一种基于语义相似度的群智能文本聚类的新方法 *

陶 红, 周永梅, 高 尚
(江苏科技大学 计算机科学与工程学院, 江苏 镇江 212003)

摘 要: 针对基于 VSM(vector space model)的文本聚类算法忽略了词之间的语义信息和各维度之间的关系,导致文本的相似度计算不够精确,提出了一种基于语义相似度的群智能文本聚类的新方法。该方法融合了模拟退火算法的全局搜索和蚁群算法的正反馈能力。其思路是,首先从语义上分析文本,利用 K-均值算法进行文本聚类,再根据 K-均值算法的结果,使用蚁群和模拟退火算法进行调整聚类。测试结果表明这种算法能够提高聚类精度和召回率,也验证了混合算法的正确性。

关键词: 文本聚类; 语义相似度; K-均值算法; 蚁群算法; 模拟退火算法
中图分类号: TP301.6 **文献标志码**: A **文章编号**: 1001-3695(2012)02-0482-03
doi:10.3969/j.issn.1001-3695.2012.02.021

New method of hybrid intelligent text clustering based on semantic similarity

TAO Hong, ZHOU Yong-mei, GAO Shang
(*School of Computer Science & Engineering, Jiangsu University of Science & Technology, Zhenjiang Jiangsu* 212003, *China*)

Abstract: The problem with the text clustering algorithm based on vector space model (VSM) is that semantic information between words and the link between the various dimensions are overlooked, resulting in inaccuracy in the text similarity calculation, this paper proposed a hybrid intelligent algorithm based on computing the text semantic similarity. This algorithm combined the good global search capability of simulated annealing algorithm and the good positive feedback ability of ant colony algorithm. It extended the algorithm to analyze the text according to its semantic, then used K-means clustering to seed the initial solution and the ant colony algorithm and simulated annealing algorithm to adjust the initial cluster. Through the result, this algorithm can improve the clustering precision and recall rate and the efficiency of the hybrid algorithm is verified.
Key words: text clustering; semantic similarity; K-means algorithm; ant colony algorithm; simulated annealing algorithm

文本聚类是一种典型的无指导的机器学习问题。它将一个文本集分成若干类,每个类中的成员之间有较大的相似性,而类间的文本具有较小的相似性。目前,人们已提出很多文本聚类算法[1],比较典型的有基于划分的聚类方法 K-means 和 K-medoies、基于层次的聚类方法 AGENS(aglomerative nesting)和 DIANA(divisive analysis)、基于密度的聚类方法 DBSCAN(density-based spatial clustering of applications with noise)、基于蚁群的聚类方法等。大部分是基于 VSM 来进行聚类。该模型假设词语间是独立的,并没有从语义上去分析文本内容,因而不能准确计算文本间的相似度,影响了聚类的精度。本文利用“知网”[2]中的丰富语义信息来分析文本,根据词语频率(term frequency, TF)和词语倒排文档频率(inverse document frequency, IDF)选择文本的主要关键词,将文本表示为一组关键词集合,再计算文本间的相似度,进行文本聚类。

蚁群算法(ant colony optimization, ACO)是一种新型的模拟蚂蚁群体智能行为的仿生优化算法,由意大利学者 Dorigo 等人[3]于 1991 年率先提出来,具有良好的正反馈特性,同时也存在易陷入局部收敛的缺陷[4]。蚁群算法常用来解决离散域中的许多优化问题,典型的应用有旅行商、调度、车辆绕径等问题[5]。模拟退火算法(simulated annealing, SA)最早由 Metropolis 等人于 1953 年提出, 1983 年 Kirkpatrick 等人将其应用于组合优化问题中。它具有渐进收敛性,在理论上已经被证明是一种以概率 1 收敛于全局最优解的全局优化算法,其初始解的设定对算法的收敛速度影响很大。

1 文本间语义相似度计算的基本思想

1.1 文本主要关键词的提取

文本聚类的一般过程是:对文本进行预处理、计算相似度和聚类算法进行聚类,预处理部分分为对文本分词、去停用词、特征提取等几个步骤。文本 i 的特征词 j 的权重 W_{ij} 按照下式:

$$W_{ij} = T_{ij} \times \log\left(\frac{N}{n_i}\right) \times a_{ti} \times b_{ti} \tag{1}$$

$$a_{ti} = \begin{cases} 1.3 & \text{词是名词} \\ 1.0 & \text{词是动词} \\ 0.9 & \text{词是形容词} \\ 0.8 & \text{词是副词} \end{cases}$$

$$b_{ti} = \begin{cases} 1.3 & \text{词在标题} \\ 1.0 & \text{词是首段} \\ 0.9 & \text{词是中间} \\ 0.8 & \text{词是末端} \end{cases}$$

收稿日期: 2011-07-20; **修回日期**: 2011-08-25 **基金项目**: 人工智能四川省重点实验室开放基金资助项目(2009RY001);“青蓝工程”资助项目

作者简介: 陶红(1988-),女,江苏靖江人,硕士研究生,主要研究方向为智能计算与技术(taohongbaozi@sina.com);周永梅(1986-),女,江苏南通人,硕士研究生,主要研究方向为智能信息处理;高尚(1972-),男,安徽天长人,副教授,博士,主要研究方向为系统工程理论与优化.

来计算的。其中:T_{ij}表示特征词 j 在文本 i 中的词语频率;N 为文本的数量;n_i 表示出现过特征词 j 的文本数量;a_{ti}表示词性的权重系数[6];b_{ti}表示位置权重系数[7],按照权重 W_{ij}的大小来进行提取,选取权重 W_{ij}最大的 10 个词作为该文本的主要关键词。

1.2　基于"知网"的文本间语义相似度的计算

基于 VSM 的文本相似度一般是利用关键词权重匹配的方法,使用余弦系数计算文本相似度。这种方法忽略了词语间的联系,忽略了语义信息,导致聚类结果不理想。"知网"是一个以汉语和英语词语代表的概念为描述对象,以揭示概念之间以及概念所具有的属性之间的关系为基本内容的常识知识库[2]。"知网"通过义原的组合标注各种单纯的或复杂的概念,以及各个概念之间、概念的属性与属性之间的关系。在 VSM 中引入"知网",将关键词转换为义原,可以在一定程度上解决同义词替换的问题,使相同主题的文本能更好地聚集在一起。

1.2.1　词语相似度计算方法

在"知网"中,每个词的语义描述由多个义原组成。把除第一义原外的义原称为其他义原。计算两个词 W_1(W_1 有 n 个义项 $S_{11},S_{12},\cdots,S_{1n}$)、$W_2$($W_2$ 有 m 个义项 $S_{11},S_{22},\cdots,S_{2m}$)的相似度 $\mathrm{sim}(W_1,W_2)$具体分为四个方面计算如下:

a)若两个词都只有一个义原,则按式(2)[6]

$$\mathrm{sim}(W_1,W_2)=\frac{\alpha}{\mathrm{dis}(W_1,W_2)+\alpha} \tag{2}$$

计算 W_1 和 W_2 第一义原的相似度 sim_1,否则按照式(3)[6]:

$$\mathrm{sim}(W_1,W_2)=\max_{i=1,2,\cdots,n;\ j=1,2,\cdots,m}\mathrm{sim}(S_{1i},S_{2j}) \tag{3}$$

判断义原描述符,并计算相似度 sim_1。其中:α 是一个可调节的参数,代表当相似度为 0.5 时的词语距离值;$\mathrm{dis}(W_1,W_2)$是词语 W_1 和 W_2 的距离值。

b)若 W_1 和 W_2 有相同义原,且该相同义原不是弱义原,则 $\mathrm{sim}(W_1,W_2)=1$,两个词同义。

c)若 W_1 只有一个基本义原,W_2 有两个以上义原,则继续判断 W_1 基本义原是否与 W_2 的其他义原相等且不是弱义原。若是,则 $\mathrm{sim}(W_1,W_2)=0.8$,说明这两个词是相关的;否则,$\mathrm{sim}(W_1,W_2)=\mathrm{sim}_1$。

d)若 W_1 和 W_2 都有两个以上义原,则两两比较两个词其他的义原,若有两个义原相等且不足弱义原,则 $\mathrm{sim}(W_1,W_2)=0.8$,结束计算;若没有义原对相等,或者有义原对相等且是弱义原,则按照式(4)[6]

$$\mathrm{sim}(W_1,W_2)=\sum_{i=1}^{4}\beta_i\,\mathrm{sim}(S_1,S_2) \tag{4}$$

计算两个词(S_1,S_2)的相似度,赋予较低权重,即对应的 β_i 值比较小。$\mathrm{sim}_1(S_1,S_2)$、$\mathrm{sim}_2(S_1,S_2)$、$\mathrm{sim}_3(S_1,S_2)$和 $\mathrm{sim}_4(S_1,S_2)$分别为两个词的第一独立义原描述式、其他独立义原描述式、关系义原描述式和符号义原描述式的相似度。

1.2.2　文本相似度计算方法

文本相似度计算的关键词以词语相似度为基础,计算两篇文本 D_1(D_1 有 x 个词语,$W_{11},W_{12},\cdots,W_{1x}$)、$D_2$($D_2$ 有 y 个词语,$W_{21},W_{22},\cdots,W_{2y}$)的相似度 $\mathrm{sim}(D_1,D_2)$:

$$\mathrm{Sim}(D_1,D_2)=\frac{1}{x\times y}\sum_{i=1,2,\cdots,x;j=1,2,\cdots,y}\mathrm{sim}(W_{1i},W_{2j}) \tag{5}$$

若 D_1(D_1 有 x 个词语,$W_{11},W_{12},\cdots,W_{1x}$)与 D_2(D_2 有 y 个词语,$W_{21},W_{22},\cdots,W_{2y}$)的词语中,至少有三个词语的第一义原是相同的,则

$$\mathrm{sim}(D_1,D_2)=\frac{1}{x\times y}\sum_{i=1,2,\cdots,x;j=1,2,\cdots,y}\mathrm{sim}(W_{1i},W_{2j})+0.2 \tag{6}$$

这样处理之后使类中的文本相似度和类间的文本相似度能够明显地区分,同一类的文本相似度更高,而类间的文本相似度相对较低。之后的聚类算法是基于文本相似度的,这样也能够提高聚类的精度。

2　基于文本语义相似度计算的群智能文本聚类方法

2.1　K-means、蚁群和模拟退火算法融合的基本思想

由于模拟退火算法的收敛速度很慢,本文提出的混合算法利用了蚁群算法的正反馈能力,可以提高模拟退火算法的收敛速度,而蚁群算法容易陷入局部收敛,利用模拟退火算法的全局搜索能力,可以避免蚁群算法出现局部收敛[8]。利用它们各自的优势,提出一种融合蚁群与模拟退火算法的算法。基本思想是,首先利用 K-means 算法对文本进行快速聚类,然后利用蚁群算法进行迭代,对每次迭代的最优结果进行模拟退火。由于模拟退火算法允许解以一定的概率变坏,这样既可以避免蚁群算法陷入局部最优,也可以提高模拟退火的速度。

2.2　K-means、蚁群和模拟退火算法的融合

2.2.1　问题描述

从原理上,基于蚁群的聚类方法可以分为两种:一种是基于蚁堆形成原理来实现聚类的方法,另一种是基于蚁群觅食原理来实现聚类的方法。本文采用的是第二种蚁群聚类方法。蚁群觅食原理的思想是,将文本视为具有不同属性的蚂蚁,而将聚类中心看做是蚂蚁要寻找的食物源,那么聚类的过程就可以看成是蚂蚁寻找食物源的过程。

已知文本集$\{X\}$有 N 个文本和 K 个分类$\{S_j\}$($j=1,2,\cdots,K$),以类内文本间平均相似度最大为聚类准则,其数学模型为

$$\max\sum_{j=1}^{K}\mathrm{AVE\_sim}(S_j) \tag{7}$$

其中:K 为聚类数目;$\mathrm{AVE_sim}(S_j)=\frac{1}{N_j\times N_j}\sum_{D_1,D_2\in S_j}\mathrm{sim}(D_1,D_2)$为 S_j 中文本的平均相似度,N_j 为 S_j 中文本的数量。

2.2.2　K-means 算法

K-means 算法的步骤如下:

a)随机产生 K 个初始的聚类中心:$Z_1,Z_2,\cdots,Z_k$;

b)各文本按照与聚类中心的相似度最大的原则选择聚类中心;

c)计算新的聚类中心 $Z_j(j=1,2,\cdots,K)$,聚类中心根据文本 D_1($D_1\in\{S_j\}$)与类内各文本的平均相似度 $\mathrm{AVE_Cluster_sim}(D_1,S_j)$与类内文本间的平均相似度 $\mathrm{AVE_sim}(S_j)$之差最小的原则来更新,其中:

$$\mathrm{AVE\_Cluster\_sim}(D_1,S_j)=\frac{1}{N_j-1}\sum_{D_1,D_2\in S_j,D_1\neq D_2}\mathrm{sim}(D_1,D_2) \tag{8}$$

d)若聚类中心发生变化,则转 b);否则算法迭代结束。

2.2.3　信息素初始化

初始的文本由 K-means 算法进行快速分类,蚁群算法根据分类的启发式信息,初始化信息素浓度。若文本 D_i 分配给聚类中心 $S_j(j=1,2,\cdots,K)$,则文本 X_i 到 S_j 的路径上的信息素浓度 τ_{ij}为

$$\tau_{ij}=\tau_{ij}(0)+Q\times\mathrm{sim}(D_i,S_j) \tag{9}$$

其中:$\tau_{ij}(0)(i,j=1,2,\cdots,K)$为初始的信息素浓度值,均为相

同的数值;Q 为一正常数;$\mathrm{sim}(D_i,S_j)$ 为文本 D_i 与聚类中心 S_j 的语义相似度 $\tau(0)$。

2.2.4　移动

第 i 只蚂蚁选择聚类中心 S_j 的概率为

$$p_{ij}=\frac{\tau_{ij}}{\sum_{j=1}^{K}\tau_{ij}} \tag{10}$$

信息素更新方程为

$$\tau_{ij}^{\mathrm{new}}=\rho\times\tau_{ij}^{\mathrm{old}}+Q\times\mathrm{sim}(D_i,S_j) \tag{11}$$

其中:ρ 表示信息素残留系数,一般取 0.5 ~ 0.9 左右[9]。

2.2.5　模拟退火局部搜索

为快速获取最优解,模拟退火邻域搜索从一次蚂蚁搜索获得的最优解 W_0 开始随机扰乱产生另一个解,定义为 W_1。然后,根据前述的目标函数式(7),计算最优解和局部搜索解的目标函数值,分别为 $f(W_0)$ 和 $f(W_1)$,并以概率式(12)的方式接受新解为当前最优解,其中 T 为当前温度。

$$p=e^{-(f(w_1)-f(w_0))/T} \tag{12}$$

2.2.6　融合的蚁群和模拟退火算法的数据流图(图 1)

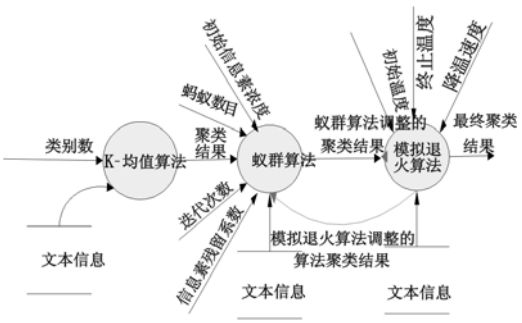

图1　融合的蚁群和模拟退火算法的数据流图

2.3　融合的蚁群和模拟退火算法的求解步骤

a)参数的初始化;

b)利用 K-means 算法进行快速分类,获得当前全局最优解,并按式(9)初始化蚁群的信息素浓度;

c)for 每次迭代 do

(a)for 每只蚂蚁 do

(b)按照式(3)计算选择概率 p_{ij},并以该概率选择聚类中心;

(c)按照式(4)局部更新信息素浓度;

end for

(d)按照式(1)计算目标函数值,并计算比较得出本次全局搜索中的最优解;

(e)设定初始温度 T,终止温度 T_0,使用当前最优解作为初始解 W_0;

(f)对当前最优解 W_0 进行随机扰动,产生变异解 W_1;

(g)以式(5)的概率接受新解 W_1 为当前最优解,存为当前最优解;

(h)检查退火终止条件,如果满足则转向步骤 d),否则转向(i);

(i)按照降温公式计算下次迭代温度,转向(f);

d)将全局最优解和当前最优解作比较,更新全局最优解;

end for

e)结束搜索过程,输出最终结果。

3　结论

本文实验语料主要从中国科学院的中文自然语言处理开放平台[10] CNLP 网站上下载 50 篇文档作为测试数据,并根据语料主题手工分为 10 类,分别为军事(10)、体育(10)、政治(10)、教育(10)、经济(10)。本文采用查准率 P 和查全率 R 来评价算法的性能。一个聚类 j 及与此相关的分类 i 的聚类精度 P 和聚类召回率 R 定义为

$$P=\mathrm{precision}(i,j)=\frac{N_{ij}}{N_j} \tag{13}$$

$$R=\mathrm{recall}(i,j)=\frac{N_{ij}}{N_i} \tag{14}$$

下面利用平均聚类精度和平均聚类召回率分别比较基于 VSM 的 K-means 算法(算法 1)、基于语义相似度的 K-means 算法(算法 2)和基于语义相似度的群智能文本聚类算法(算法 3)性能。本文算法参数为 $\rho=0.9$,$Q=80$,迭代次数 $NC=1000$,蚂蚁数目 $R=30$,初始温度 $T=1000$,终止温度 $T_0=0.01$,降温速度 $\alpha=0.95$,算法测试 50 次。表 1 为聚类精度比较结果,表 2 为聚类召回率比较结果。

表 1　聚类精度比较结果

算法	军事	体育	政治	教育	经济
算法 1	0.33	0.28	0.28	0.57	0.3
算法 2	0.7	0.7	0.7	0.75	0.6
算法 3	1	1	0.9	0.9	1

表 2　聚类召回率比较结果

算法	军事	体育	政治	教育	经济
算法 1	0.2	0.33	0.28	0.2	0.3
算法 2	0.8	0.8	0.8	0.75	0.8
算法 3	1	1	1	0.9	0.86

通过以上结果比较可以看出,本文采用的基于语义相似度的群智能文本聚类算法无论是在聚类精度还是在聚类召回率哪个主题下,都远远高于基于 VSM 的 K-means 聚类算法,也高于基于语义相似度的 K-means 算法,尤其是军事、体育主题聚类精度和召回率都达到了最高,说明蚁群和模拟退火算法融合调整了不好的聚类结果,提高了聚类结果的精度和召回率。聚类的效果不仅受聚类算法的影响,也受文本预处理部分的影响。在文本的预处理部分,对文本特征进行提取的过程又是预处理部分的重要一步,本文利用传统的 Tf-Idf 方法并结合词语词性以及词语在文本中的位置来计算词语在文本中的权重,选出权重最大的 10 个词语作为文本的关键词,达到了不错的效果,进而利用混合算法提高了文本聚类的精度。

4　结束语

文本聚类是文本挖掘领域一个重要技术,当前应用的文本聚类算法大都是基于 VSM,这种方法忽略了词语间的联系和语义信息,导致聚类精度不高。本文从语义上具体分析文本内容,引入语义计算文本间的相似度,先进行 K-means 聚类,接着采用蚁群和模拟算法融合对 K-means 聚类的结果进行调整,达到提高聚类结果的精度和召回率的目的。本文仅测试了 50 篇文章,对于多文本的聚类,还需要进一步研究。如何调整算法使得算法的时间消耗降低也是下一步研究的方向。

参考文献:

[1]　李凡,林爱武,陈国社. 一种基于 VSM 文本分类系统的设计与实现[J]. 华中科技大学学报:自然科学版,2005,33(3):53-57.

(下转第 532 页)

有对分类性能产生多大的作用。对于 CE 和 CHI 也是类似的，而且对于 CE 方法，它的分类性能在达到一定程度之后，则不再随着特征数的增加而增加。同时笔者发现，当选择的特征数达到某个阈值时，各特征选择方法性能均会达到最佳状态，如果此时继续增加选择的特征数，性能不但不会进一步提高，而且还有可能下降。对于这个使得性能达到最佳状态的阈值的确定，则需要通过大量实验才能得到。

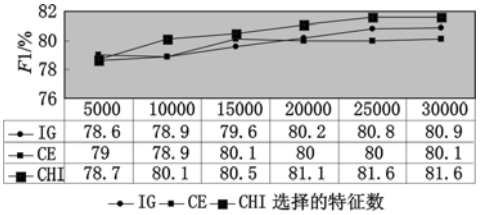

图1　各特征选择方法在不同特征数下的性能比较

表 3 给出了各个特征选择算法对应的实体关系抽取方案的性能比较。

表 3　实体关系抽取方案性能比较

方案	选择的特征		分类性能/%			时间开销/s	
	数量	百分比	*P*	*R*	*F*1	训练	预测
SVM	121 587	100	78.1	85.4	81.6	2 092	935
SVM + IG	30 000	24.7	76.7	85.7	80.9	784	338
SVM + CE	15 000	12.3	75.5	85.4	80.1	565	198
SVM + CHI	25 000	20.1	77.1	86.6	81.6	691	223

比较表 3 的分类性能数据可以发现，无论使用哪一种特征选择方法都没有提高实体关系分类性能，最好的情况也就是不降低它的性能。这是由于在实体关系抽取方案中加入了特征选择算法之后，降低了分类时特征空间维数，而在这个降维过程中，有一些对实体关系抽取有用的信息被丢掉。虽然增加了特征选择的实体关系抽取方案可能会降低实体关系分类性能，但从表 2 的数据可以看出，该类方案依然是有其价值的。这是因为首先这类方案只是略微降低了分类性能，比如 SVM + IG 方案只降低了 0.7%，SVM + CE 方案只降低了 1.5%；其次，该类方案有效地减少了分类时的特征数，提高了效率，比如 SVM + CE 方案以将性能降低 0.7% 为代价将特征数也减少到了 24.7%，而 SVM + CHI 方案则在保持分类性能的基础上将特征数减少到了 24.1%。由此可以看出，该类方案是将分类性能和效率作了一个权衡，在尽量保证分类性能的同时提高分类效率。在实际应用中可以根据需要选择合适的实体关系抽取方案。

对于 IG、CE 和 CHI 三种特征选择方法，从图 1 和表 3 的实验结果可以看出，CHI 是更适合于实体关系抽取的。因为在选择相同特征数时，以 CHI 得到的实体关系抽取性能最好。

3　结束语

由于实体关系抽取问题与文本分类问题的相似性，本文引入了文本分类中的特征选择算法，用于解决基于特征向量的实体关系抽取问题中特征空间维数过高的问题。实验结果表明，本文引入的基于信息增益、期望交叉熵和 x^2 统计的特征选择算法均能有效地降低实体关系抽取中的特征维数，减少抽取的时间开销，且保持了实体关系抽取的 *F*1 值。然而，特征选择过程希望最好在降低特征维数的同时提高抽取性能，这个目标是困难的，也将是笔者下一步的研究方向。另外，考虑到本文只是简单引入了文本分类中的特征选择算法，下一步也可以组合多个特征选择算法，以期更进一步地进行有效特征降维。

参考文献：

[1] 黄鑫. 基于特征向量的中文实体间语义关系抽取研究[D]. 苏州：苏州大学，2009.

[2] TYMOSHENKO K, GIULIANO C. Semantic relation extraction using Cyc[C]//Proc of the 5th International Workshop on Semantic Evaluation. Stroudsburg, PA: Association for Computational Linguistics, 2010: 214-217.

[3] GIULIANO C, LAVELLI A, PIGHIN D, *et al*. Kernel methods for semantic relation extraction[C]//Proc of the 4th International Workshop on Semantic Evaluations. Stroudsburg, PA: Association for Computational Linguistics, 2007: 141-144.

[4] ZHOU Guo-dong, QIAN Long-hua, FAN Jian-xi. Tree kernel-based semantic relation extraction with rich syntactic and semantic information[J]. Information Sciences, 2010, 180(2010): 1313-1325.

[5] 庄成龙，钱龙华，周国栋. 基于树核函数的实体语义关系抽取方法研究[J]. 中文信息学报，2009，23(1)：1003-0077.

[6] 黄高辉，姚天昉，刘全升. 基于 CRF 算法的汉语比较句识别和关系抽取[J]. 计算机应用研究，2010，27(6)：2062-2064.

[7] LLORENS H, SAQUETE E, NAVARRO-COLORADO B. TimeML events recognition and classification: learning CRF models with semantic roles[C]//Proc of the 23rd International Conference on Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2010: 725-733.

[8] 张彪. 文本分类中特征选择算法的分析与研究题名[D]. 合肥：中国科学技术大学，2010.

[9] 苏金树，张博锋，徐昕. 基于机器学习的文本分类技术研究进展[J]. 软件学报，2006，17(9)：1848-1859.

[10] NOVOVIČOVÁ, SOMOL P, HAINDL M, *et al*. Conditional mutual information based feature selection for classification task[C]//Proc of the 12th Iberoamerican Conference on Congress on Pattern Recognition. Berlin: Springer-Verlag, 2007: 417-426.

（上接第 484 页）

[2] 董振东. 知网[EB/OL]. (2010-08-20)[2011-07-06]. http://www.keenage.com/html/c_index.html.

[3] DORIGO M, MANIEZZO V, COLORNI A. Ant system: optimization by a colony of cooperating agents[J]. IEEE Trans on Systems, Man, and Cybernetics-Part B, 1996, 26(1): 1-13.

[4] 段海滨，王道波，朱家强，等. 蚁群算法理论及应用研究的进展[J]. 控制与决策，2004，19(12)：1321-1340.

[5] 吴启迪，汪镭. 智能蚁群算法及其应用[M]. 上海：上海科技出版社，2004：54-58.

[6] 刘群，李素建. 基于《知网》的词汇语义相似度计算[J]. 计算语言学及中文信息处理，2002，7(2)：59-76.

[7] 楼华锋，刘功申. 一种基于段落同现频率的加权方法[J]. 信息安全与通信保密，2009(12)：57-63.

[8] 舒湘沅，杨铭，王延平. 航空项目资源均衡优化问题的蚁群—模拟退火算法[J]. 航空制造技术，2010，18(13)：77-81.

[9] 高尚，汤可宗，杨靖宇. 一种新的基于混合蚁群算法的聚类方法[J]. 微电子学与计算机，2006，23(12)：38-40.

[10] 中国科学院计算技术研究所数字化室. 中文自然语言处理开放平台[DB/OL]. (2011-05-25)[2011-07-06]. http://www.nlp.org.cn/.