计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(1):92−97 [doi: 10.15888/j.cnki.csa.006127]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于词性特征的特征权重计算方法①

胡雯雯, 高俊波, 施志伟, 刘志远

(上海海事大学 信息工程学院, 上海 201306)

摘　要: 短文本因其具有特征稀疏、动态交错等特点, 令传统的权重加权计算方法难以得到有效使用. 本文通过引入翻译决策模型, 将某种词性出现的概率作为特征, 提出一种新的基于词性特征的特征权重计算方法, 并用文本聚类算法进行测试. 测试结果表明: 与 TF-IDF、QPSO 两种权重计算算法相比, 改进的特征权重计算算法取得更好的聚类效果.

关键词: 翻译决策模型; TDQO 算法; 词性; 聚类

引用格式: 胡雯雯,高俊波,施志伟,刘志远.基于词性特征的特征权重计算方法.计算机系统应用,2018,27(1):92–97. http://www.c-s-a.org.cn/1003-3254/6127.html

Feature Weight Calculation Method Based on Part of Speech Characteristics

HU Wen-Wen, GAO Jun-Bo, SHI Zhi-Wei, LIU Zhi-Yuan

(College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China)

Abstract: Because of the sparse and dynamic crisscross characteristics, the short text makes the weight of traditional weighted method difficult to use effectively. This paper presents a new feature weight calculation algorithm based on part of speech. This algorithm is the quantum particle swarm optimization algorithm introduced into translation decision model which can calculate the probability of a feature with certain part of speech. Then it is tested by the text clustering algorithm. The test results show that the improved feature weight calculation algorithm on the clustering accuracy is better than TF-IDF and QPSO algorithm.

Key words: translation decision model; TDQO algorithm; part-of-speech; clustering

1　引言

面对大规模短文本形式的数据, 快速并准确地获取所需的关键信息以及提高聚类的效率、准确率一直都是人们关注的重点. 但短文本固有的特点, 使得传统的特征权重计算方法无法准确计算. 因此, 学者们采用不同的方法去解决这一缺陷, 总体分为三个方面, 一用特征子集评价方法从特征空间上改进, 包括信息增益[1]、卡方检验 (CHI-sqare, CHI)[2]、期望交叉熵 (Expected Cross Entropy, ECE)[3]等, 这些评价算法在给定阈值的情况下, 通过计算文本集中每个特征项的权重值, 选择特征项的权重值大于阈值的特征加入特征子集或选择权重值最大的特征项子集直到满足特征子集大小阈值. 例如李凯齐, 刁兴春等[4]提出一种改进的特征权重计算方法, 通过引入信息论中信息增益的概念, 实现对短文本特征分布具体维度的综合考虑, 克服传统公式存在的不足. 实验结果表明, 改进后的特征权重计算算法在计算特征权重时更加有效. 二在搜索空间策略上进行改进, 包括顺序选择算法、遗传算法、粒子群算法等, 这些算法通过搜索叠加的方式在实现特征空间降维的同时提高算法自身的准确率. 例如杜坤, 刘怀亮等[5]考虑特征项间的语义关联构造复杂网络并进行特征选择, 定义类别相关系数并结合特征选择结果, 提出一种改

① 收稿时间: 2017-03-24; 修改时间: 2017-04-13; 采用时间: 2017-04-17; csa 在线出版时间: 2017-12-22

进的特征权重计算方法, 并进行中文文本分类实验. 实验结果表明, 改进后的算法较 TFIDF算法有更好的分类效果. 三从特征属性上进行改进, 包括词频[6]、特征在文本中的位置[7]、词共现分析等, 以上特征属性作为影响因子加入实验中. 例如李欣蓬等[8], 提出双维度特征关系和特征位置对类别学习的影响, 实验结果反映了词性对于特征权重的积极影响.

多种实验表明从特征属性上改进特征权重要优于其他两种方法[9-11]. 其中于海燕等[12]提出一种基于词性嵌入的特征权重计算方法, 从词性对情感分类的贡献度嵌入到 TF-IDF 算法中. Gang Wang, Zhu Zhang 等[13]提出基于词性情绪分类的 PSO-RS 算法, 实验表明 POS-RS 情绪分类可以作为一个可行的方法, 有可能被成功地应用于其他文本分类问题. 这些研究表明词性对于特征权重上的改进能够提高后续验证实验的准确率, 对于本文的研究有重大意义. 本文从词性属性出发, 提出一种新的基于词性特征的特征权重计算算法 (Translation Decision Model Of Quantum-behaved Particle Swarm Optimization, TDQO). 在特征选择阶段中将词性引入到翻译决策模型 (Translation Decision Model, TD) 中, 以改进后的 TDQO 算法对聚类的效率与准确性进行改善.

2 传统的特征权重计算方法

传统的特征权重计算方法有很多, 例如 TF 算法、TF-IDF 算法、PageRank 算法等等. 其中 TF 算法仅从文本词频的角度考虑, 一方面考虑到了高词频所带来的高权重, 另一方面却暴露其大量无意义词所产生的高冗余、高复杂度等缺点. 另外 PageRank 算法是根据网页中的超链接链入的网页数来判断某个网页是否重要. 本文语料为文本数据, 为了使初始化的特征权重有较好的可信度, 本文在计算初始权重计算方法上选择 TF-IDF 算法.

2.1 TF-IDF 算法

TF-IDF 算法在计算特征权重时考虑三点: 词频 (tf)、反文档频率 (idf) 以及归一化 (normalization). 其中词频 tf 表示特征在该文档中出现的频率; 反文档频率表示特征在各个文档中的区分能力; 归一化 (normalization) 用来防止偏向长文档. 考虑三个条件, TF-IDF 公式可以表示如下:

$$weight_{tfidf}f(t_k)=\frac{tf(t_k,d_i)*lb(N/n_k+0.01)}{\sqrt{\sum_{k=1}^{m}[tf(t_k,d_i)*lb(N/n_k+0.01)^2}} \quad (1)$$

其中 $tf(t_k, d_i)$ 表示特征 t_k 在文档 d_i 中出现的频率. N 表示为文档总数. m 表示文档中的特征数. n_k 表示包含特征 t_k 的文档数.

2.2 TF-IDF 算法的缺陷

TFIDF 认为一个特征出现的文档频率越小, 则区分类别文档的能力越大. 逆文本频度 IDF 在一定程度上抑制无意义特征, 但在另一方面重要特征的凸显也造成无意义标注. 而 TFIDF 的计算为 IDF 对于 TF 的权重调整, IDF 本身无法有效区分重要特征及无意义特征分布, 使得 TFIDF 计算特征权重的精度并不是很高.

举例说明该算法的不足. 假设总文档量为 100 篇. 在 2000 特征词的文档中“亲情”, “友情”, “的”, “魅力”分别出现 30, 90, 100, 5 次, “亲情”出现在 20 篇文档中, “友情”出现在 90 篇文档中, “的”出现在 100 篇文档中, “魅力”出现在 5 篇文档中. 在其 TF, IDF, TF-IDF 数据如表 1.

从表 1 可以分析出“友情”与“的”权重最低, 但是却表示两个极端, “的”对于特征来说是无意义的特征, 只会增加特征冗余, 而“友情”却是每篇文档的主题词, 经文本聚类可以将文档归为一类. 由此可见 TF-IDF 算法在特征的重要程度上无法准确判断.

表 1 特征在 TF, IDF, TF-IDF 上的表现

特征	TF	IDF	TF-IDF
亲情	0.015	0.698	0.01047
友情	0.045	0.0458	0.002061
的	0.05	0	0
魅力	0.0025	1.301	0.003275

3 TDQO 特征权重改进算法

TDQO 算法在 TF-IDF 算法的基础上引入词性加权权重 (TDF) 以及特征词作为某种词性出现概率 (PF), 由此改进 TF-IDF 算法. 其中 TDF 加权了词性特征权重, 例如在文本中名词相对于动词、形容词更能代表一篇文档的主题特征, 对于词性加权有效权衡了词性所带来的权重影响. 而 PF 有效抑制大量某一种词性权重影响.

3.1 词性加权权重

词性加权公式如下:

$$TDF=\sum_{j=1}^{n}x_i \quad (2)$$

其中 n 为特征作为粒子的总群数, x_i 表示第 i 个特征粒子, j={1, 2, 3}表示某种词性.

3.2 特征作为某种词性概率

特征词为某种词性概率公式如下:

$$PF = p(t_j|t) \tag{3}$$

其中 t_j 表示特征 t 出现的词性特征.

3.3 TDQO 算法

大多数的短文本在文本预处理阶段, 通过词性筛选, 保留下所需要的词性, 李英[14]提出基于词性的特征预处理方法, 在文本预处理环节过滤掉副词、叹词等贡献度很小的词性, 只保留对分类贡献较大的名词、动词、缩略词等, 实验证明这一方法有效的降低了文本空间的特征维度. 特征权重计算为特征空间中的文本向量的每一维确定合适的数值, 以表达对应特征在文本的重要程度. 特征 t_i 在文本 d_i 中的权重表示为 $w_{i,j}=w(t_i, d_i)$, 文本 d_i 的权重向量表示为 $w_j=w(d_j)$.

在特征选择算法之后进行词性筛选, 只保留名词、动词、形容词. 一方面更好地通过词性将词频中较高的干扰词性过滤掉, 另一方面可以通过观察哪些词性的词本身虽不具有特征属性, 但对权重产生影响, 比如标题中一些权重较高的词.

本文在不同词性上进行不同程度的加权, 得出一种基于词性的权重计算方法公式如下:

$$weight_{TDQO}(t_k) = weight_{tfidf}(t_k) * PF * TDF \tag{4}$$

其中 $PF*TDF$ 表示为特征 t 在改进后的量子粒子群优化算法的最优词性加权总值.

3.3.1 TDQO 算法流程

TDQO 算法在量子粒子群算法的基础上引入 TD 模型, 它的范围搜索能力极大高于一般 QPSO 算法. 以下介绍 TDQO 算法具体实现过程.

(1) 初始化粒子速度与位置. 图 1 模块①为 TDF 的计算通过迭代不断判断局部极值 pBest 和全局极值 gBest[15]来更新自己的速度及位置, 最终找到最优解. 粒子根据公式 (5)(6) 来优化自己的速度和位置, 公式 (7) 为词性加权权重, 即 TDF.

$$\begin{aligned} V_{i,j}(t+1) = {} & V_{i,j}(t) + C_1 * r_{i,j}(t) * \left[P_{i,j}(t) - X_{i,j}(t)\right] \\ & + C_2 * r_{2,j}(t) * \left[G_j(t) - X_{i,j}(t)\right] \end{aligned} \tag{5}$$

$$X_{i,j}(t+1) = X_{i,j}(t) + V_{i,j}(t+1) \tag{6}$$

$$TDF = X_{i,j}(t+1) \tag{7}$$

其中, i 表示第 i 个粒子, j 为粒子的第 i 维, t 为进化代数, C_1, C_2 为加速方向常数, r_1, r_2 为[0, 1]上均匀分布的随机数.

(2) 以 (0, 1) 随机函数赋值 X_i, 并将其作为初始特征权重, V_i=2.0, 初始化每个粒子, 使用 k-means 聚类算法, 计算聚类准确率作为粒子的适应度值. 粒子在迭代过程中, 当前位置的适应度值大于局部或全局最优解的适应度值, 则更新为粒子当前位置, 否则继续迭代, 最终输出计算的词性加权权重.

(3) 建立翻译决策模型, 将每个特征作为粒子, 并标注词性及对应的布尔值. 图 1 模块②中 TDQO 算法中建立的 TD 模型是最大熵[16]模型的分支模型, 也是 PF 计算的过程. 其中 TD 模型函数的建立用来计算 PF 值, 即特征作为某种词性出现概率. 其公式如下:

$$PF = p(t_j|t) = \sum_{y} e^{\sum_i \lambda_i f_i(x,y)} \tag{8}$$

其中 λ_i 初始化为 0, $f_i(x, y)$ 表示定义的特征函数, x 表示特征, y 表示对应词性.

(4) 计算当前模型分布期望, 计算最优估计, 最终得到粒子作为词性权重的加权权重.

TDQO 算法流程图如图 1.

4 实验与分析

使用爬虫工具在豆瓣小说上获取 22 篇小说书评, 共计 24 450 条评论. 经预处理剩有 17 765 个词, 通过 TF-IDF 计算初始权重, 并设置阈值为 0.01, 过滤大量冗余特征. 此时剩有 2215 个词作为后续对比实验的初始特征集, 根据建模需要, 需再次对词性进行降维, 只保留名词、动词、形容词, 最终特征选择的词剩有 1816 个.

为了验证词性对文本的贡献度有助于提高聚类的准确率, 本文通过 TF-IDF 算法、QPSO 算法、TDQO 算法进行对比实验. 其中 TF-IDF 方法得到特征向量并直接进行聚类输出; QPSO 算法中不标记词性, 通过粒子迭代得到最优加权权重, 其中粒子个数为 39 952 个, 迭代次数为 100 次, 得到未加权词性的特征权重, 进而进行聚类输出; TDQO 算法实验在 QPSO 算法实验的基础上, 引入 TD 模型, 加权计算特征作为某种词性出现的概率并聚类输出. 实验环境为 Windows 8 操作系统, 2 GB 内存, 利用 MATLAB 及 PYTHON 开发.

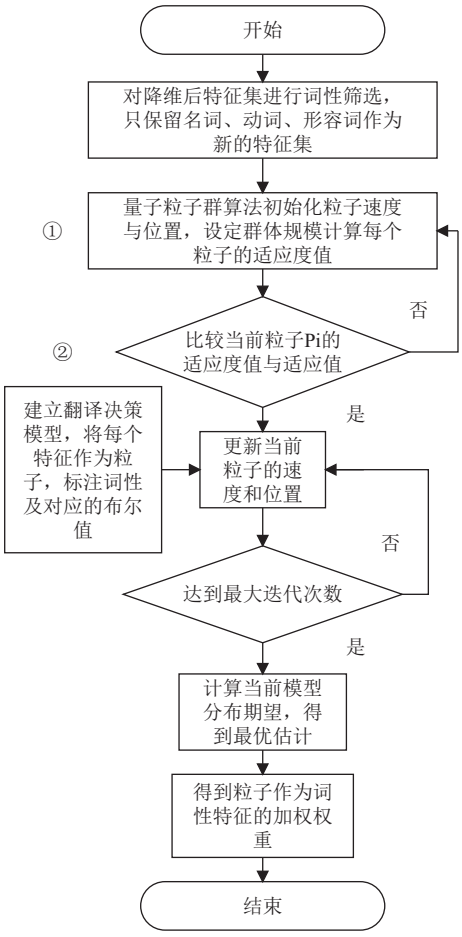

图 1 TDQO 算法流程图

输入: TF-IDF 算法权重数据标记粒子词性, 粒子总数
输出: 改进后的特征权重加权, 改进前后的 F 值

(1) 使用中国科学院计算技术研究所 ICTCLAS2014 分词器对原始语料进行分词处理;
(2) 使用 TF-IDF 算法对词频进行排序, 选取词频在 0.01 以上的词作为新的特征集; 此处是避免大量的非有效特征增加特征冗余;
(3) 对新的特征集进行词性筛选, 只保留名词、动词、形容词;
(4) 引入 TD 模型的量子粒子群优化算法. 通过 TD 模型建模函数得到特征作为词性出现的概率加权到粒子迭代中, 当前位置的适应度值大于局部或全局最优解的适应度值, 则更新为粒子当前位置, 否则继续迭代, 最终输出计算的词性最优加权权重;
(5) 将得到的加权后的数据经 k-means 聚类, 通过修改 k 值, 在不同类别中使用三种方法进行实验并得出结论.

4.1 实验数据分析

为验证提出方法的有效性, 将 TF-IDF 算法、QPSO 算法及 TDQO 算法三种方法进行聚类实验, 以检验它们在文本挖掘中的表现. 实验采用聚类领域常用的 F-measure 作为指标来评价文档聚类方法的效果.

F-measure[17]是一种结合了 precision 和 recall 的聚类评价指标. F-measure 的取值范围为[0, 1]. 对应的检索粒子分布表如表 2.

表 2 检测粒子分布

	相关	不相关
检索到的粒子	A	B
未检索到的粒子	C	D

在翻译决策模型建模中, 将特征转化成随机粒子. 根据文档粒子采用分散规则赋值, 转化的粒子共 39952 个, 与之相对应产生 39952 个初始权重, 相同的特征在分散文档中的权重也会有所不同, 因而在建模过程中, 特征用集中的权重表示, 并用 TRUE 和 FALSE 标注. TRUE 的情况以二进制 1 代表, FALSE 的情况以二进制 0 代表, 粒子词性特征以三维向量表示, 并转化成相应十进制, 取值为 rand(2, 4, 6), 同时量子粒子群算法仍然使用分散初始权重生成向量作为输入. 初始化粒子速度与位置同步进行, 设置位置 x_i=(0, 1), 速度 v_i=2.0, 迭代次数 MAXGEN=100, 加速常数 C_1, C_2 均为 2.0.

为了验证在引入翻译决策模型的量子粒子群优化算法对聚类的准确度, 将三种方法计算出特征权重构造特征向量, 并进行聚类上的评价比较. 其中聚类类别 k=[3, 7], 实验数据 recall 值及 F 值上的比较如表 3、表 4 所示.

表 3 三种权重计算方法在聚类上 recall 比较

聚类类别	实验方法		
	TF-IDF算法	QPSO算法	TDQO算法
3	0.4286	0.5	0.6
4	0.5	0.5714	0.6667
5	0.5556	0.625	0.7143
6	0.5455	0.6	0.6667
7	0.5385	0.5833	0.6364

表 3、表 4 中的 3 种实验算法在聚类指标 recall 值及 F-measure 值上均表现出无论 k 取何值, TDQO 算法始终要优于前两种算法.

根据评价标准 F 值绘制成折线图如图 2 所示.

表 4 三种权重计算方法在聚类上 F-measure 比较

实验方法 聚类类别	TF-IDF算法	QPSO算法	TDQO算法
3	0.4615	0.5455	0.6667
4	0.5333	0.6154	0.7273
5	0.5882	0.6667	0.7692
6	0.5714	0.6316	0.7059
7	0.56	0.6087	0.6667

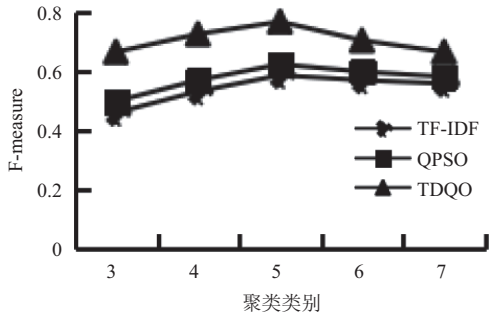

图 2 三种权重计算方法在 F 值走势图

从图 2 折线趋势图可以明显看出, 使用 QPSO 算法提高了聚类准确率, 而本文提出的 TDQO 算法更加有效地提高了聚类准确率. 当类别越大或越小时, QPSO 算法准确率虽然与 TF-IDF 算法准确率很接近, 但是整体准确率有所提高; 当聚类类别数为 5 时, 准确率提高最大 (7.85%). TDQO 算法在各个类别上的准确率均大大高于 QPSO 算法的准确率, 这证明了不同的词性对于文本聚类的贡献度是有影响的. 从整体上来看, 当聚类类别从 3 开始, 聚类效果呈上升趋势, 当类别数超过 5 时, 普遍的呈下降趋势. 所以聚类 k 值为 5 时, 聚类准确率达到最高.

此时, 将 k 设定 5 作为不变量, 测试用三种不同方法在不同特征维度中的聚类效果. 具体实验数据如图 3-图 5 所示.

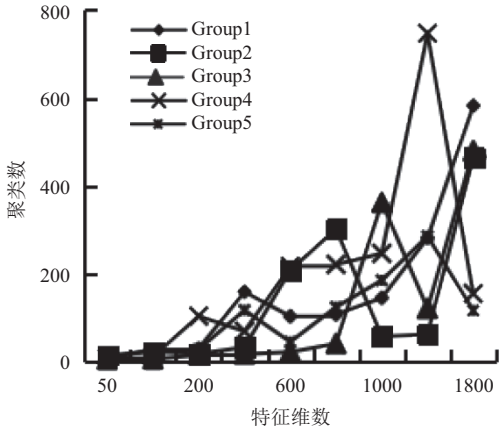

图 3 TF-IDF 算法在各维度上聚类效果

从图 3 和图 4 可以看出共同点: 在低特征维度上聚类分布改善不明显, 在高特征维度上, 聚类分布效果较好. 区别在于 TF-IDF 算法在[1500, 1800]高维度区间上的聚类效果要好于 QPSO 算法, 而 QPSO 算法在[600, 1000]区间上展现了较好的聚类效果.

从图 5 得出结论: 随着特征维数的增大, 聚类分布显著. 与图 3 和图 4 比较来看, TDQO 算法在[200, 1800]区间的聚类分布依然表现出良好的聚类效果. 本文提出的 TDQO 算法一方面提高聚类准确率, 另一方面在不同特征维度也展现了较好的聚类效果, 同时具有更广泛的应用范围.

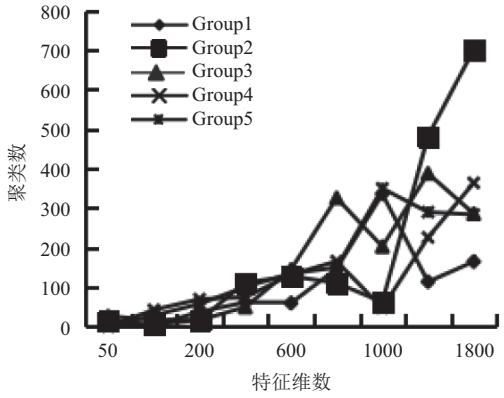

图 4 QPSO 算法在各维度上聚类效果

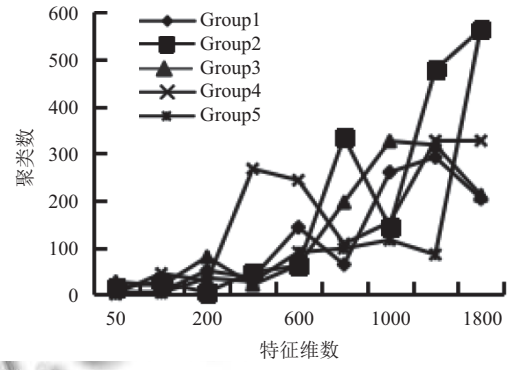

图 5 TDQO 算法在各维度上聚类效果

5 结束语

目前短文本在特征权重计算的方法上很大程度上仍按照长文本的特征计算方法, 然而短文本在特征属性上更具有贡献度, 传统的方法会降低其准确率. 本文在现有的特征权重计算方法的基础上, 提出了 TDQO 算法[18]. 该算法引入某种词性作为特征出现时的概率, 并将粒子作为特征在迭代中寻找最优权重配比. 实验表明该算法在聚类中准确率有所提高, 因此也证明了词性权重对于聚类结果是有影响的. 另外, 对于聚类类别 k 值的选取也会对实验结果有所影响. 对于本文的算法依然还存在改进的地方, 可以在实验的不同环节或者算法内部提高效率.

参考文献

1 Reineking T. Active classification using belief functions and information gain maximization. International Journal of Approximate Reasoning, 2016, (72): 43–54. [doi: 10.1016/j.ijar.2015.12.005]

2 Rempala GA, Wesolowski J. Double asymptotics for the chi-square statistic. Statistics & Probability Letters, 2016, (119): 317–325.

3 Zhong RX, Fu KY, Sumalee A, *et al*. A cross-entropy method and probabilistic sensitivity analysis framework for calibrating microscopic traffic models. Transportation Research Part C: Emerging Technologies, 2016, (63): 147–169. [doi: 10.1016/j.trc.2015.12.006]

4 李凯齐, 刁兴春, 曹建军. 基于信息增益的文本特征权重改进算法. 计算机工程, 2011, 37(1): 16–18.

5 杜坤, 刘怀亮, 郭路杰. 结合复杂网络的特征权重改进算法研究. 现代图书情报技术, 2015, 31(11): 26–32. [doi: 10.11925/infotech.1003-3513.2015.11.05]

6 lbrahim A, Cowell PE, Varley RA. Word frequency predicts translation asymmetry. Journal of Memory and Language, 2017, (95): 49–67. [doi: 10.1016/j.jml.2017.02.001]

7 Kao CY. The effects of stimulus words ' positions and properties on response words and creativity performance in the tasks of analogical sentence completion. Learning and Individual Differences, 2016, (50): 114–121. [doi: 10.1016/j.lindif.2016.07.015]

8 李欣蓬. 双维度特征关系和特征位置对类别学习的影响[硕士学位论文]. 天津: 天津师范大学, 2009.

9 黄文涛, 徐凌宇, 李严, 等. 基于柔性区间的多文本融合提取方法. 计算机工程, 2007, 33(24): 217–219. [doi: 10.3969/j.issn.1000-3428.2007.24.076]

10 吴光远, 何丕廉, 曹桂宏, 等. 基于向量空间模型的词共现研究及其在文本分类中的应用. 计算机应用, 2003, 23(S1): 138–140.

11 许建潮, 胡明. 中文 Web 文本的特征获取与分类. 计算机工程, 2005, 31(8): 24–25, 39.

12 于海燕, 陆慧娟, 郑文斌. 情感分类中基于词性嵌入的特征权重计算方法. 计算机工程与应用, 2016, 53(22): 121–125.

13 Wang G, Zhang Z, Sun JS, *et al*. POS-RS: A random subspace method for sentiment classification based on part-of-speech analysis. Information Processing & Management, 2015, 51(4): 458–479.

14 李英. 基于词性选择的文本预处理方法研究. 情报科学, 2009, 27(5): 717–719, 738.

15 Sun J, Xu WB, Feng B. A global search strategy of quantum-behaved particle swarm optimization. Proceedings of 2004 IEEE Conference on Cybernetics and Intelligent Systems. Singapore, Singapore. 2004. 111–115.

16 Li R, Tao X, Tang L, *et al*. Using maximum entropy model for Chinese text categorization. Journal of Computer Research &Development, 2005, 42(1): 578–587.

17 常鹏, 马辉. 高效的短文本主题词抽取方法. 计算机工程与应用, 2011, 47(20): 126–128, 154. [doi: 10.3778/j.issn.1002-8331.2011.20.036]

18 奚茂龙, 盛歆漪, 孙俊. 基于多维问题的交叉算子量子粒子群优化算法. 计算机应用, 2015, 35(3): 680–684. [doi: 10.11772/j.issn.1001-9081.2015.03.680]