软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(6):1486−1499 [doi: 10.3724/SP.J.1001.2012.04073]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种结合主动学习的半监督文档聚类算法*

赵卫中[1,3+], 马慧芳[2,3], 李志清[1], 史忠植[3]

[1](湘潭大学 信息工程学院,湖南 湘潭 411105)

[2](西北师范大学 数学与信息科学学院,甘肃 兰州 730070)

[3](中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100190)

Efficiently Active Learning for Semi-Supervised Document Clustering

ZHAO Wei-Zhong[1,3+], MA Hui-Fang[2,3], LI Zhi-Qing[1], SHI Zhong-Zhi[3]

[1](College of Information Engineering, Xiangtan University, Xiangtan 411105, China)

[2](College of Mathematics and Information, Northwest Normal University, Lanzhou 730070, China)

[3](Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100190, China)

+ Corresponding author: E-mail: zhaoweizhong@gmail.com

Zhao WZ, Ma HF, Li ZQ, Shi ZZ. Efficiently active learning for semi-supervised document clustering. *Journal of Software*, 2012,23(6):1486−1499. http://www.jos.org.cn/1000-9825/4073.htm

Abstract: Semi-Supervised document clustering and employing limited prior knowledge to aid in unsupervised clustering, have recently become a topic of significant interest to data mining and machine learning communities. Because receiving supervised data may be expensive, it is important to attain the most informative knowledge to improve the clustering performance. This paper presents a semi-supervised document clustering algorithm with active learning for pairwise constraints, aiming at getting improved clustering performance. The semi-supervised document clustering algorithm is a constrained DBSCAN (cons-DBSCAN) algorithm, which incorporates pairwise constraints to guide the clustering process in DBSCAN. Basing on measure of constraint set utility and analysis of DBSCAN algorithm, an active learning approach is proposed to select informative document pairs for obtaining user feedbacks. Experimental results show that this proposed approach is effective in document clustering. The clustering performance of active Cons-DBSCAN has dramatically improved with selected pairwise constraints. Moreover, the proposed approach performs better than the two representative methods.

Key words: semi-supervised clustering; document clustering; active learning; pairwise constraint

摘 要: 半监督文档聚类,即利用少量具有监督信息的数据来辅助无监督文档聚类,近几年来逐渐成为机器学习和数据挖掘领域研究的热点问题.由于获取大量监督信息费时费力,因此,国内外学者考虑如何获得少量但对聚类性能提高显著的监督信息.提出一种结合主动学习的半监督文档聚类算法,通过引入成对约束信息指导DBSCAN的聚

* 基金项目: 国家自然科学基金(61105052, 61070232); 湖南省自然科学基金(11JJ4051); 湖南省教育厅一般项目(10C1262); 湘潭大学博士启动基金(10QDZ42); 中国科学院计算技术研究所智能信息处理重点实验室开放基金(IIP2010-6); 西北师范大学青年教师科研能力提升计划骨干项目(NWNU- LKQN-10-1)

收稿时间: 2010-11-03; 修改时间: 2011-03-31; 定稿时间: 2011-06-24

类过程来提高聚类性能,得到一种半监督文档聚类算法 Cons-DBSCAN.通过对约束集中所含信息量的衡量和对DBSCAN 算法本身的分析,提出了一种启发式的主动学习算法,能够选取含信息量大的成对约束集,从而能够更高效地辅助半监督文档聚类.实验结果表明,所提出的算法能够高效地进行文档聚类.通过主动学习算法获得的成对约束集,能够显著地提高聚类性能.并且,算法的性能优于两个代表性的结合主动学习的半监督聚类算法.

关键词: 半监督聚类;文档聚类;主动学习;成对约束

中图法分类号: TP181 **文献标识码:** A

文档聚类是将文档集划分为若干个聚类,使得在同一个聚类中的文档尽可能相似,在不同聚类中的文档尽可能相异.到目前为止,文档聚类已经广泛应用于数据挖掘[1]、信息检索[2]和主题检测[3]等领域.关于文档聚类的研究包括文献[4–6].传统的文档聚类算法是一种无监督的学习方法,即处理的文档都是没有标签的.但是在实际应用中,有时可以获得少量有关数据的先验知识,包括类标签和文档的划分约束条件(比如成对约束信息)等.如何利用这些仅有的先验知识来对大量没有先验知识的文档进行聚类分析,成为一个非常有意义的问题.半监督文档聚类就是针对这类问题提出的,即研究如何利用少量具有先验知识的数据辅助无监督的文档聚类.近几年来,半监督文档聚类逐渐成为机器学习和数据挖掘领域研究的热点问题,国内外学者提出了各种各样的算法,代表性的算法包括文献[7–10]等,文献[11]对已有的算法进行了总结.

在已提出的大部分半监督文档聚类算法中,少量的监督信息都是通过被动的方式给出的.因此,如何获得更有价值的监督信息成为提高聚类性能的关键.但是,通过检验所有可能的监督信息而获得最有价值的监督信息是不现实的.一种解决该问题的方法是,在聚类过程中主动地选取监督信息,通过用户的反馈而获得对提高聚类性能更有帮助,更有价值的信息.

在本文中,使用的监督信息的形式是 Wagstaff 等人给出的两种成对约束:must-link 和 cannot-link[7].首先,提出一种半监督的密度聚类算法(constrained DBSCAN,简称 Cons-DBSCAN).算法的基本思想是,在 DBSCAN 算法[12]中加入成对约束,使其能够指导 DBSCAN 的聚类过程,从而提高聚类性能.为了提高成对约束集中所含的信息量,本文还提出一种主动学习算法来选择对提高算法聚类性能更有帮助的约束集,从而达到付出较小的代价来尽可能高地提高聚类性能的目的.在两个真实文档数据集上的实验表明,结合了主动学习策略的 Cons-DBSCAN 算法能够高效地进行文档聚类.

本文第 1 节介绍相关工作,包括半监督聚类算法、主动学习算法以及 DBSCAN 算法的基本思想.第 2 节介绍本文提出的半监督文档聚类算法和主动学习算法.在第 3 节中,用两个文档数据集来测试算法的性能.第 4 节给出结论与展望.

1 相关工作

1.1 半监督聚类算法

半监督聚类算法大致可以分为 3 类:

一类是基于约束的(constraint-based)半监督聚类算法.这类算法利用类标签数据或者成对约束信息改进聚类算法本身.常用的方法有:(1) 通过修改聚类算法的目标函数满足成对约束,已有的算法见文献[13–15];(2) 在聚类过程中遵循约束条件,使得到的聚类结果满足所有成对约束信息,已有的算法见文献[7,8,16];(3) 依据类标签数据初始化聚类参数并约束聚类过程,已有的算法见文献[14,9].

另一类是基于距离的(metric-based 或 distance-based)半监督聚类算法.这类算法利用类标签数据或者成对约束信息学习一种新的距离测度函数来满足约束条件.常用的方法有:(1) 利用成对约束信息和最短路径算法调整距离测度,已有的算法见文献[17–20];(2) 利用成对约束构造最优化问题,通过求解该凸优化问题得到新的距离测度函数[21,22];(3) 利用同类成对约束信息学习新的马氏距离矩阵,利用新得到的距离进行聚类[23];(4) 利用成对约束信息对原始数据进行基于约束的特征投影,在得到的新子空间进行聚类[24].

第 3 类算法是结合这两种基本思想得到的半监督聚类算法,已有的算法见文献[14,15,24].

已有的大部分半监督据聚类算法都是基于 *k*-means 算法的,因此算法不适合识别数据集中非凸形状的聚类.并且,算法需要提前确定数据集中聚类的个数,这一点在实际中也难以做到.因此,本文考虑基于密度的半监督聚类算法解决这两个问题.

1.2 主动学习算法

主动学习在分类算法中的应用是一个研究得比较多的课题,国内外学者已经提出了多种不同的主动学习策略来选择样本.Schohn 等人提出一种主动选择策略,选择最靠近分类边界的最不确定的样本来辅助分类[25].Tong 等人提出的主动学习方法选择那些使变形空间减小的样本[26].在文献[27,28]中,主动学习策略是选择那些使泛化误差最小的样本.文献[29,30]中研究了属性层面的主动学习方法.

到目前为止,国内外学者对主动学习在聚类中应用的研究比较少.Basu 等人在文献[10]中提出一种结合主动学习的半监督聚类算法 PCKMeans.其主动学习策略是一种两阶段的基于最远优先的主动学习算法,该主动学习算法的缺点是对数据集中的噪声点比较敏感.Huang 等人在文献[31]中提出一种结合主动学习的半监督文档聚类算法.在该算法中,每个聚类用一个中心点表示,使用一个称为 Explore 的步骤搜索每个聚类的邻居.当某个聚类中的样本个数较多或者每个聚类的样本个数差别较大时,该方法就会失效.Huang 等人还在文献[32]中提出一种主动学习算法,通过用户的反馈来选择含信息量大的文档样本对.由于在每次主动选择中算法需要考虑全部的词与词(term-term)之间的相互关系,因此算法的时间效率不能得到保证.

1.3 DBSCAN算法概述

DBSCAN(density based spatial clustering of applications with noise)[12]算法是一种有代表性的基于密度的聚类方法,它根据一个密度阈值控制聚类的增长,将具有足够高密度的区域划分为聚类,并可在带有噪声的空间数据库中发现任意形状的聚类.算法需要两个参数:区域半径 *Eps* 和最小样本数目阈值 Min*Pts*.其主要思想是:一个聚类中每一个点关于一个给定半径 *Eps* 的邻域内必须至少包含 Min*Pts* 的点,即邻域的密度必须超过 Min*Pts*.DBSCAN 中的相关基本概念包括 *Eps*-邻域、核心对象、边界对象、直接密度可达、密度可达、密度连接等.

DBSCAN 首先依次检查数据库中每个点的 *Eps*-邻域,如果某个点的 *Eps*-邻域中至少包含 Min*Pts* 个点,就创建一个以这个点为核心对象的聚类;然后搜寻从该核心对象直接密度可达的样本,并把它们添加到这个核心对象所在的聚类中,其间可能涉及几个密度可达聚类的合并,直到没有新的样本能够添加到任何一个聚类为止.DBSCAN 算法如果采用 *R**-树的数据结构,其理论时间复杂度为 $O(n\log n)$,其中,n 是数据集中的样本个数.如果不采用 *R**-树的数据结构,DBSCAN 算法的时间复杂度为 $O(n^2)$.

DBSCAN 算法不使用任何先验信息,而将选择参数值的任务留给了用户,这对于真正的高维数据集而言,参数的设置通常是依靠经验,难以确定.并且算法的性能依赖于参数 *Eps* 和 Min*Pts* 的选取,当参数不合适时,算法聚类结果很差.因此,我们考虑在 DBSCAN 算法中引入成对约束信息来改善聚类的性能.

2 算 法

本节首先提出一种半监督文档聚类算法 Cons-DBSCAN(constrained DBSCAN),算法在 DBSCAN 中引入成对约束来指导聚类过程,从而达到提高聚类性能的目标.然后,通过对约束集中含有信息量大小的定义和对 DBSCAN 算法本身的分析,提出一种启发式的主动学习算法,能够获取对提高聚类性能作用更显著的成对约束集,来改善 Cons-DBSCAN 算法的聚类性能.

2.1 基于成对约束的半监督文档聚类算法Cons-DBSCAN

在本文中,我们考虑的先验知识的形式是两种成对约束:must-link 和 cannot-link.有 must-link 约束的两个样本要求算法将它们划分到同一个聚类中;有 cannot-link 约束的两个样本要求算法将它们划分在不同的聚类中.

2.1.1 Cons-DBSCAN 算法

在 Cons-DBSCAN 算法中,我们将两个约束集合 C_{ML} 和 C_{CL} 引入算法来提高算法的聚类性能.其中,C_{ML} 是 must-link 的集合,C_{CL} 是 cannot-link 的集合.可知,must-link 约束是在样本集上满足自反、对称、传递关系,即 must-link 是样本集上的一个等价关系.因此,首先求得 must-link 关系等价类的集合 *TCS*(transitive closures sets,简称 TCS),在同一个等价类中的样本在聚类结果中需要满足在同一个聚类中.

为了保证聚类结果满足给定的成对约束,在 Cons-DBSCAN 算法中,用给定的约束集 C_{ML} 和 C_{CL} 指导 DBSCAN 的聚类过程.基本思想是:从某个样本点出发进行扩展时,需要先判断加入待扩展样本是否与已知的成对约束矛盾,在不矛盾时才进行扩展;否则不作本次扩展,而处理其他的样本.通过加入成对约束,可以确保算法在参数不合适的时候仍然能够得到合理的解.Cons-DBSCAN 算法的框架见算法 1.

算法 1. Cons-DBSCAN.

输入:文档集 *D*、must-link 约束集 C_{ML}、cannot-link 约束集 C_{CL}、区域半径 *Eps*、最小样本数目阈值 *MinPts*;

输出:若干聚类和噪声点集.

```
1    初始化样本集 D 中的样本为无标记;
2    从 C_ML 中计算传递闭包 TCS={c_1,c_2,…,c_s};
3    ClusterId:=0;
4    for D 中的每一个样本 p do
         if (p 没有被标记过) then
             if Cons-ExpandCluster (D,p,Eps,MinPts,C_CL,TCS,ClusterId) then
                 ClusterId:=ClusterId+1;
             End if
         End if
     End for
5    End
```

Cons-DBSCAN 是基于 DBSCAN 算法的.Cons-DBSCAN 算法依次检查每个样本,如果当前样本被标记过,则跳过该样本,处理下一个样本;如果当前样本没有被标记过,则算法尝试从该样本出发构造一个新的聚类.如果当前样本是核心对象,则查找从该核心对象密度可达的所有样本,组成一个新的聚类;如果当前样本是边界对象,则暂时将该样本标记为噪声点.Cons-DBSCAN 中最重要的步骤是从样本 *p* 进行聚类扩展的步骤 Cons-ExpandCluster,其框架见算法 2.

算法 2. Cons-ExpandCluster.

输入:文档集 *D*、初始样本 Point、区域半径 *Eps*、最小样本数目阈值 Min*Pts*、cannot-link 约束集 C_{CL}、传递闭包集合 *TCS*、当前聚类标记 ClusterId;

输出:一个布尔状态.

```
1    初始化候选扩展样本队列为空队列,记作 seeds;
2    计算样本 Point 的 Eps-邻域,记作 neighborhood;
3    if neighborhood 中的样本个数小于 MinPts
         暂时把 Point 标记为噪声点;
         return false;
     End if
4    if 样本 Point 属于 TCS 中的某个传递闭包 c_i, then
         把 c_i 中的所有对象标记为 ClusterId 并添加到 seeds 中;
5    else
         将样本 Point 标记为 ClusterId 并添加到 seeds 中;
```

End if

6 while (seeds 不空)

(a) 取出 seeds 中队头元素,记作 seed;

(b) if seed 属于 *TCS* 中的某个传递闭包 c_j, then

for c_j 中没有标记为任何聚类的对象 *o* do

将 *o* 标记为 ClusterId 并添加到 seeds 中;

End for

End if

(c) 计算 seed 的 *Eps*-邻域,记作 neighborhood;

(d) if neighborhood 中的样本个数大于等于 Min*Pts*

for neighborhood 中的样本 *p* do

if 把 *p* 加入到 seeds 中不违反已知的 cannot-link 约束并且 *p* 没有标记, then

将 *p* 标记为 ClusterId 并添加到 seeds;

End if

End for

End if

(e) 从 seeds 中删除 seed;

End while

7 return true;

Cons-ExpandCluster 算法在执行扩展聚类的过程中,通过引入成对约束信息,使得算法能够得到理想的聚类结果:

1) 在步骤 4 中,如果初始扩展的样本 Point 属于 *TCS* 中的某个传递闭包,则该传递闭包中所有的样本都加入到当前聚类中,从而保证满足 must-link 约束;

2) 在步骤 6(b)中,如果候选扩展样本队列中的样本 seed 属于 *TCS* 中的某个传递闭包,则将该传递闭包中的无标记或标记为噪声的样本加入到当前聚类中,从而保证满足 must-link 约束;

3) 在步骤 6(d)中,如果候选扩展样本队列中的样本 seed 是核心对象,则将其 *Eps*-邻域内的样本加入到当前聚类时,需要先判断加入该样本是否违反已知的 cannot-link 约束;如果待加入的样本 *p* 与当前聚类中的某个样本 *q* 构成的样本对 $\{p,q\}\in CL$,则加入该样本 *p* 违反给定的 cannot-link 约束,不执行该操作,从而保证满足 cannot-link 约束.

由于在半监督聚类算法中引入先验知识 must-link 约束和 cannot-link 约束,使得 Cons-DBSCAN 算法的鲁棒性更强.如果参数 *Eps* 的值过小或者 *MinPts* 值过大,DBSCAN 算法会使原本属于同一聚类的两个样本划分在两个不同的聚类中;但是有了 must-link 约束,则可以保证将二者划分在同一个聚类中.如果参数 *Eps* 的值过大或者 *MinPts* 值过小,DBSCAN 算法会使原本属于两个不同聚类的两个样本划分在同一个聚类中,但是有了 cannot-link 约束,可以保证将二者划分在两个不同的聚类中.

2.1.2 与 C-DBSCAN 算法的比较

C-DBSCAN 算法[16]也是通过在 DBSCAN 算法中引入 must-link 和 cannot-link,并在聚类过程中考虑给定的成对约束,从而使最终的聚类结果满足给定的约束集.C-DBSCAN 算法的思想是:首先对样本空间进行递归地划分,直到每个叶结点中含有少于 Min*Pts* 个样本,得到一棵 KD-Tree.然后检索 KD-Tree 中的所有叶结点,并将密度可达的样本识别为若干个局部聚类(local cluster),在此过程中考虑 cannot-link 约束;如果在某个叶结点中的两个样本之间存在 cannot-link 约束,则该叶结点中的全部样本暂时视为噪声点;如果没有 cannot-link 约束,则按照 DBSCAN 算法进行扩展聚类.接着考虑 must-link 约束,进行局部聚类的合并.如果在不同局部聚类中的两个样本之间存在一个 must-link 约束,则将这两个局部聚类合并为核心局部聚类(core local cluster).最后,应用单连接

的凝聚层次聚类算法合并核心局部聚类,得到最终的聚类结果.在此过程中,考虑 cannot-link 约束,如果待合并的核心局部聚类中的两个样本之间存在 cannot-link 约束,则不执行此次合并.

与 C-DBSCAN 算法相比,Cons-DBSCAN 算法在执行效率和聚类结果性能上都有明显的优势.

假设样本个数是 n,样本的维数是 m.在最理想情况下,KD-Tree 是一个完全 2^m-叉树,并且每个叶结点中有(MinPts−1)个样本.可求得该树中叶结点的个数约为 $\frac{n}{\text{Min}Pts-1}$,树的深度约为 $\frac{1}{m}\log_2\left(\frac{n}{\text{Min}Pts-1}\right)$.由于 $m<<n$,Min$Pts<<n$,m 和 MinPts 相对 n 可视为常数.因此,树中结点总数的量级为 $O(n)$,即 C-DBSCAN 算法比 Cons-DBSCAN 算法至少多用 $O(n)$的存储单元.

在 Cons-DBSCAN 算法中,聚类的扩展过程与 DNSCAN 算法类似,只是在扩展的时候要求不破坏给定的 must-link 约束和 cannot-link 约束.在时间性能方面,Cons-DBSCAN 算法与 DBSCAN 算法相当,时间复杂度是 $O(n^2)$(不采用 R^*-树数据结构的情况).对于 C-DBSCAN 算法,m 和 MinPts 相对 n 可视为常数,则构造 KD-Tree 的时间复杂度为 $O(n\log n)$.在最坏情况下,构造局部聚类的过程与 DNSCAN 相同,时间复杂度是 $O(n^2)$.考虑 must-link 约束,进行局部聚类合并过程的时间复杂度与 must-link 约束的个数相同,可视为常数时间复杂度 $O(1)$.应用单连接的凝聚层次聚类算法,合并核心局部聚类的过程要计算两两聚类之间的距离,该过程的时间复杂度是 $O(n^2)$.综上所述,C-DBSCAN 算法总的时间复杂度是 $O(n^2)$.尽管与 Cons-DBSCAN 算法的时间复杂度在量级上相同,但是由分析可知,Cons-DBSCAN 算法比 C-DBSCAN 算法的时间复杂度的常数因子要小,第 3 节实验部分验证了我们的分析.

在聚类性能方面,Cons-DBSCAN 算法是在 DBSCAN 的执行过程中同时考虑给定的 must-link 和 cannot-link 约束,能够保证聚类满足所有给定的成对约束.C-DBSCAN 算法采用 3 个独立的步骤使算法满足给定的成对约束.其中,前两个步骤的处理方式会影响聚类结果的性能.在检索 KD-Tree 中的所有叶结点构造局部聚类时,如果在某个叶结点中的两个样本之间存在 cannot-link 约束,该叶结点中的全部样本都视为噪声点.这样处理会使一些原本属于同一个聚类的样本,由于其他样本之间的 cannot-link 约束,而被划分在不同的聚类中.在进行局部聚类合并时,由于只考虑了 must-link 约束而没有同时考虑 cannot-link 约束,在合并时有可能产生矛盾.因为在不同局部聚类中的两个样本之间存在一个 must-link 约束,但是在其他样本之间可能存在 cannot-link 约束,合并的结果会破坏这一类的 cannot-link 约束.由以上的分析可知,C-DBSCAN 算法处理成对约束的方法会影响到聚类结果的性能,同时有可能破坏一些成对约束.因此,相对于 C-DBSCAN 算法,Cons-DBSCAN 算法在聚类结果方面性能更优.

2.2 主动学习算法

主动学习的目标是选择含信息量较多的成对约束,能够用尽可能少的约束信息来尽量多地提高聚类性能.Davidson 等人在文献[33]中使用 *Informativeness* 定性地评价一个约束集含信息量的大小. *Informativeness* 是与特定的算法相联系的,一个约束集所含信息量的大小定义为该算法本身不能确定的那部分信息的多少.

给定 *Eps* 和 MinPts,DBSCAN 对处于聚类边界的样本和两个或多个聚类重叠部分的样本不能有效地识别,如图 1 所示.

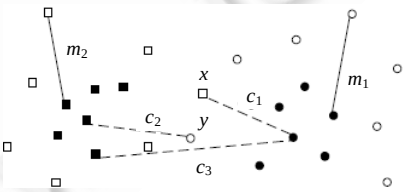


Fig.1 An example which is difficult for DBSCAN to process

图 1 DBSCAN 难处理的实例

图 1 中,不同的形状表示不同的聚类;实心点表示核心点,空心点表示边界点;must-link 用实线表示,cannot-link 用虚线表示.

可以看出,图 1 中的两个聚类有部分重叠,并且对于处于聚类边界的两个样本 x 和 y,DBSCAN 算法本身难以正确识别,需要两个 cannot-link 约束 c_1 和 c_2 去处理两个聚类的重叠部分.另外,还需要两个 must-link 约束 m_1 和 m_2 来控制聚类的边界.

由上面的分析可得,对于 DBSCAN 算法,一个 *Informativeness* 大的约束集需要满足两个条件:1) 每个聚类中至少有一个元素出现在约束集中;2) 必须含有相应的成对约束来控制聚类的边界.

基于以上分析,我们提出一种启发式的主动学习算法.算法需要两个参数:*Eps* 和 *MinPts*,其意义与 DBSCAN 两个参数相同.在实际中应用时,与 Cons-DBSCAN 中的相应参数设置为相同的值.流程见算法 3.

算法 3. Active-Selecting.

输入:文档集 D、区域半径 *Eps*、最小样本数目阈值 Min*Pts*、成对约束数目 Q、判定成对约束类别的领域专家;

输出:成对约束集.

1 根据给定的 *Eps* 和 Min*Pts* 确定核心点集和边界点集,核心点集和边界点集分别记作 *CS* 和 *BS*;
2 初始化成对约束集 *ConsSet* 为空集;
3 初始化已选择的核心点集 *SCS* 为空集;
4 while 已获得的成对约束个数小于 Q
(a) if *SCS* 是空集, then
从 *CS* 中随机选择一个核心点 x,并添加到 *SCS* 中;
(b) else
从 *CS* 中选择距离 *SCS* 最远的核心点 x;
for *SCS* 中的每个样本 y, do
构造成对约束集$\{x,y\}$,并由领域专家判定其类别;
将成对约束$\{x,y\}$添加到 *ConsSet* 中;
End for
(c) End if
(d) 在 *BS* 中选择距离核心点 x 最远的边界点 b_1;
构造成对约束集$\{x,b_1\}$,并由领域专家判定其类别;
将成对约束$\{x,b_1\}$加入 *ConsSet* 中;
(e) 在 *BS* 中选择距离核心点 x 最近的边界点 b_2;
构造成对约束集$\{x,b_2\}$,并由领域专家判定其类别;
将成对约束$\{x,b_2\}$加入 *ConsSet* 中;
(f) End while
5 return *ConsSet*;
6 End

在算法 3 的步骤 4(b)中,我们采用最远优先的策略选择核心点,其中,核心点 x 与核心点集 *SCS* 之间的距离定义为 $d(x,SCS)=\min_{y\in SCS}d(x,y)$,两个样本之间的距离采用余弦距离.文献[10]已经证明,按照最远优先的策略选择核心点,可以在足够小的尝试次数下,在每个聚类中至少可以选择一个核心点,从而可以满足每个聚类中至少有一个元素出现在约束集中.

在算法的步骤 4(d)和步骤 4(e)中,分别选择距离核心点 x 最远的和最近的一个边界点,这样选择的约束可以很好地控制聚类的边界.

3 实 验

3.1 测试数据集

在本节中,我们使用两个实际的文档数据集来测试算法的性能.

第 1 个是新闻分类数据集 20-Newsgroups,该数据集中包含 20 类不同的新闻信息,每类中包含 1 000 条不同的新闻.为了实验结果比较的公平性,数据选取方法与 Basu 等人在文献[10]中的做法相同,从每一类新闻集中随机选取 100 条新闻构造一个包含 20 类新闻的数据集,记作 News-all20.News-all20 中含有 2 000 个样本,维度是 16 089.从 News-all20 数据集中分别选出 3 类构造两个数据集:News-sim3 和 News-diff3.其中,News-sim3 数据集中包含 3 个相近的主题(comp.graphics, comp.os.ms-windows, comp.window.x),该数据集中各类之间有较大的重叠;News-diff3 数据集中包含 3 个不同的主题(alt.atheism, rec.sport.baseball, sci.space),该数据集中各类之间的边界比较清晰.经过处理后,News-sim3 含有 300 个样本,维度是 3 225;News-diff3 含有 300 个样本,维度是 3 251.

第 2 个数据集是 TDT5,该数据集来自话题检测和追踪项目(topic detection and tracking),包含多种渠道和语言的新闻语料.TDT5 数据集包含从 2003 年 4 月~2003 年 9 月的新闻,其中的语言包括英语、普通话和阿拉伯语等.在实验中,我们只使用其中的标有类别的英文文档.经过处理后,TDT5 中有 3 905 个样本,包含 40 个主题,维度是 19 325.

对于以上两个文档数据集,采用 Dhillon 等人[34]给出的预处理方法进行处理,并除去高频词和低频词,因为高频词和低频词对识别每个聚类的作用不明显.对于数据集 News-all20,News-sim3 和 News-diff3,高频词和低频词的阈值设置分别为 100 和 1;对于数据集 TDT5,高频词和低频词的阈值设置分别为 300 和 1.

3.2 评价指标

采用归一化互信息(normalized mutual information,简称 NMI)[35]和成对 F 测度(pairwise f-measure)[10,31,32]作为评价聚类结果的指标.NMI 是一种外部评价标准,它用来评价在一个数据集上的聚类结果与该数据集的真实划分的相似程度.NMI 是介于 0~1 之间的值.NMI 的值越大,说明聚类性能越好.成对 F 测度是综合准确率和召回率评测标准后的评价指标,成对 F 测度的值介于 0~1 之间,并且其值越大,说明聚类性能越好.

3.3 实验方法

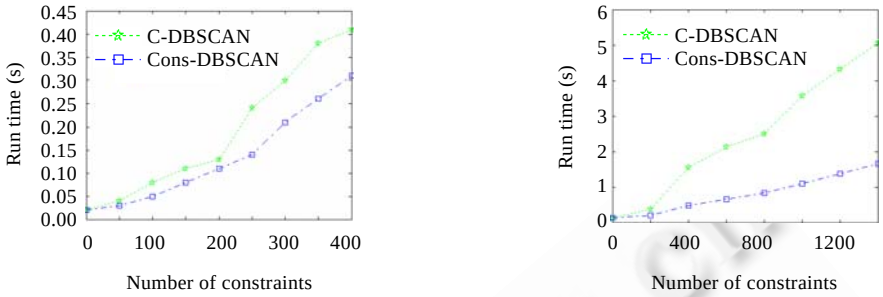
在本节中,对于主动 Cons-DBSCAN 算法(active cons-DBSCAN),其成对约束集按照第 2.2 节中给出的主动学习策略选取;对于非主动的 Cons-DBSCAN(简记为 Cons-DBSCAN),其成对约束集随机选取.

本节的实验包括 5 部分:

1) 比较 Cons-DBSCAN 与 C-DBSCAN 的性能;
2) 比较主动 Cons-DBSCAN 与 Cons-DBSCAN 的性能;
3) 比较主动 Cons-DBSCAN 算法与两个代表性的结合主动学习的半监督文档聚类算法的性能,其中,Basu 的算法[10]记作 Active PCKMeans,Huang 的算法[32]记作 Active LM;
4) 比较本文给出的主动学习策略与文献[10]提出的最远优先主动学习策略;
5) 比较主动 Cons-DBSCAN 和 Active LM 的时间效率.

对于算法中的参数 *Eps* 和 Min*Pts*,实验中按照文献[12]给出的方法确定.在 Cons-DBSCAN 和 C-DBSCAN 的性能比较中,每次实验中两算法采用完全相同的约束集.考虑到 Cons-DBSCAN 和 C-DBSCAN 中的约束集是随机产生的,对于每个数据集运行 Cons-DBSCAN 和 C-DBSCAN20 次,取 20 次的平均值作为最终的统计结果.另外,对于 Active PCKMeans 和 Active LM 算法,聚类个数 k 设为数据集中的真实类别个数.对于每个数据集,在实验中产生的约束集都是混合约束集,即包括 must-link 约束和 cannot-link 约束.实验结果如图 2~图 8 所示,其中,图 2 是 Cons-DBSCAN 与 C-DBSCAN 时间性能比较的实验结果,图 3 是 Cons-DBSCAN 与 C-DBSCAN 聚类结

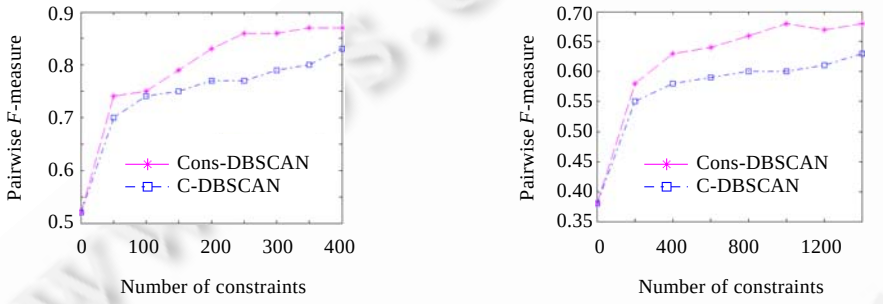
果性能比较的实验结果,图 4~图 6 是 Cons-DBSCAN、主动 Cons-DBSCAN、Active PCKMeans 和 Active LM 分别在数据集 News-diff3,News-all20 和 TDT5 上的实验结果,图 7 是在数据集 News-sim3 上主动策略比较的实验结果,图 8 是主动 Cons-DBSCAN 和 Active LM 时间效率比较的实验结果.



(a) News-diff3 (b) News-all20

Fig.2 Time efficiency comparison results

图 2 时间性能比较结果



(a) News-diff3 (b) News-all20

Fig.3 Clustering effectiveness comparison results

图 3 聚类性能比较结果

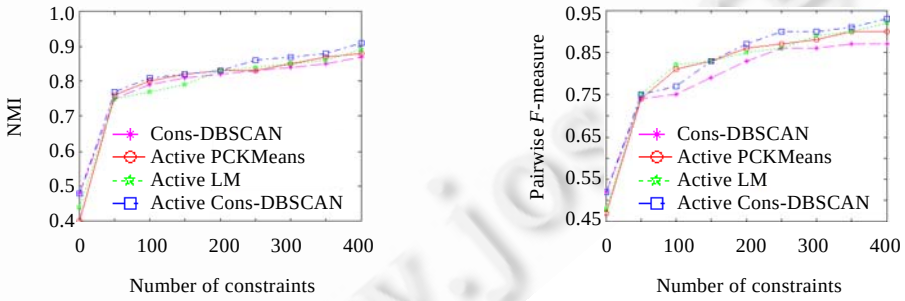


Fig.4 Comparison results on News-diff3

图 4 在 News-diff3 上的比较结果

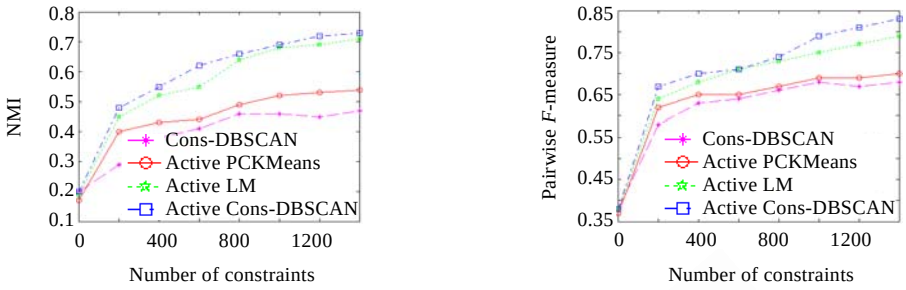


Fig.5 Comparison results on News-all20

图 5 在 News-all20 上的比较结果

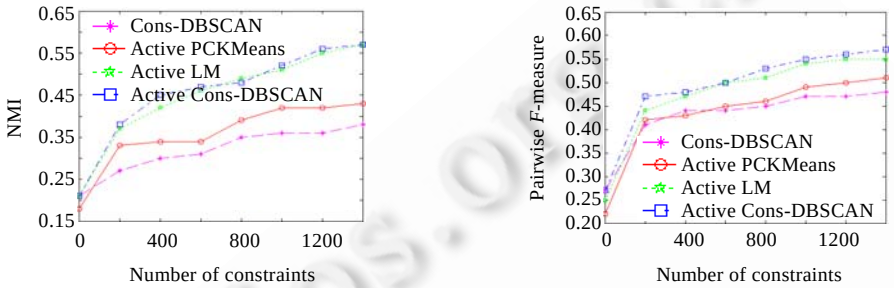


Fig.6 Comparison results on TDT5

图 6 在 TDT5 上的比较结果

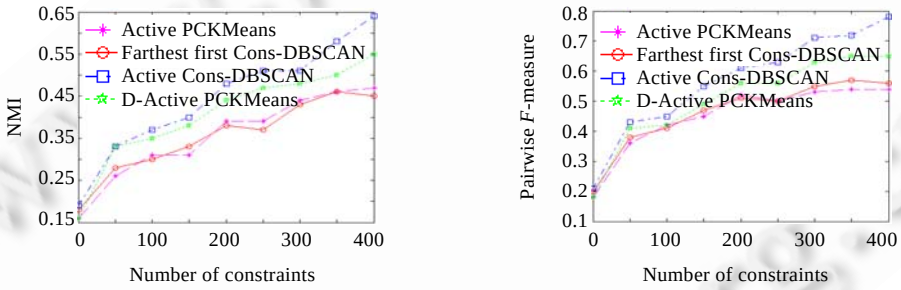


Fig.7 Active strategy comparison results on News-sim3

图 7 在 News-sim3 上的主动策略比较结果

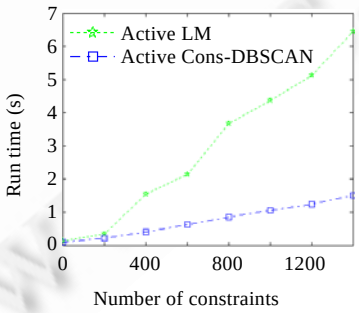


Fig.8 Time complexity results on News-all20

图 8 在 News-all20 上的时间复杂度比较结果

3.3.1 Cons-DBSCAN 与 C-DBSCAN 的性能比较

由于篇幅所限,本节只展示了 Cons-DBSCAN 与 C-DBSCAN 在两个数据集 News-diff3 和 News-all20 上的实验结果;并且在聚类性能比较中,只展示了两种算法在成对 F 测度上的比较结果.当成对约束个数为 0 时,Cons-DBSCAN 和 C-DBSCAN 都退化为 DBSCAN 算法,此时,二者的执行结果相同.

从图 2 中可以看到,Cons-DBSCAN 算法的时间效率优于 C-DBSCAN 算法.并且,当样本集和成对约束集越大时,这种优势越明显.这是因为,C-DBSCAN 算法中构造 KD-Tree 的步骤,当样本集越大时,这一步骤执行时间越长.并且,C-DBSCAN 算法采取 3 个独立的步骤,使聚类结果满足给定的成对约束,其中涉及多次成对约束的判断.因此当约束集越大时,算法的执行时间越长.Cons-DBSCAN 算法的处理方法与之不同,Cons-DBSCAN 算法只需执行一次 DBSCAN 扩展聚类的过程,在其中同时考虑给定的成对约束,利用成对约束的效率高.因此,Cons-DBSCAN 算法的时间效率优于 C-DBSCAN 算法.

图 3 给出了两个算法在成对 F 测度上的比较结果.由于在实验中两个算法采用的是随机产生的约束集,没有考虑每个约束所含信息量的大小,因此二者利用成对约束对聚类性能的提高有限.但是与 C-DBSCAN 算法相比,Cons-DBSCAN 算法还是有微弱的优势.原因在于两算法利用成对约束的方式不同,C-DBSCAN 算法在构造局部聚类时,如果在某个叶结点中的两个样本之间存在 cannot-link 约束,则该叶结点中的全部样本暂时视为噪声点.这样简单的处理方式,会使一些本来属于同一个聚类的样本划分在不同的聚类中.进行局部聚类合并时,只考虑了 must-link 约束,没有同时考虑 cannot-link 约束,这样在合并时有可能产生矛盾.因为在不同局部聚类中的某两个样本之间存在一个 must-link 约束,但是在其他样本之间可能会存在 cannot-link 约束,合并的结果会破坏这一类 cannot-link 约束.在实验中,C-DBSCAN 算法的聚类结果就存在破坏这一类 cannot-link 约束的情况.Cons-DBSCAN 算法执行扩展聚类的过程中,同时考虑给定的 must-link 约束和 cannot-link 约束,这样可以保证聚类结果不破坏任何给定的成对约束.因此,Cons-DBSCAN 算法比 C-DBSCAN 算法的聚类性能更好.

由以上的实验分析可知,无论是在执行效率还是在聚类性能方面,Cons-DBSCAN 算法比 C-DBSCAN 算法都表现得更好,也验证了第 2.1.2 节中对两种算法的比较分析.

3.3.2 主动 Cons-DBSCAN 与 Cons-DBSCAN 的比较

在实验中,主动 Cons-DBSCAN 和 Cons-DBSCAN 都能准确识别出每个数据集中的所有聚类.从图 4~图 6 中可以看出,主动 Cons-DBSCAN 算法的聚类性能明显优于 Cons-DBSCAN.总的来说,对于主动 Cons-DBSCAN,随着成对约束的个数增多,算法的聚类性能越来越好.对于 Cons-DBSCAN 则不然,从图 5 中可以看到,在数据集 News-all20 上,当成对约束个数从 1 000 增加到 1 200 时,聚类性能反而有所下降.这是因为在随机产生的成对约束中,有些约束会对聚类性能有损害作用.这一结果在文献[33]中也有讨论.通过采用本文提出的主动学习策略,选择的成对约束对算法来说都是含信息量高的成对约束,因此,主动 Cons-DBSCAN 算法的性能随着成对约束的个数增加而越来越好.

当成对约束的个数等于 0 时,Cons-DBSCAN 退化为 DBSCAN 算法.从实验结果可以看出,通过引入成对约束,Cons-DBSCAN 能够显著地提高聚类的性能.

3.3.3 主动 Cons-DBSCAN 与 Active PCKMeans 和 Active LM 的比较

与 Active PCKMeans 相比,主动 Cons-DBSCAN 对聚类性能的提高更显著.具体来说,对于各个聚类之间边界比较清晰的样本集 News-diff3,两个算法的聚类结果差别不大.但是,对于聚类边界有较多重叠的样本集,比如 News-all20 和 TDT5,无论是 NMI 还是成对 F 测度,主动 Cons-DBSCAN 的聚类性能比 Active PCKMeans 明显要好.这是因为通过采用文中给出的主动学习策略,成对约束集中所含信息能够更好地控制聚类的边界.因此,主动 Cons-DBSCAN 可以更高效地处理聚类边界有重叠的样本集.

从图 4~图 6 中可以看出,Active LM 也能够显著地提高聚类性能,并且引入的成对约束越多,聚类的性能越好.实验结果表明,在绝大多数数据集上,Active LM 算法的性能都要优于 Active PCKMeans.但是总体来说,Active LM 算法的性能比主动 Cons-DBSCAN 略差.这是因为,在我们提出的主动学习策略中,采取了专门的步骤来控制识别数据集中聚类的边界.因此,得到的成对约束集能够更有效地辅助半监督聚类过程.采用了这些高质

量的成对约束,主动 Cons-DBSCAN 算法就能够更显著地提高聚类性能.

3.3.4 主动学习策略的比较

本节比较本文提出的主动学习策略和文献[10]中采用的最远优先主动学习策略.采用的数据集是 News-sim3 数据集,这是因为 News-sim3 中包含有重叠的类,并且所包含的 3 个类相似,所以不容易区分.在实验中,我们将本文提出的主动学习策略和最远优先策略分别应用在 Cons-DBSCAN 和 PCKMeans 两个半监督聚类算法中.其中,采用最远优先策略的 Cons-DBSCAN 记作 Farthest first Cons-DBSCAN,采用本文中的主动学习策略的 PCKMeans 记作 D-Active PCKMeans.

从图 7 中可以看出,主动 Cons-DBSCAN 的性能明显优于 Farthest first Cons-DBSCAN.采用最远优先策略时,Cons-DBSCAN 的聚类性能并不总是随着成对约束个数的增加而提高.这是因为最远优先策略对噪声点比较敏感,数据集中的噪声点较多时,Farthest first Cons-DBSCAN 算法的聚类结果并不理想.相反,采用文中提出的主动学习算法,得到的成对约束集能够更有效地指导聚类过程,用它们辅助 Cons-DBSCAN 算法可以得到更理想的聚类结果.

与 Active PCKMeans 算法相比,D-Active PCKMeans 算法的聚类性能更好.这是因为采用本文中提出的主动学习策略,得到的成对约束集更好地控制了数据集中聚类的边界,即所含信息量更大.因此,采用这些约束集来辅助聚类能够更显著地提高聚类性能.但是从结果中可以看到,D-Active PCKMeans 的聚类性能比主动 Cons-DBSCAN 略差.原因可能是我们提出的主动学习策略是基于对 DBSCAN 算法的观察和分析,因此本文提出的主动学习算法与 Cons-DBSCAN 结合,聚类的性能更好.

3.3.5 时间效率的比较

我们采用 News-all20 数据集比较主动 Cons-DBSCAN 和 Active LM 的时间效率,实验结果如图 8 所示.随着成对约束个数的增加,主动 Cons-DBSCAN 的运行时间增长缓慢.但是对于 Active LM 算法,情况则不同.当成对约束个数大于 200 时,Active LM 的运行时间迅速增长.原因是:一方面,在其主动学习算法中,每选取一个成对约束,需要考虑词和词之间的相互关系,计算量大;另一方面,在每次迭代中,成对约束的获取过程和半监督文档聚类过程要交替执行.对于主动 Cons-DBSCAN 中,只需要一次执行就可以得到全部的成对约束集.因此,主动 Cons-DBSCAN 可以高效地完成文档聚类过程.

4 结论与展望

本文提出一种结合主动学习的半监督文档聚类算法.对于无监督聚类算法 DBSCAN,通过引入先验信息 must-link 约束集和 cannot-link 约束集来指导 DBSCAN 的聚类过程,使能够提高聚类性能,得到一种半监督文档聚类算法 Cons-DBSCAN.通过对约束集中所含信息量的衡量和对 DBSCAN 算法本身的分析,提出了一种的启发式的主动学习算法,每次选取含信息量大的成对约束集,从而能够更高效地辅助半监督聚类算法 Cons-DBSCAN.实验结果表明,本文提出的算法能够高效地进行文档聚类.通过主动学习算法获得的成对约束,能够显著地提高聚类的性能.并且,文中算法的性能优于两个代表性的结合主动学习的主动半监督聚类算法——Active PCKMeans 和 Active LM.

关于结合主动学习的半监督文档聚类,有些问题还需要进一步研究:第一,研究不同的成对约束如何影响聚类性能;第二,研究文档对之间的相互关系,从而能够在已选择的成对约束的基础上选取对聚类过程辅助更大的成对约束.

References:

[1] Han J, Kamber M. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, 2001.

[2] Frakes WB, Baeza-Yates R. Information Retrieval: Data Structure and Algorithms. Prentice-Hall PTR, 1992.

[3] Allan J. Topic Detection and Tracking: Event-Based Information Organization. 2002.

[4] Hu XH, Zhang XD, Lu CM, Park EK, Zhou XH. Exploiting wikipedia as external knowledge for document clustering. In: Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. 2009. 389−396. [doi: 10.1145/1557019.1557066]

[5] Zheng HT, Kang BY, Kim HG. Exploiting noun phrases and semantic relationships for text document clustering. Information Sciences, 2009,179(13):2249−2262. [doi: 10.1016/j.ins.2009.02.019]

[6] Mahdavi M, Abolhassani H. Harmony *K*-means algorithm for document clustering. Data Mining and Knowledge Discovery, 2009, 18(3):370−391. [doi: 10.1007/s10618-008-0123-0]

[7] Wagstaff K, Cardie C. Clustering with instance-level constraints. In: Proc. of the 17th Int'l Conf. on Machine Learning. 2000. 1103−1110.

[8] Wagstaff K, Cardie C, Rogers S, Schroedl S. Constrained *k*-means clustering with background knowledge. In: Proc. of the 18th Int'l Conf. on Machine Learning. 2001. 577−584.

[9] Basu S, Banerjee A, Mooney RJ. Semi-Supervised clustering by seeding. In: Proc. of the 9th Int'l Conf. on Machine Learning. 2002. 19−26.

[10] Basu S, Banerjee A, Mooney RJ. Active semi-supervision for pairwise constrained clustering. In: Proc. of the SIAM Int'l Conf. on Data Mining. 2004. 333−344.

[11] Li KL, Cao Z, Cao LP, Zhang C, Liu M. Some developments on semi-supervised clustering. Pattern Recognition and Artificial Intelligence, 2009,22(5):735−742 (in Chinese with English abstract).

[12] Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. of the 2nd Int'l Conf. on Knowledge Discovery and Data Mining. 1996. 226−231.

[13] Demiriz A, Bennett KP, Embrechts MJ. Semi-Supervised clustering using genetic algorithms. In: Proc. of the Artificial Neural Networks in Engineering Conf. 1999. 809−814.

[14] Bilenko M, Basu S, Mooney RJ. Integrating constraints and metric learning in semi-supervised clustering. In: Proc. of the 21st Int'l Conf. on Machine Learning. 2004. 81−88. [doi: 10.1145/1015330.1015360]

[15] Basu S, Bilenko M, Mooney RJ. A probabilistic framework for semi-supervised clustering. In: Proc. of the 10th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. 2004. 59−68. [doi: 10.1145/1014052.1014062]

[16] Ruiz C, Spiliopoulou M, Menasalvas E. C-DBSCAN: Density-Based clustering with constraints. In: Proc. of the Rough Sets, Fuzzy Sets, Data Mining and Granular Computing. LNCS 4482, 2007. 216−223. [doi: 10.1007/978-3-540-72530-5_25]

[17] Kamvar SD, Klein D, Manning C. Spectral learning. In: Proc. of the 18th Int'l Joint Conf. on Artificial Intelligence. 2003. 561−566.

[18] Xu QJ, Desjardins M, Wagstaf K. Constrained spectral clustering under a local proximity structure assumption. In: Proc. of the 18th Int'l Conf. of the Florida Artificial Intelligence Research Society. 2005. 866−867.

[19] Klein D, Kamvar SD, Manning C. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In: Proc. of the 19th Int'l Conf. on Machine Learning. 2002. 307−314.

[20] Wang L, Bo LF, Jiao LC. Density-Sensitive semi-supervesed spectral clustering. Journal of Software, 2007,18(10):2412−2422 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/18/2412.htm [doi: 10.1360/jos182412]

[21] Xing EP, Ng AY, Jordan MI, Russell S. Distance metric learning with application to clustering with side-information. Advances in Neural Information Processing Systems 15. 2003. 505−512.

[22] Schultz M, Joachims T. Learning a distance metric from relative comparisons. Advances in Neural Information Processing Systems. 2003. 40−47.

[23] Bar-Hillel A, Hertz T, Shental N, Weinshall D. Learning distance functions using equivalence relations. In: Proc. of the 20th Int'l Conf. on Machine Learning. 2003. 11−18.

[24] Tang W, Xiong H, Zhong S, Wu J. Enhancing semi-supervised clustering: A feature projection perspective. In: Proc. of the 13th Int'l Conf. on Knowledge Discovery and Data Mining. 2007. 707−716. [doi: 10.1145/1281192.1281268]

[25] Schohn G, Cohn D. Less is more: Active learning with support vector machines. In: Proc. of the 17th Int'l Conf. on Machine Learning. 2000. 839−846.

[26] Tong S, Koller D. Support vector machine active learning with applications to text classification. In: Proc. of the 17th Int'l Conf. on Machine Learning. 2000. 287−295. [doi: 10.1162/153244302760185243]

[27] Roy N, McCallum A. Toward optimal active learning through sampling estimation of error reduction. In: Proc. of the 18th Int'l Conf. on Machine Learning. 2001. 441−448.

[28] Sugiyama M. Active learning in approximately linear regression based on conditional expectation of generalization error. Journal of Machine Learning Research, 2006,7:141−166.

[29] Raghavan H, Madani O, Jones R. Active learning with feedback on both features and instances. Journal of Machine Learning Research, 2006,7:1655−1686.

[30] Veeramachaneni A, Olivetti E, Avesani P. Active sampling for detecting irrelevant features. In: Proc. of the 23rd Int'l Conf. on Machine Learning. 2006. 961−968. [doi: 10.1145/1143844.1143965]

[31] Huang RZ, Lam W, Zhang Z. Active learning of constraints for semi-supervised text clustering. In: Proc. of the SIAM Int'l Conf. on Data Mining. 2007. 113−124.

[32] Huang RZ, Lam W. An active learning framework for semi-supervised document clustering with language modeling. Data and Knowledge Engineering, 2009,68(1):49−67. [doi: 10.1016/j.datak.2008.08.008]

[33] Davidson I, Wagstaff KL, Basu S. Measuring constraints-set utility for partitional clustering algorithms. In: Proc. of Conf. on Principles and Practice of Knowledge Discovery in Databases. 2006. 115−126. [doi: 10.1007/11871637_15]

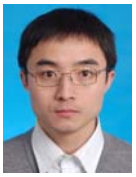
[34] Dhillon IS, Modha DS. Concept decompositions for large sparse text data using clustering. Machine Learning, 2001,42(1):143−175. [doi: 10.1023/A:1007612920971]

[35] Strehl A, Ghosh J, Mooney R. Impact of similarity measures on Web-page clustering. In: Proc. of the Workshop on Artificial Intelligence for Web Search. 2000. 58−64.

附中文参考文献:

[11] 李昆仑,曹铮,曹丽苹,张超,刘明.半监督聚类若干新进展.模式识别与人工智能,2009,22(5):735−742.

[20] 王玲,薄列峰,焦李成.密度敏感的半监督谱聚类.软件学报,2007,18(10):2412−2422. http://www.jos.org.cn/1000-9825/18/2412.htm [doi: 10.1360/jos182412]



赵卫中(1981－),男,山东菏泽人,博士,讲师,CCF 会员,主要研究领域为机器学习,数据挖掘,算法分析与设计.

李志清(1975－),男,博士,副教授,CCF 会员,主要研究领域为图像理解,机器学习,视觉信息挖掘.

马慧芳(1981－),女,博士,副教授,主要研究领域为机器学习,数据挖掘.

史忠植(1941－),男,研究员,博士生导师,CCF 高级会员,主要研究领域为人工智能,机器学习,神经计算,认知科学.