基于ontology抽取优化初始选择的检索结果聚类

陈毅恒, 秦 兵, 宋 凡, 刘 挺, 李 生
(哈尔滨工业大学计算机科学与技术学院信息检索研究室, 黑龙江哈尔滨 150001)

摘 要: 本文针对互联网的数据量的不断增加, 准确搜索引擎的作用日益困难的问题, 为了提高搜索引擎返回结果结构化聚类的效果, 让信息的定位更迅速, 本文采用基于标签的聚类算法, 并使用自然语言处理技术中的依存句法分析和词典资源, 深度挖掘语义结构, 提出基于优化初始选择的K均值聚类方法. 本文深入分析K均值聚类算法特点, 并利用类别标签技术对该算法进行有效改进. 实验证明该算法不仅在效果上优于一般聚类算法, 对结果描述也有很大帮助, 在效率上也得到很大提高.

关键词: 检索结果聚类; ontology; 标签;
中图分类号: TP391.2 文献标识码: A 文章编号: 0372-2112 (2008) 12A-166-05

Search Result Clustering Based on Centroid Optimization by Ontology Extraction

CHEN Yi-heng, QIN Bing, SONG Fan, LIU Ting, LI Sheng
(*School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China*)

Abstract: Along with the constant development of the Internet and the ever-increasing amount of data, the role of search engines has become increasingly evident. More users rely on search engines to find the information needed. In order to more effectively cluster the search results, thus facilitating the positioning of information among the original unstructured results, a new label-based clustering algorithm is introduced in this paper. The key idea is to use the dictionary resource and Dependency Syntax Parsing in NLP to extract the ontologies related to the query. These extracted ontologies will further guide the choosing of centroids in K-means clustering. Furthermore, the various features of K-means algorithm have been fully investigated, and a way of improvement is proposed by using the cluster labels. Experiments show that this algorithm not only yields more effective cluster results but also provides more informative descriptions of the results; meanwhile, the efficiency has also been largely improved.

Key words: search results clustering; ontology; label

1 引言

搜索引擎作为web用户查询和浏览信息的重要工具, 以google、百度为代表的搜索引擎往返回的结果使得用户查找相关信息变得越来越困难. 例如: 输入查询"苹果"返回结果中会混杂作为"一种水果"、"电脑品牌"、"音乐网站"多种类别的结果; 即使查询为"苹果电脑"同样包含"苹果品牌机"、"苹果笔记本"、"苹果服务器"等信息需要用户在返回结果中进行再次查询. 近几年开始了关于用户可以在感兴趣的簇中查看结果. 检索结果聚类[4]还对文本挖掘, 信息抽取和自动问答等领信息搜索引擎返回结果的研究, 通过对检索结果进行聚类[1], 将其分成若干簇(clusters), 域起着重要的作用. 有些搜索引擎已经将聚类技术集成在一起, 如Vivisimo[2]、Carrot2[3]、WEBCAT[4]等. Grouper2是最早对搜索结果在线聚类的系统; Vivisimo中搜索apachelucene也取得了不错的效果; 而Carrot2是近期很成功的聚类搜索引擎框架. 本文利用ontology抽取标签, 采用基于标签的聚类方法, 利用标签技术优化聚类算法, 提出基于初始选择的k-means聚类算法. 本文首先介绍现有检索结果聚类方法的特点, 及本文提出方法的优点; 并给出系统框架及获取标签的方法和基于标签的聚类算法; 最后给出实验结果和结论.

2 研究现状

经典的文档聚类算法有层次聚类算法和基于划分的聚类算法. 层次聚类算法当文档集数据量大时计算开销大, 考虑检索结果聚类速度的要求, 层次聚类一般不被使用. 基于划分算法将数据集合分割为平坦结构的分区, K-means算法通过多次迭代, 达到逐步求精的目的,

收稿日期: 2008-03-18; 修回日期: 2008-12-26
基金项目: 国家自然科学基金重点项目(No. 60435020); 国家自然科学基金面上项目(No. 60503072)

该算法具有线性的时间复杂度. K-means 是通过最优化一个准则函数来生成聚类划分, 准则函数是一个定义在整个数据集合上的函数, 利用组合搜索的方法来找最优解显然在计算复杂度上不可行, 经常取多个不同的初始状态. 初始划分好坏对最终聚类的质量有较大影响. 选择初始聚点一般有经验选择、随机选择、最小最大原则等方法, 对经验知识的依赖也较小. 综上, K-means 聚类方法在搜索结果文档聚类的缺陷[5]:

(1) 初始值难以确定: 由于搜索结果成千上万, 热点结果往往排在最前面, 这就给 K-means 聚类算法随机选取初始点带来更大的困难;

(2) 聚类主题不易刻画: 现有基于词频统计的方法, 不包含语义信息, 无法清楚描述主题[6];

针对如上问题, 本文提出了基于优化初始选择的 K 均值搜索结果聚类方法(Osrc, Ontology Based Search Result Clustering Algorithm) , 通过对检索结果在语义层面的理解, 对 k-means 方法进行了改进, 提出了一种利用 ontology 抽取选择聚点和 k 值的方法来克服选择初始聚点的盲目性. 并利用检索使用的 query 信息及结果排序信息对起始点进行优化, 为聚类描述提供依据. 该方法通过优化初始选择来减少迭代次数, 提高了系统性能和效率.

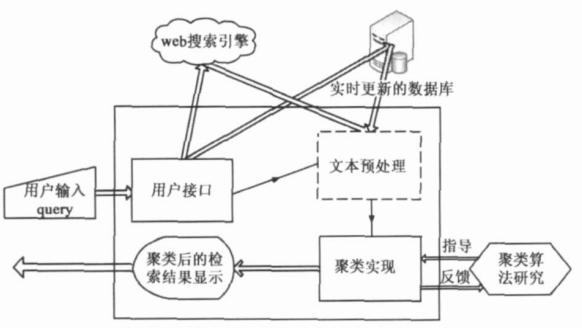


图1　搜索结果聚类系统总体设计方案示意图

如图 1 所示, 对于输入的 query 通过用户接口从 web 搜索引擎中下载返回结果, 经过分词和停用词去除处理后交给聚类程序; 聚类模块把聚类的结果和标签信息一起返回给用户.

3　系统设计

3.1　基于标签的聚类方法

检索结果的聚类方法可以分为两类: 基于文档的方法和基于标签的方法[7], 如图 2 所示.

(1) 基于文档的聚类方式使用传统的文本聚类方法, 把搜索引擎返回的文档聚成多个类别, 然后从各类别中抽取出合适的标签来标注各个类别[8,9].

(2) 基于标签的方法首先从文档集中抽取合适的有代表性的词、短语、片段作为类别标签, 然后对抽取的类别标签进行合理的评价与筛选, 以抽取的标签为基础作进一步的文档聚类工作[10].

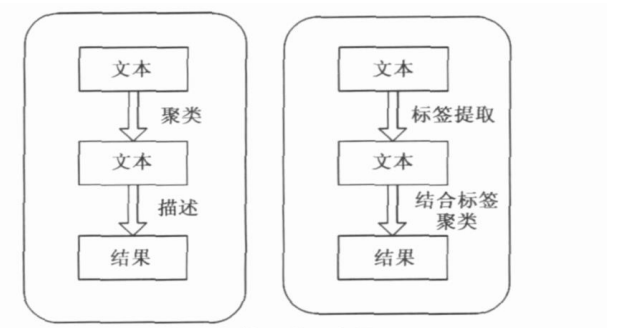


图2　聚类方法示意图

其中基于标签的方法由于对网页信息中不规律的文本良好的指导性被广泛应用在检索结果聚类系统中, 本文提出一种基于中文语言理解的 ontology 抽取的标签提取方法.

3.2　算法概述

Osrc 体系结构如图 3 所示, 可以分为 4 个主要模块: 聚类文档的获取、ontology 抽取、聚类初始质心的确定、聚类实现以及类别描述(即标签提取) . 聚类文档聚类的获取包括检索结果下载和文档预处理; 检索结果系统中的 query 信息和网页结果排序在初始类别确定和初始质心确定起到重要作用.

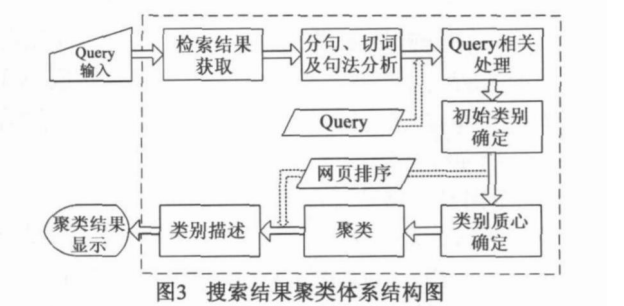


图3　搜索结果聚类体系结构图

3.3　聚类文档的获取

检索结果聚类方法一般在搜索引擎返回的文档摘要(snippet) 的基础上进行. 本文系统利用“百度”和“google”返回结果页面利用规则的方法抽取检索结果, 每个结果由标题(title) 、摘要(snippet) 、原文链接(URL) 以及检索结果排序(order) 构成的四元组[title, snippet, URL, order] . 其中, title 和 snippet 组成供聚类使用的文档.

3.3.1　ontology 抽取

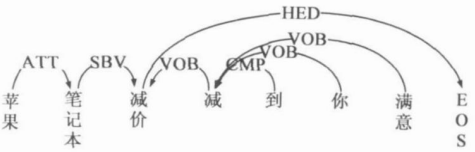
检索结果聚类与文本聚类的不同在于额外提供两个可用的资源: 用户查询的 query 和检索结果排序信息. 据统计, 大多数查询趋向于短查询[10]. 其中, 通过对 query 的抽取寻找 query 的修饰和限制成分以及概念的外延, 确定潜在存在的类别信息对后面的聚类算法加以指导. 本文采用内容挖掘方法, 通过分析文档内容来

抽取web知识,而并非一定是频繁词项[11].

对query的抽取可分为概念限定、概念理解以及概念延伸三个方面,分别利用以下三种方法进行抽取:

●利用依存句法分析进行限制修饰关系抽取:

对query的ontology抽取最基础的方法是在snippet中寻找修饰、限制和支配成分.本文使用的是哈尔滨工业大学信息检索实验室依存句法系统[12].依存句法将句子由一个线性序列转化为一棵结构化的依存分析树,通过依存弧反映句子中词汇的依存关系,例如:"苹果笔记本减价减到你满意",依存分析结果如下:



首先对检索结果中包含query的句子进行分析,表1是利用句法分析抽取的几个实例:

表1 句法抽取实例

关系类型	实例	结果
定中关系	[16]苹果_[15]笔记本 ATT	笔记本
DE关系	[11]腐烂_[8]的(ATT) [8]的_[7]苹果(DE)	腐烂
主谓关系	[10]苹果_[9]腐烂(SBV)	腐烂

利用句法分析帮助理解query实际上是一个模拟人理解的过程.从结果来看,例如对"苹果"抽取结果"笔记本"、"电脑"、"网站"都很准确的表达了"苹果"在各个方面的属性.

●利用同义词词林进行概念的分解:

使用句法分类对query进行修饰和限制成分的抽取的方法对一些特定词语效果并不理想.例如query为"水果",得到的结果"腐烂水果"、"新鲜水果"、"过季水果"等抽取结果缺乏对语言的理解性,并不是合理的类别体系.所以,在这里引入《同义词辞林》[13]对query进行概念抽取.如图4所示,依据《词林》的知识对于"水果"的ontology抽取结果为"瓜果"、"石榴"、"柑子"、"橙子"等.利用《词林》进行概念描述的抽取是具有启发性的,把语言理解和人的知识融入检索模型得到的结果更符合人对类别理解要求.

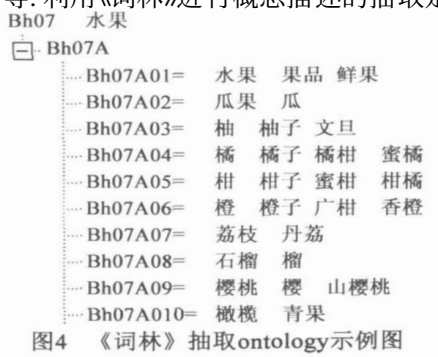
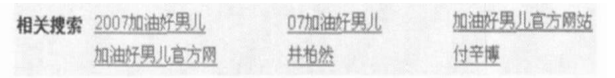


图4 《词林》抽取ontology示例图

需要说明的是利用《词林》进行ontology抽取只局限于该query出现在《词林》的前四层,不包括出现在第五层词汇.这是根据《词林》特点,位于第五层的词汇都是单义词且概念不可分解.

●利用相关搜索进行概念的延伸:

检索结果是根据网页随时变化的,对于新概念或者新名词,前两种抽取方法都受到局限.这种情况下,本文使用网络引擎提供的"相关搜索"进行ontology的抽取.例如:query为"加油好男儿",分词系统不会认为这是一个词,而字典中也不会有相关概念,使用相关搜索可以更好的解决这个问题.得到结果如下:



从上面结果中我们不仅能抽取"07年"、"官方网站"这样准确的关键词,还可以抽取出"井柏然"之类的代表性人物,这样结果是传统自然语言理解根本无法实现的.

相关搜索来自于以往搜索引擎的用户,集成了人的背景知识,实际上是一种非常有价值的查询词相关的背景知识库,因此利用相关搜索对于检索结果聚类将有很强的指导意义.

3.3.2 聚类初始质心的确定

聚类初始质心的确定可分为两个步骤:首先通过以上三种ontology抽取方法得到的结果组成一个质心候选集合,然后通过质心集合在snippet集合中寻找代表性文本.质心集合是有可能成为质心的词组成,ontology抽取的方法尽量找出所有可能,在此基础上还需要通过计算集合所有词在snippet中出现的次数去除候选集合中很少甚至没有出现过的词,最后得到质心序列.抽取质心候选集合过程如下:

(1) query的处理;

(2) 从[title, snippet]从提取含有query的句子;

(3) 对query按2.3.1介绍的三种方法进行ontology抽取,得出质心候选集合;

(4) 计算DF,选取前 N 个结果作为聚类初始质心序列集合 $C=\{C_1, C_2, \cdots, C_m\}$.质心是由一个词或者几个词组成的类别描述的核心,并不能指导聚类.由此,得到聚类初始质心序列集合后,使用词频(DF)以及检索结果排序(order)信息,利用公式1计算得到每个质心对每篇文本贡献度 tf,从检索结果中找出该质心对应的代表性文本.

$$tf(m, d_i) = \frac{f(m, d_i)}{\sum_{j \neq i}^{n} f(m, d_j)} \quad (1)$$

其中 $f(m, d_i)$ 表示修饰词 m 在文本 d_i 中的出现频率, D 为文本集合.

$$C(m) = \{d_i \mid f(m, d_i) > \delta, o(d_i) < \mu\} \quad (2)$$

其中 $o(d_i)$ 表示文本 d_i 在检索结果中的排序, δ 和 μ 为经验阈值, 和待聚类文档颗粒度以及候选质心个数相关.

3.3.3 改进的 K-means 聚类算法

Orsc 不仅在精度上对对 K-means 方法在两方面进行改进, 还充分利用基于标签的聚类算法的优势, 加入了聚类顺序信息. 优化的质心对聚类有指导作用, 优化的质心在聚类过程中偏移较小. 待聚文本与 $C(M)$ 中所有质心进行计算, 按降序顺序得到聚类顺序. 加入聚类顺序实际是减少质心偏移距离, 从而使聚类算法迭代次数减少, 提高算法效率. K-means 时间复杂度是 $O(mkn)$, 其中 m 为文本总数, k 为聚类数目, n 为迭代次数. 改进的 K-means 算法加入聚类顺序信息, 使每次迭代后重新计算质心的偏移降低, 使算法更快达到最优值, 减少了迭代次数, 提高算法效率.

输入: 初始质心集合 $C = \{C_1, C_2, \cdots, C_m\}$
待聚类文本集合 $D^* = \{d_i \mid d_i \in D$ 且 $d_i \notin C\}$
输出: 结果类别集 $R = \{R_1, R_2, \cdots, R_m\}$,
其中 $R_i \supseteq C_i$, $i = 1, 2, \cdots, m$
步骤:
1 S1: $R_i = C_i$ for $i = 1, 2, \cdots, m$
2 S2: for D^* 里的所有文本 d_i do
3 $r(d_i) = \arg\min_{j=1,\cdots,m} f(d_i, R_j)$
4 end
5 S3: $sort(r(D^*))$
6 S4: for $r(D^*)$ 里所有文本 d_i do
7 找到类别 k, 使
$f(d_i, R_k) = \arg\min_{j=1,\cdots,m} f(d_i, R_j)$
8 将 d_i 加入类别 R_k
9 调整 R_k 权值向量
10 end

3.3.4 类别标签提取

标签是聚类结果描述的最基本方式. 标签可以起到对聚类结果进行校验和核对作用. 标签提取基本标准是词频特征, DF 是指包含标签 label 的文档数; 同时还考虑标签词所处位置, 经常出现在标题(title)中或者经常和 query 同处一个句子的词也被认为重要性高于其他词; 最后, 本文使用依存句法分析树型结构, 计算标签词和 query 的语义距离.

$$W_i = DF + \alpha DF_T + \beta DF_S \quad (3)$$

其中 DF_T 表示 W_i 在所有标题中出现的频率, DF_S 表示 W_i 在这样的句子中出现的频数: 该句子既包含 W_i 和 query 且它们在该句的依存句法分析中距离(弧长)不超过 3. 通过实验证明语义信息的标取更准确的表达该类别的意义.

4 实验和讨论

4.1 实验设置

在检索结果聚类评价中, 还没有一个标准的测试集, 以往对聚类方法的评价均采用了自己构建的评价集. 我们得到中科院张刚博士共享的人工标注测试集, 该测试集共包含从"百度"索引擎按查询类型分类的 30 个 query 的前 100 个返回结果, 共计 3000 篇文档,

4.2 聚类结果评价

4.2.1 评价方法

聚类结果的质量评价方法可分为外部和内部两种, 有数据集先验知识的是外部评价方法, 没有先验知识的是内部评价方法. 本文使用一种常用的外部评价方法 F-measure[2]. F-measure 的定义则参照信息抽取的评测方法, 将每个聚类结果看作是查询的结果, 对于最终的某一个聚类类别 j 和原来的预定类别 i:

$$Recall(i, j) = N(i, j)/n_i \quad (4)$$

$$Precision(i, j) = N(i, j)/n_j \quad (5)$$

这里 $N(i, j)$ 是聚类 j 中包含类别 i 中的文档的个数, n_i 是最后聚类的个数, n_i 是预定义类别的个数. 则聚类 j 和类别 i 之间的 F 值计算如下:

$$F(i, j) = \frac{2 \times Recall(i, j) \times Precision(i, j)}{Recall(i, j) + Precision(i, j)} \quad (6)$$

最终聚类结果的评价函数为:

$$F = \sum_i \frac{n_i}{n} \max\{F(i, j)\} \quad (7)$$

这里 n 是所有测试文档的个数.

4.2.2 实验结果

(1) 三种特征 ontology 抽取方法比较实验

为验证抽取 ontology 方法的效果, 本文对三种方法组合使用得到表 2 所示结果. 其中表示 $Orsc_1$ 只使用句法分析进行 ontology 抽取; $Orsc_2$ 使用句法分析和字典方法进行 ontology 抽取; $Orsc_3$ 表示三种方法同时使用.

表 2 查询类型对照表

	准确率	召回率	F 值
$Orsc_1$	0.562874	0.670714	0.61208
$Orsc_2$	0.581886	0.677143	0.625911
$Orsc_3$	0.595525	0.716429	0.650406

从上表结果看出三种 ontology 抽取方法组合使用的效果有明显提高.

(2) 与其他聚类系统比较实验

图 2 是三种方式与广泛认同的 STC 算法和 Snaket 聚类算法和 GBCA 算法进行比较, Orsc 算法体现了更

好的聚类性能. Snaket 算法主要利用了标签的特点而没有重视文档在类别中分布因素,性能上比其他方法稍差.

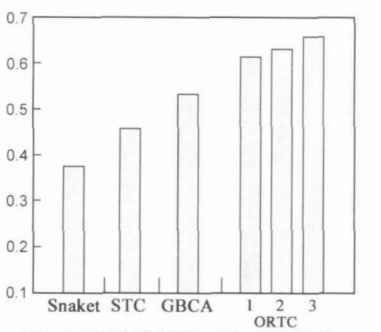


图5 不同聚类算法F-Measure比较

4.3 标签提取评价

在标签评价方面使用类似检索评价的 $P@N$[16] 评价类别标签抽取评价方法. 把抽取出的类别标签按照重要性进行排序,通过评价前 N 个标签中,有多少个用户期望的标签作为类别标签的评价标准, $P@N$ 被定义为

$$P@N = \frac{M@N}{N}$$

$M@N$ 在排序前 N 个标签中和手工标注的数目,考虑用户浏览兴趣在 10 个标签内,采用 $P@3$, $P@5$, $P@7$ 和 $P@10$ 来进行评价. 从聚类结果的标签评价可以看出, Orsc 算法效果在 $P@5$ 和 $P@7$ 效果最好. 由于算法生成类别数为 8 类,所以在 $P@3$ 和 $P@10$ 比其他算法略低.

表3 标签抽取结果准确率

	$P@3$	$P@5$	$P@7$	$P@10$
STC	0.29	0.33	0.30	0.31
Snaket	0.43	0.47	0.43	0.40
GBCA	0.46	0.47	0.48	—
Orsc	0.38	0.48	0.48	0.32

5 总结

本文针对检索结果聚类问题,给出一种改进的 k-means 聚类算法. 算法采用了基于标签的聚类思想,利用自然语言处理术中 ontology 抽取技术,将特征融合到统一的模型中,进行类别标签的抽取. 并利用标签和检索结果返回结果排序信息对聚类算法进行指导和改进. 在实验中验证了该算法的有效性,通过与 STC、Sarrket, GBCA 检索结果聚类算法的比较,验证了在 F 值、$P@N$ 等重要指标上得到提高,而且效率也达到实用效果. 未来希望从二方面展开工作: 第一,聚类结果呈现对于用户产生最直观的影响,标签提取的结果是至关重要的,未来工作加入更多语言处理技术来完善标签提取; 第二,文档重叠性是指一篇文档可能含有多个主题,在本文中没有考虑这种情况.

作者简介:

陈毅恒 男,1979 年生于黑龙江哈尔滨. 哈尔滨工业大学计算机科学与技术学院博士生. 研究方向为自然语言处理.
E-mail: cyh@ir.hit.edu.cn

秦 兵 女,1968 年生于黑龙江哈尔滨. 哈尔滨工业大学计算机科学与技术学院教授,硕士生导师. 研究方向为自然语言处理、信息检索.

参考文献:

[1] Campos R, Dias G, Nunes C. WISE: Hierarchical soft clustering of web page search results based on web content mining techniques[A]. Proceeding of the 2006 WIC/ACM International Conference on Web Intelligence[C]. Hong Kong, 2006.

[2] Chuang SL, Chien LF. A practical web-based approach to generating Topic hierarchy for text segments[A]. Proceeding of CIKM'04[C]. Washington D.C., USA, 2004. 127-136.

[3] Osinski S, Weiss D. Conceptual clustering using lingo algorithm: Evaluation on open directory project data[A]. IIPWM04[C]. Sapporo, Japan, 2004. 81-88.

[4] Giannotti F, Nanni M, Pedreschi D. Webcat: Automatic categorization of web search results[A]. SEBD03[C]. Cetraro, Italy. 71-82.

[5] Salton G. The SMART Retrieval Systems[M]. Prentice Hall, Englewood Cliffs, N.J, 1971.

[6] Geraci F, Pellegrini M, Maggini M, Sebastiani F. Cluster generation and cluster labeling for web snippets[A]. SPIRE 2006, LNCS[C]. Glasgow, UK, 2006. 25-36.

[7] Hiroyuki Toda, Ryoji Kataoka. A search result clustering method using informatively named entities[A]. Proceedings of the ACM Workshop on Web Information[C]. Louisiana, USA, 2005. 81-86.

[8] Hearst M A, Pedersen J O. Reexamining the cluster hypothesis: Scatter/gather on retrieval results[A]. Proceedings of the ACM Special Interest Group on Information Retrieval Conference[C]. 1996. 76-84.

[9] F Giannotti, M Nanni, D Pedreschi. Webcat: Automatic categorization of web search results[A]. Proceedings of the Eleventh Italian Symposium on Advanced Database Systems[C]. Italia, 2003. 507-518.

(下转第 156 页)

[6] Chinese folk song[Z]. http://www.china.org.cn/chinese/minge/435499.htm

[7] Chih-chung Chang, Chih-jen Lin. LIBSVM: a library for support vector machines[R]. 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

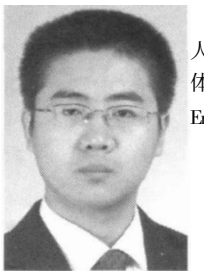
[8] Feng Chu, Lipo Wang. Active mining discriminative gene sets [A]. The 8th International Conference[C]. Zakopane, Poland. Springer Berlin/Heidelberg. 2006. 880-889.

[9] Changsheng Xu, Namunu C. Maddage, Xi Shao. Automatic music classification and summarization[J]. IEEE Transactions on Speech and Audio Processing, 2005, 13(3): 441-450.

[10] Tao Li, George Tzanetakis. Factors in automatic musical genre classification of audio signals[A]. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics[C]. New Paltz, NY, 2003. 143-146.

作者简介:

刘 怡 女, 1949 年出生于北京, 中国人民大学信息学院副教授. 研究方向: 数据库技术、多媒体数据挖掘、多媒体信息智能检索.
Email: liuyilee@ruc.edu.cn



蔚 磊 男, 1983 年出生于山东东平, 中国人民大学信息学院硕士研究生, 研究方向: 多媒体数据挖掘、音乐信息智能检索.
Email: sunraising@163.com

(上接第 170 页)

[10] Franzen K, Karlgren J. Verbosity and interface design[A]. Technical Report T2000: 04[C]. Swedish Institute of Computer Science, 2000. 61-69.

[11] Kosala R, Blockeel H. Web mining research: A survey[J]. ACM SIGKDD Exploration, 2000, 2(1): 1-15.

[12] Ting Liu, Jinshan Ma, Huijia Zhu, Sheng Li. Dependency parsing based on dynamic local optimization[A]. Proceedings of Tenth Conference on Computational Natural Language Learning[C]. CoNLL shared task, New York, 2006. 111-115.

[13] 梅家驹, 竺一鸣, 高蕴琦, 殷鸿翔. 同义词词林[M]. 上海: 上海辞书出版社, 1996.
Mei J, Zhu Y, Gao Y, Yin H. Tong Yi Ci Ci Lin[M]. Shanghai: Shanghai Lexicographical Publishing House, 1996. (in Chinese)