称为标签原子 $\boldsymbol{A}$,它是一个标签的集合,并满足下列条件:

1) 如果一个标签原子 $\boldsymbol{A}$ 包含一个标签 T,则它必须也包含标签辞典中其他与标签 T 词汇相关的标签;

2) 对标签原子 $\boldsymbol{A}$ 中任意两个标签 T_1 和 T_2,它们必须是词汇相关的。

如果一个标签在标签辞典中具有多种词义,那么它将出现在多个标签原子中。对所有标签原子需要构建标签与标签原子之间的倒排索引表,其具体定义如下:

定义 2 标签原子倒排表　标签原子倒排索引表中每一项的形式如下:

$$\langle T_i, \boldsymbol{A}_{i,1} \text{ 的 id}, \boldsymbol{A}_{i,2} \text{ 的 id}, \cdots \rangle$$

其中,$\boldsymbol{A}_{i,j}$ 为包含标签 T_i 的标签原子,id 代表唯一标识符,此倒排表可简称为 TAIL(tag atom inverted list)。

对于任意标签 T_i,可以通过查询标签原子倒排表来得到该标签的同级(语义级)扩展标签集合 $\boldsymbol{E}(T_i) = \bigcup_j \boldsymbol{A}_{i,j}$。在此基础上,再给出以下查询支持的概念:

定义 3 查询支持　给定查询 $\boldsymbol{q}$(含查询标签集 $\{T_1, T_2, \cdots, T_n\}$),任意含有标签集 $\{\tilde{T}_1, \tilde{T}_2, \cdots, \tilde{T}_n\}$ 的图片都被认为是与此查询标签集 $\{T_1, T_2, \cdots, T_n\}$ 相关的图片,其中 $\tilde{T}_i \in E(T_i)$。标签集合 $\{\tilde{T}_1, \tilde{T}_2, \cdots, \tilde{T}_n\}$ 也称为被查询 $\boldsymbol{q}(\{T_1, T_2, \cdots, T_n\})$ 所支持的查询标签集合。

1.2　标签间关联度矩阵

对于与同一标签描述对应的不同实体可以依据上下文(即关联出现的其他标签描述)来明确其标签含义,特别是对于由消除多义标签引起的查询模糊性。本文通过对整个标签图片空间中标签的使用统计信息和关联信息进行分析来得到标签间的关联矩阵。

定义 4 标签关联度矩阵　对于一个标签集合 T,可采用 Jaccard 系数通过计算其中任意两标签间的关联度值来得到此标签集合的关联度矩阵,比如第 i 个标签 T_i 和第 j 个标签 T_j,$\boldsymbol{I}(T_i)$ 表示含有第 i 个标签 T_i 的图片集,$\boldsymbol{I}(T_j)$ 是含有第 j 个标签 T_j 的图片集,标签 T_i 与 T_j 间的关联度值为

$$R(T_i, T_j) = \frac{\left|\boldsymbol{I}(T_i) \cap \boldsymbol{I}(T_j)\right|}{\left|\boldsymbol{I}(T_i) \cup \boldsymbol{I}(T_j)\right|}。$$

2　聚类算法

图 1 给出了图片检索聚类的流程图。该流程用到的主要数据结构有(在图中用粗线标出):图片数据库(包含被检索的所有图片)、图片标签集(图片数据库中包含的所有标签)、标签原子集(标签集中提取的所有标签原子)以及索引结构(图片倒排索

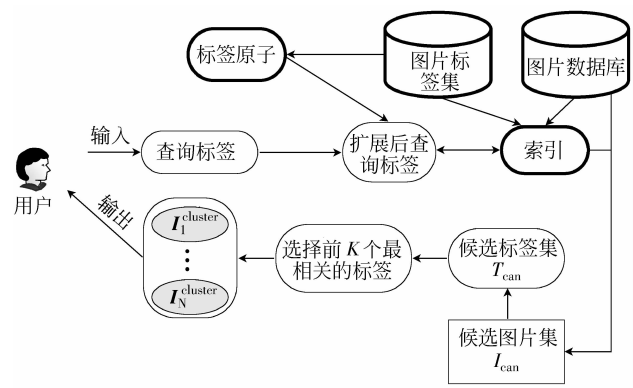

图 1　图片检索聚类的流程图

Fig. 1　The dataflow chart of image searching and clustering

引和 TAIL)。对待搜索图片数据库,系统是通过预先处理来得到运行时需要的主要数据结构,如标签原子及索引等。当有新的图片和标签加入,这些结构能很方便地自动更新。

在线查询和聚类过程分为以下主要步骤(图 1):

1) 用户发出一条查询信息(包含一个或多个查询标签),系统首先通过查询标签原子倒排表来得到各个标签的语义级扩展标签集。然后将从图片数据库得到的所有包含被查询标签集所支持的标签集的图片作为候选图片,标记为 $\boldsymbol{I}_{\text{can}}$,而与候选图片相关的所有标签(除了查询标签)称为候选标签集,记作 $\boldsymbol{T}_{\text{can}}$;

2) 根据标签与查询之间的相关度,系统从候选标签集 $\boldsymbol{T}_{\text{can}}$ 中选取前 K 个最相关的标签,用于对候选图片集进行语义关联分析;

3) 将选出的前 K 个最相关的标签,根据标签间的关联度值划分成最优的若干个聚类;

4) 而候选图片集则根据每幅图片与标签聚类间的关系被相应聚类。

2.1 查询扩展

这一步是在查询初始,先通过构建的标签间的词汇关系结构对查询标签进行查询扩展,然后选用扩展后的标签来查询获得跟查询可能相关的所有候选图片集 $\boldsymbol{I}_{\text{can}}$,步骤如下:

1) 对于含有 n 个查询标签的查询 $\boldsymbol{q}(\{T_1,T_2,\cdots,T_n\})$,通过标签原子倒排索引表 TAIL 得到所有被查询 $\boldsymbol{q}$ 支持的查询 $\tilde{\boldsymbol{q}}(\{\tilde{T}_1,\tilde{T}_2,\cdots,\tilde{T}_n\})$。

2) 对于查询 $\boldsymbol{q}$ 或每个被查询 $\boldsymbol{q}$ 支持的查询 $\tilde{\boldsymbol{q}}$,先通过图片倒排索引来获得包含某个查询中所有标签词的图片。然后将查询 $\boldsymbol{q}$ 与由其支持的所有查询 $\tilde{\boldsymbol{q}}$ 获得的结果图片集合并,作为候选图片集 $\boldsymbol{I}_{\text{can}}$,相应地得到候选图片结果集 $\boldsymbol{I}_{\text{can}}$ 所包含的候选标签集 $\boldsymbol{T}_{\text{can}}$。

由于标签原子倒排索引结构所占的空间有限,且查询扩展在内存操作,其 CPU 计算时间可以忽略,另外,由于图片查询采用倒排索引设计与当前实用系统相当。因此,这步额外的时间开销可以忽略不计。

2.2 选择前 K 个与查询最相关的标签集

候选标签集 $\boldsymbol{T}_{\text{can}}$ 通常很大,而且可能包含大量与查询无关的标签,这一步的目的是从 $\boldsymbol{T}_{\text{can}}$ 中选出若干与查询高相关的标签。本文通过定义一种标签与查询之间相关度的计算度量,从候选标签集 $\boldsymbol{T}_{\text{can}}$ 中选出前 K 个与查询最相关的标签,具体定义为

定义 5 标签与查询的相关度 $r(T,\boldsymbol{q})$ 表示标签 T 与查询 $\boldsymbol{q}$ 之间的相关度计算度量,该相关度的计算要结合文档率和倒文档率进行,具体操作如下:

1) 计算标签 T 和扩展后查询标签集之间共同出现的频率,等同于计算该标签在候选图片集内的使用频率 $f(T)$,即候选图片集 $\boldsymbol{I}_{\text{can}}$ 中包含标签 T 的图片数;

2) 计算标签 T 在整个图片数据库被使用的倒文档频率 $f_{\text{ID}}(T)$:$f_{\text{ID}}(T)=\log\dfrac{|\boldsymbol{I}|}{|\boldsymbol{I}(T)|}$,其中 $\boldsymbol{I}$ 表示整个图片数据集,$\boldsymbol{I}(T)$ 表示包含标签 T 的图片集;

3) 标签与查询间的相关度值($r(T,\boldsymbol{q})=f_{\text{D}}(T)\times f_{\text{ID}}(T)$)。

标签的文档率 f_{D}(D 代表 document)和倒文档率 f_{ID}(下角 ID 代表 inverted document)都可以根据索引结构直接获得,本文已用实验证明,这部处理的运算开销很有限。

2.3 标签聚类

对查询相关标签进行聚类的目的来自于如下的观察和假设。同一个标签描述可能表达多种含义,比如“apple”可能指一种水果,也可以指代苹果公司相关的产品,或者是一种食品“苹果派”。通过对这些高相关标签集之间的关联进行聚集分析和提取,并进行查询结果的语义划分,可一定程度地区分查询语义的模糊性。而且事件和主题相关的图片的标签也呈现一定的聚合性,基于关联度的聚合分析也能将不同事件或主题的图片结果区分开。本文采用一种自顶向下启发式的图划分算法,自动快速地将 K 个标签划分成不超过设定划分上限的最优聚类结果。具体步骤如下:

取出前 K 个最相关的标签的关联子矩阵,如果将 K 个标签看做 K 个顶点,将两标签 T_i 与 T_j 间的关联度值看做两标签相连边的权重 $w(i,j)$,那么对 K 个标签的聚类问题就可以看做是对含 K 个顶点的带权重无向图的划分问题,基于文献[10]的工作,本文采用的图划分算法是基于一个关于带权重无向连接图在一个划分下的商值度量法。

定义 6 划分模块性函数 假设图 $\boldsymbol{G}$ 被划分为 l 个顶点集合,可通过定义划分模块性函数 $Q(\boldsymbol{P}_l)$ 来度量该划分 $\boldsymbol{P}$,即 $Q(\boldsymbol{P}_l)=\sum_{c=1}^{l}\left[\dfrac{C(\boldsymbol{V}_c,\boldsymbol{V}_c)}{C(\boldsymbol{V},\boldsymbol{V})}-\left(\dfrac{C(\boldsymbol{V}_c,\boldsymbol{V})}{C(\boldsymbol{V},\boldsymbol{V})}\right)^2\right]$,其中 $\boldsymbol{V}_c$ 代表划分后第 c 个顶点集合($c=1,2,\cdots,$

l),V 代表图 $\boldsymbol{G}$ 中的整个顶点集合,$C(\hat{\boldsymbol{V}},\tilde{\boldsymbol{V}})=\sum_{i\in\hat{V},j\in\tilde{V}}w(i,j)$是两个顶点集合 $\hat{\boldsymbol{V}}$ 和 $\tilde{\boldsymbol{V}}$ 之间所有边的权重之和。

一个好的划分应该使划分块内部各点之间的关联度越大越好,而划分块之间各点的相互关联度越小越好。根据划分模块性函数的定义,Q 值越大表示图划分的结果越好,所以可以基于此划分度量函数 Q 来实现一种自顶向下启发式的图划分算法,并以逐步最大化图划分的 Q 值的方式,快速地找到划分数不超过 θ 的最优的 l 个划分结果。

在介绍算法之前,先给出算法运算过程中用到的第 l 个特征向量矩阵 $\boldsymbol{U}_l$ 的计算,即

首先给定一个无向连接图 $\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E})$,其中 $\boldsymbol{V}$ 是节点集、$\boldsymbol{E}$ 是边的集合,构造图 $\boldsymbol{G}$ 的权重矩阵 $\boldsymbol{W}(\boldsymbol{W}\in\mathbf{R}^{K\times K})$,同时构造对角矩阵 $\boldsymbol{D}(\boldsymbol{D}\in\mathbf{R}^{K\times K})$,除了对角元素,其余的值为 0,其中第 i 个对角元素 d_i 的值为$\sum_{j=1}^{K}W_{i,j}$。

然后计算迁移矩阵 $\boldsymbol{M}(\boldsymbol{M}=\boldsymbol{D}^{-1}\boldsymbol{W})$,并用迁移矩阵 $\boldsymbol{M}$ 的前 $l-1$ 个特征向量构成特征向量矩阵 $\boldsymbol{U}_l$($\boldsymbol{U}_l=[u_1u_2\cdots u_{l-1}]$)。

算法伪代码的大致流程如下:

GRAPH_P 算法:一种启发式的图划分算法

初始化 $l=2$
初始化划分 $\boldsymbol{P}$ 为一个聚类,包含整个图 $\boldsymbol{G}$
重复
　设置 $\boldsymbol{P}_l=\boldsymbol{P}$
　对于 $\forall\boldsymbol{V}_c\in\boldsymbol{P}$
　　做
　　通过保留矩阵 $\boldsymbol{U}_l$ 中与 $\boldsymbol{V}_c$ 相关的行来得到矩阵 $\boldsymbol{U}_{l,c}$;
　　运用 k-平均聚类算法将 $\boldsymbol{U}_{l,c}$ 分裂成两个新的子类 $\boldsymbol{V}_{c,1}$ 和 $\boldsymbol{V}_{c,2}$;
　　用 $\boldsymbol{V}_{c,1}$ 和 $\boldsymbol{V}_{c,2}$ 替换划分 $\boldsymbol{P}$ 中的 $\boldsymbol{V}_c$ 来获得新的分割 $\hat{\boldsymbol{P}}$;
　　如果 $Q(\hat{\boldsymbol{P}})>Q(\boldsymbol{P})$
　　当　接受新的划分 $\hat{\boldsymbol{P}}$, $\boldsymbol{P}_l\leftarrow\hat{\boldsymbol{P}}$
　　则　不改变 $\boldsymbol{P}_l$
　　$l\leftarrow\boldsymbol{P}_l$ 中的聚类数
　设置 $\boldsymbol{P}=\boldsymbol{P}_l$
直到 $l>\theta$ 或者不能继续分割
根据聚类的内聚度度量值给 $\boldsymbol{P}$ 中的划分聚类排序

该算法采用依次二分划分的启发式图划分方式,最初划分值 l 为 2,初始划分 $\boldsymbol{P}$ 就是先将整个图 $\boldsymbol{G}$ 作为一个聚类,然后重复以下过程:

1)对于任意一个属于划分 $\boldsymbol{P}$ 的集合 $\boldsymbol{V}_c$,利用快速聚类法(k-mean)将由集合 $\boldsymbol{V}_c$ 形成的子图二分,分裂得到两个更小的集合 $\boldsymbol{V}_{c,1}$ 和 $\boldsymbol{V}_{c,2}$;

2)通过将集合 $\boldsymbol{V}_{c,1}$ 和 $\boldsymbol{V}_{c,2}$ 取代划分 $\boldsymbol{P}$ 中的集合 $\boldsymbol{V}_c$ 来得到新的划分 $\hat{\boldsymbol{P}}$;

3)如果 $Q(\hat{\boldsymbol{P}})>Q(\boldsymbol{P})$,则接受此次划分,更新划分 $\boldsymbol{P}$,否则保持划分 $\boldsymbol{P}$ 不变;

(1)如果 $l>\theta$,或者划分 $\boldsymbol{P}$ 不能再继续被划分,则算法停止;

(2)将划分 $\boldsymbol{P}$ 内的集合根据集合的聚合度进行排序,聚合度按照以下公式计算:

$$Coh(\boldsymbol{V}_c)=\left(\frac{C(\boldsymbol{V}_c,\boldsymbol{V}_c)}{C(\boldsymbol{V},\boldsymbol{V})}-\left(\frac{C(\boldsymbol{V}_c,\boldsymbol{V})}{C(\boldsymbol{V},\boldsymbol{V})}\right)^2\right)\times\log\frac{|\boldsymbol{V}|}{|\boldsymbol{V}_c|}$$

2.4　图片结果聚类

得到以上相关标签集的聚类结果后,再将候选图片集 $\boldsymbol{I}_{\text{can}}$ 根据图片与每个标签聚类的关联度相应地被划分为不同聚类,具体过程可以描述如下:

1)根据以上标签的聚类结果,对于候选图片集 $\boldsymbol{I}_{\text{can}}$ 中任意一幅图片,如果该图片含有 m 个或者 m 个以上第 i 个标签集 $\boldsymbol{V}_i$ 中的标签,则该图片被归为第 i 个图片聚类 $\boldsymbol{I}_i^{\text{cluster}}$;

2)如果一幅图片不属于第 1 个图片聚类 $\boldsymbol{I}_1^{\text{cluster}}$ 到第 l 个图片聚类 $\boldsymbol{I}_l^{\text{cluster}}$ 的任何聚类,则该图片被归为其他聚类 $I_{\text{other}}^{\text{cluster}}$;

3)最终的候选图片结果集 $\boldsymbol{I}_{\text{can}}$ 被划分为 $l+1$ 个聚类,值得注意的是,一幅图片可能属于第 1 个聚类 $\boldsymbol{I}_1^{\text{cluster}}$ 到第 l 个聚类 $\boldsymbol{I}_l^{\text{cluster}}$ 中的多个聚类。

3　实验分析

为了评价和分析本文提出的图片检索聚类算法的效率和实际应用价值,通过 Flickr 的 API,从目前最流行的图片标签共享网站 Flickr 上随机下载了 60 多万幅图片及与图片相关的标签元数据,经过处理去除掉无法显示的图片,最终得到包含 523 746 幅图片的图片集,并得到了相关联的 427 482 个不同的标签。通过检索前的预处理工作,完成了标签与图片的倒排索引、标签原子倒排索引及标签关联矩阵的构建。其中为获得标签集的标签原子,借助于词汇数据库 WordNet2.1[11] 的词汇知识来获得具有同义词关系的标签和采用文献[12]提出的算法来查找有后缀变化的标签等途径。

以下所有的实验都是在已经实现的图片检索原型系统 PivotBrowser[13] 上执行的。PivotBrowser 运行在 P2,2 G RAM,160 G 硬盘的 PC 上,底层的操作系统是 Windows XP。实验分为定性的实例分析和定量的性能和效率评价两部分。其中实例分析部分,首先从一些实例给出检索聚类算法各个步骤的聚类结果,然后给出具体的图片聚类结果实例,用来说明本文的聚类算法如何解决标签空间查询的模糊性等问题;定量评价部分,首先通过查询扩展前后返回图片数目的比较,用于说明通过扩展在捕获了标签空间的不一致性的同时,提高了系统的查全率;然后通过用户评级比较和聚类算法的性能测试,分别给出了检索聚类算法在聚类效果和性能上的整体评价。

检索聚类算法中涉及的主要参数如下:

1)高相关标签选取时,选取的标签个数 K;

2)相关标签聚类时,设置的最大聚类数 θ;

3)图片聚类时,图片与标签聚类关联的标签个数下限 m;

根据实验图片集合标签集的规模,实验中采用的参数默认值分别为 $K=100$、$\theta=10$、$m=2$。

3.1 实例分析

首先通过实例分析呈现查询例子在每一步聚类操作后的聚类结果,然后通过给出最终聚类结果的实例来说明本文提出的查询聚类算法的效果,在一定程度上解决了查询模糊性等问题。

3.1.1 标签聚类过程结果

表 1 给出了一些查询和根据查询获得的若干最相关的标签。从表 1 部分选择结果可以看出,相关的标签确实反映了查询标签的不同语义。比如对于查询 apple mac,选出的标签包含了苹果公司的不同产品 macintosh、imac、speaker、iphone、ipod 等;对于查询 dog,选出的标签包含了不同狗的品种 corgi、basenji、dalmatian、welshcoigi、poodle 等,以及对狗的其他称谓 puppy、pet 等;而附加了查询标签 dalmatian 的查询 dog dalmatian,选出的标签就含有对 dalmatian 这个品种狗的特征描述 spots 等、常取的昵称 romeo 等。不难发现,这些与查询相关的标签能反映查询的不同语义,比如种类、属性、事件、地点等。

表 1 高相关标签选择结果

Tab.1 Result of selecting highly relevant tags

查询标签	相关标签列表
apple	mac、iphone、ipod、macbook、york、imac、fruit…
apple mac	macintosh、imac、speakers、iphone、ipod…
movie	motion、picture、magazine、film、lifemagazine…
baby	family、jack、christmas、cute、boy、bewborn、portrait、child、girl、kids…
baby cat	chicago、birthday、europe、england、kitten…
window	store、fashion、light、old、display、glass、windows、shopping、mannequin…
dog	corgi、puppy、basenji、pet、dogs、dalmatian、animal、welshcoigi、cute、poodle…
dog dalmatian	spots、romeo、puppy、dogs、Dalmatians、dalmata…
dog poodle	pet、animal、toypoodle、standardpoodle、poedel、pudel、black、tommy…

表 2 给出了标签聚类的结果,鉴于论文篇幅,下面仅选出 3 个查询的聚类结果。每个聚类内的标签按照标签出现的频率排序。从聚类结果可以看出,选出的相关标签有效地被划分落入了若干不同语义类。这些标签聚类确实反映了一些共性,比如对于查询 apple 的聚类,通过标签聚类这步可反映出 apple 的不同含义,{mac iphone ipod macbook…}代表了苹果公司相关的概念,{pie food applepie baking…}指美国人最流行的食物——苹果派,{fruit apples…}指苹果这种水果,而{picking…}则表示摘苹果这个事件等。从聚类结果看出,候选标签的模糊语义被自动区分了。

表 2 标签聚类的结果

Tab.2 Result of clustering the candidate tags

查询标签	聚类结果
apple	{mac iphone ipod macbook…}
	{pie food applepie baking…}
	{fruit apples…}
	{picking…}
	……
apple mac	{macintosh ibook Microsoft applemacintosh…}
	{imac iphone ipod…}
	{speakers dj djsando…}
	{room wii xbox dvds…}
	……
window	{store fashion display windows shopping…}
	{view airplane condo…}
	{white green red nikon canon blue…}
	{poland stainedglass krakow}
	……

3.1.2 图片分类结果

由以上标签的聚类效果可以看出,经过聚类的标签,确实反映了不同语义方面的聚合度,比如不同的语义内涵、不同的地点、不同的事件、不同的场景等等。下面再通过具体的图片聚类实例来展示图片聚类后的效果。PivotBrowser 系统在导航栏里显示

每次聚类的结果,每个聚类由属于该聚类的 3 幅图片缩图和相关的若干标签表示。通过代表缩图和相关的标签,检索者可以大致知道该聚类的内容。以下就具体查询得到的导航栏的内容截图来阐释聚类效果。

由图 2 给出的查询 nature 的图片聚类结果可见,聚类结果集被分为自然界的花{flower flowers macro …},海天相关的自然风光{blue red canon water nikon white …},自然界的花草{blossom blumen flores nature flori …},自然界的鸟{animal animals bird birds wildlife …},自然界的昆虫{butterfly insect butterflies insects …},野生公园的风景{park wild outdoors natural …}等。

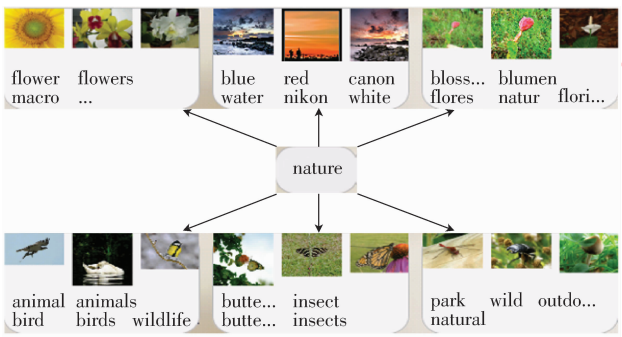

图 2　PivotBorwser:查询“nature”图片的聚类结果

Fig. 2　PivotBorwser: Clustered results of “nature”

图 3 给出了查询 window 图片的聚类结果,查询结果被分为商品展示窗{fashion windows shopping …}、窗外的视野{blue red canon green nikon white …}、冬天结冰的窗户{winter snow ice frost …}、室内窗帘搭配的窗户{chair curtain}等等。

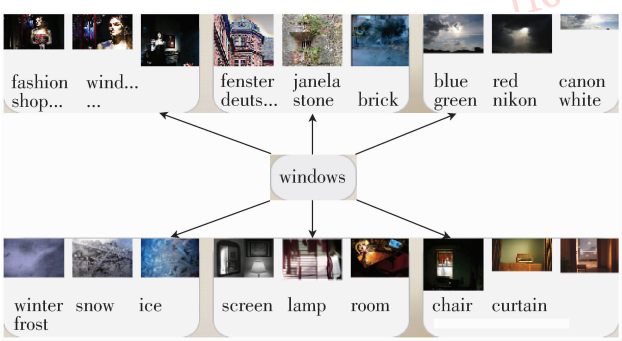

图 3　PivotBorwser:查询“window”图片的聚类结果

Fig. 3　PivotBorwser: Clustered results of “window”

更进一步,图 4 给出了结合查询词 window 和 nature 图片的聚类结果,由图 4 可以发现,不同地点被自动聚类{brasil brazil Portugal azul verde …},{Switzerland schweiz suisse},{England geotachnic london geo …},{italy italia},{nyc newyork …},不同场景自动聚类为{winter snow morning cold ice …}、{rain raindrops}等。

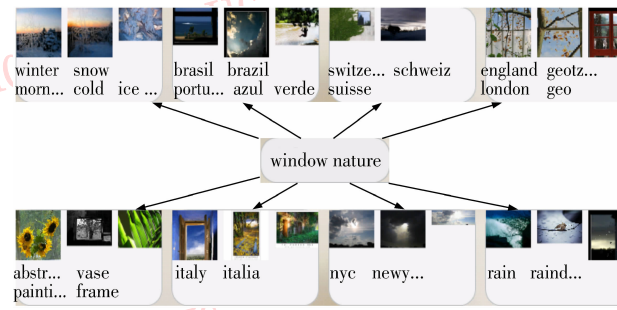

图 4　PivotBorwser:查询“window nature”图片的聚类结果

Fig. 4　PivotBorwser: Clustered results of “window nature”

3.2　整体评价

由于对图片标签共享空间进行查询很难获得对应查询的标准结果集,因此不适合采用传统的查全率和查准率来度量。本文首先从返回结果数上评估查询扩展的作用。然后采用二元的用户评级方法来度量查询结果的相关度,即对于返回的图片结果,用户可以标记为相关和不相关,并和 Flickr 提供的标签聚类结果进行比较,以评价算法的查询准确度。聚类过程考虑最相关元素的策略和启发式的算法,不但可大幅度提高计算速度,也能有效去掉非相关元素产生的噪音影响,从而取得了更加满意的图片聚类结果。实验还给出了聚类各个步骤的平均执行时间。

3.2.1　查询扩展效果

本文提出的检索聚类算法是通过查询扩展来返回所有查询支持的图片,所有被标签词典包含的与查询表达不一致,但意义相同的标签都能被捕获,且返回的图片数比绝对匹配返回的图片数增加。实验用 27 811 个标签依次单个做查询,分别统计了其在绝对匹配和查询扩展下返回的图片数。图 5 给出了这两类的查询结果,即经过查询扩展(PivotBrowser)和绝对匹配的(exact match)返回的图片数和对应标签分布的曲线。横坐标表示返回的图片数不少于对应纵坐标值的标签比例。由图 5 曲线显示结果可以看出,查询扩展能返回更多的图片,尤其对于中低频的查询标签,查询结果可被扩大到更大的范围。

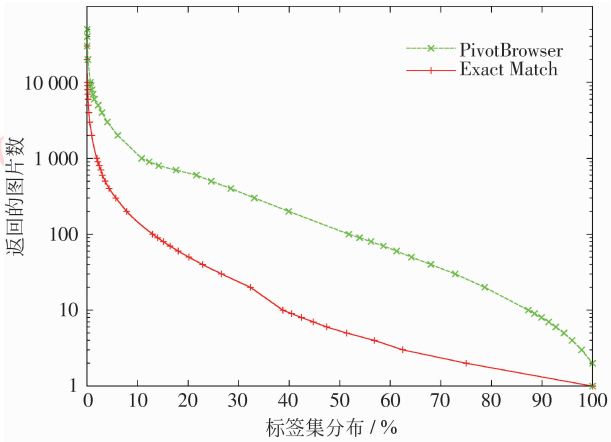

图 5 返回图片数与对应标签分布图

Fig. 5 The number of retrieved images versus the distribution of tags

3.2.2 聚类结果的用户满意度评价分析

目前还没有文献报道关于标签图片的类似聚类算法研究,但在产业界,Flickr 已推出了聚类的应用。图 6 是 Flickr 聚类应用的界面截图,其给出了关于查询 window 的 4 个聚类,用户点击“See more in this cluster…”便可以浏览该聚类图的结果集。虽然 Flickr 目前只能提供对单个标签的查询聚类,但由于本文实验的数据集也是从 Flickr 网站随机下载的,所以二者之间的查询满意度评级具有一定可比性。比较方式如下:

实验样本是从标签集里选取的 30 个可能产生歧义或拼写不一致的标签,包括出现频率很高的词(如 baby,window,apple,tv 等),出现频率适中的词(如 ball,tiger,beetle 等),还包括一些出现次数很少但是较有意义的词(如,brand,cuisine,pda 等)。检索实验时,招募 20 位志愿者(平时熟悉网络检索,常使用图片检索的在校学生),每个人从以上准备的 30 个标签中随意选择 20 个标签用来执行 20 次查询,对于每个查询,Flickr 和本文聚类算法(PivotBrowser 系统)分别返回聚类结果。参加实验者先选定一个与查询最相关的聚类,然后对该聚类的前 100 个聚类结果依次评定相关或不相关。表 3 给出了两种聚类结果的在前 n 个(n=20、40、60、80、100)返回结果中的平均查准率。结果表明,本文的聚类算法优于 Flickr 的聚类效果。

表 3 平均查准率比较

Tab. 3 Comparison of query accuracy

算法	平均查准率/%				
	n=20	n=40	n=60	n=80	n=100
本文聚类算法(PivotBrowser)	92.5	92.5	91.2	90.6	89.8
Flickr	83.2	82.5	82	81.4	80.8

3.2.3 聚类算法的性能分析

实验进行了 200 个不同的查询,每个查询重复执行 100 次,同时计算聚类算法每一步的运算时间和整个算法的执行时间(不包括检索实际图片的时间)。整个检索聚类算法的平均执行时间是 753.1 ms。

图 6 Flickr:查询“window”图片的聚类结果

Fig. 6 Flickr: Clustered results of “window”

表 4 给出了得到候选图片集合 $\boldsymbol{I}_{\text{can}}$ 的平均时间 $\bar{t}_I$ 是 535.3 ms,选择相关标签的平均时间 $\bar{t}_T$ 仅仅是 15.5 ms,标签聚类的平均时间 $\bar{t}_T^{\text{clus}}$ 是 97.5 ms,图片聚类的平均时间 $\bar{t}_I^{\text{clus}}$ 是 102.3 ms。结果表明,从倒排索引中得到候选图片是算法主要的开销,而这同样也是传统检索算法必须执行的,因此本文的算法仅带来很小的额外开销代价。

表 4　聚类每阶段的 CPU 平均时间开销 /ms

Tab.4　Run-time CPU cost in each phase

获取候选图片集合	选择相关标签	标签聚类	图片聚类
535.3	15.5	97.5	102.3

4　结　论

由于协作标签空间普遍存在同义标签和多义标签,从而使得基于标签的检索存在查询语义的模糊性以及检索结果由于标签的不一致性而导致的查询效率低和不完整的问题。本文提出的图片检索聚类算法可支持多个标签的查询,同时通过有效的查询扩充,一定程度上解决了 Web2.0 标签空间的表述不一致的问题,通过对查询相关联的标签进行聚类分析,实现了图片结果的有效聚类,一定程度上解决了查询语义模糊的问题。实验结果表明,本文提出的检索聚类方法,可以实现对高动态、海量标签图片进行快速、有效的检索,并可在一定程度上解决标签空间中标签本身语义的一致性问题和查询语义模糊的问题。

在进一步的工作中,将继续研究和挖掘标签空间中其他词汇知识和常识,以不断丰富标签词典。还将继续优化聚类算法,以及考虑引入视觉特征的聚类和排序。

参考文献(References)

[1] Golder S A, Huberman B A. Usage patterns of collaborative tagging systems[J]. Journal of Information Science, 2006, 32(2): 198-208.

[2] Schmitz P. Inducing ontology from flickr tags[C]//Proceedings of the Workshop on Collaborative Web Tagging at WWW. Edinburgh, UK: ACM Press, 2006.

[3] Schmitz C, Hotho A, Jaschke R, et al. Mining association rules in folksonomies[C]// Proceedings of the 10th IFCS Conference. Berlin, Heidelberg, German: Springer, 2006, 260-270.

[4] Mika P. Ontologies are us: A unified model of social networks and semantics[J]. Journal of Web Semantics, 2007, 5(1): 5-15.

[5] Leydesdorff L, Vaughan L. Co-occurrence matrices and their applications in information science: extending ACA to the web environment[J]. Journal of the American Society for Information Science and Technology, 2006, 57(12): 1616-1628.

[6] Jing F, Wang C, Yao Y, et al. Igroup: Web image search results clustering[C]//Proceedings of the 14th Annual ACM International Conference on Multimedia. New York, USA: ACM, press, 2006, 377-384.

[7] Gao Ying, Liu Dayou, Xu Yi. Framework of feature weighted clustering algorithm[J]. Computer Science, 2008, 135(110): 152-154.[高滢, 刘大有, 徐益. 一种特征加权的聚类算法框架[J]. 计算机科学, 2008, 135(110): 152-154.]

[8] Lei Xiaofeng, Xie Kunqing, Lin Fan, et al. An efficient clustering algorithm based on local optimality of K-means[J]. Journal of Software, 2008, 19(7): 1683-1692.[雷小锋, 谢昆青, 林帆, 等. 一种基于 K-means 局部最优性的高效聚类算法[J]. 软件学报, 2008, 19(7): 1683-1692.]

[9] Pothen A, Simon H D, Liou K P. Partitioning sparse matrices with eigenvectors of graphs[J]. SIAM Journal on Matrix Analysis and Applications, 1990, 11(3): 430-452.

[10] White S, Smyth P. A spectral clustering approach to finding communities in graphs[C] //Proceedings of the Fifth SIAM International Conference on Data Mining. Newport Beach, CA, USA: SIAM, 2005, 274-285.

[11] Miller G A, Beckwith R, Fellbaum C, et al. Introduction to wordnet: an on-line lexical database[J]. International Journal of Lexicography, 1990, 3(4):235-244.

[12] Newman M, Girvan M. Finding and evaluating community structure in networks[J]. Physical Review E, 2004, 69(2): 026113.

[13] Li Xiaoyan, Shou Lidan, Chen Gang, et al. PivotBrowser: A tag-space image searching prototype[C]//Proceeding of the 17th International Conference on World Wide Web. New York: ACM Press, 2008, 1111-1112.