方差和词向量用于文本降维的研究①

王甜甜, 康 宇

(中国科学技术大学 自动化系, 合肥 080602)

摘 要: 文本分类中的高维数据和噪声一直是影响文本分类准确率的主要因素, 特征选择和特征提取是降维和去噪的主要手段. 本文提出根据词的类间概率分布方差和文档分布方差改进 TF-IDF 的特征选择方法(VAR-TF-IDF), 调整 Word2vec 中的 CBOW+HS 词向量训练框架, 用特征词词向量的叠加作为文本的特征向量, 有效地提高了文本分类的准确率和召回率. 实验算例证明了所提方案的有效性.

关键词: 方差; 词向量; 文本分类; 衰减系数

Research of Variance and Word Embedding in Text Classification

WANG Tian-Tian, KANG Yu

(Department of Automation, University of Science and Technology of China, Hefei 080602, China)

Abstract: High dimensional data and noise have always been the major factors affecting the accuracy of text classification. Feature selection and feature extraction is the main methods of dimensionality reduction and denoising. In this paper, the words probability distribution variance and document distribution variance is used to improve the TF-IDF feature selection method (VAR-TF-IDF). After selecting good features, it tuned the CBOW+HS frame work of word2vec. The superposition of word embedding of the selected words is used as eigenvector which could improve accuracy of text classification. Experiment shows the proposed method is effective.

Key words: variance; word embedding; text classification; attenuation efficient

1 引言

随着互联网的发展, 人们的生活的方方面面, 沟通交流, 信息获取, 互动娱乐都高度网络化, 网民的数量也是指数增长速度在增加. 网络中获取新闻更是现在人们主要的信息来源, 网络中的新闻每天以数以十万计的速度增长着, 如果这些新闻不加分类的呈现在人们面前, 人们很难从海量新闻中找到自己需要的信息, 新闻分类技术的发展迫在眉睫. 新闻分类是文本分类的一种. 文本分类中的难题之一就是如何从高维空间中提取有用的特征, 以适应文本分类算法和提高分类的准确率. 一篇文本中的词汇少则几百个多则几千也有, 文本集里面可能出现的词更是数以百千万计. 我们不可能把这些词全部输入分类器, 数据维度高增加了计算的复杂度, 里面参杂的噪声对分类精度也有很大影响.

特征选择和特征提取则是降维和去噪的主要手段. 特征选择是指从文本所有特征中根据特定的评价函数选出指定数量的特征[1]. 特征提取是指将选出的特征表示成分类器可以识别的向量的过程[2]. 现有的特征选择和特征提取在文本分类中虽有着不错的效果, 但也都存在着一些缺陷. 例如, 在特征选择方面, TF-IDF 方法和信息增益方法本质上是针对“文档集”做特征选择而忽略了特征词类间分布情况; 而卡方检验方法和互信息方法有低频词倾向, 夸大了低频词的作用. 有很多研究者针对他们的缺陷提出了不同的改进方法: 赵世奇[3]是对每个类别分别计算自己的特征集合, 各个类别的特征集合的并集作为全局的特征集合; 张玉芳[4]提出根据特征类的内分布词频、文档频率、词频

① 基金项目:国家高技术研究发展计划(863)(2014AA06A503);国家自然科学基金(61422307)
收稿时间:2016-03-07;收到修改稿时间:2016-04-21 [doi:10.15888/j.cnki.csa.005473]

和全局的文档频率, 词频差异来决定是否选择该特征, 倾向于选择和文本正相关并且只对一个类别有好的分辨能力的特征; 彭时名[5]用类别内包含特征的文档数量去改善IDF, 一定程度上提高了特征的质量, 使得选出的特征更有类别代表性; 邱云飞[6]用词频改善卡方检验的低频词缺陷, 用特征词方差修正所有类别中分布均匀的特征的权重, 使之降低, 用r因子衡量特征词概率与类中所有特征平均概率的偏离程度, 这几个因子相乘一起改进卡方检验的特征选择, 这种调整因子过多的情况会导致其中一种因素起到主导作用, 而削弱其他因素对特征权重的影响. 特征提取方面, 一直以来, 向量空间模型是典型的文本特征表示方法, 这种特征表示方法是稀疏的, 高维的: 一方面, 基于向量空间模型的文本特征向量将单词看做词典里的一个索引编号, 这种情况下特征的语义是原子性的, 不能衡量特征之间的语义语法关系, 如同义词、近义词、反义词、词性、词的隐含语义关联等; 另一方面是基于向量空间模型的的文本特征向量的维数就是特征的个数, 通常是上万维特征, 特征向量是高维并且稀疏的, 维数过高会增加计算难度, 维数太少选择的特征又不能很好地代表文本.

词向量的思想是通过训练将某种语言的一个词映射成一个固定长度的向量, 语料中所有的词向量组成一个向量空间, 向量空间中词向量之间的欧式距离和余弦相似度就可以判断他们之间的语义语法上的联系. 词向量在自然语言处理中应用广泛, 也取得了巨大的成功. Zhang[7], Sun[8]等人改进词向量学习方法, 成功应用到中文分词, 取得了目前最好的结果; Socher[9]用词向量的表达进行微博的情感分析. 在文本分类方面, 江大鹏[10]将词向量用于短文本分类, 取得了不错的效果; Hu 开始将词向量用在文本分类算法里面[11], 证明词向量的使用确实可以提高文本分类的效果. 好的词向量会使的自然语言处理的效果更好.

针对上述特征选择和特征提取方法的不足和词向量在自然语言处理中的成功应用. 本文提出根据词的类间概率分布方差和文档分布方差改进 TF-IDF 的特征选择方法(VAR-TF-IDF), 并调整 Word2vec 的 CBOW+HS 训练框架, 训练出词向量后, 用文章所选出特征的词向量叠加作为文本的特征向量.

2 基于方差的特征选择和Word2vec的CBOW+HS词向量模型

特征选择的目的是为了选出对类别有用的分类信息, 对分类有用的特征一定是在本类别出现频概率和其他类别的出现概率差异大的特征[12]. 我们注意到方差就是一个衡量特征在类间的分布概率差异大小的好的度量方式, 将特征的类间概率分布方差和文档分布方差信息融合到特征选择中, 必然会有效地提高文本分类效果. 同时, 特征提取所用的词向量训练过程中在用词的上下文去预测目标词时, 上下文中的词对目标词的贡献随着距离目标词的远近呈指数衰减, 将这种衰减方式应用到 Word2vec 的 CBOW+HS 训练框架中, 这样训练出的词向量可以更准确地代表词的语义.

2.1 特征的概率分布方差和文档分布方差(VAR-TF-IDF)

理想特征词当然是只在一个类别的大部分文档里出现频率比较高, 在其他所有类别的文档里出现的频率都很低, 但是符合这种条件的特征数量是比较少的. 对于大部分的有效特征来说, 他们会在某几个类别的文本中都有较高的出现概率, 在其他类别中出现的概率则很低. 例如"电商"既可能在"互联网"类别的新闻中出现, 又可能"财经"类别的新闻中出现; 国外地名或国家名字既可能在"旅游"类别的新闻中出现, 又可能在"军事"类别的新闻中出现. 为了提高这类特征的选择质量同时保证"类别"专有特征的选择效果[3], 这里提出用词的类间概率分布方差和文档分布方差去改进 TF-IDF 特征选择的方法.

设 w_i 是文本集中的一个词, 词 w_i 的类间概率分布方差为:

$$\mathrm{var}(w_i)=\sqrt{\frac{\sum_j p(w_i,c_j)-\overline{p(w_i)}}{c}} \tag{1}$$

c 为类别总数,

$$p(w_i,c_j)=\frac{N(w_i,c_j)}{\sum_i N(w_i,c_j)} \tag{2}$$

式(2)是词 w_i 在类别 c_j 中的出现的概率, $N(w_i,c_j)$ 代表 w_i 在类别 c_j 中出现的词频, $\sum_i N(w_i,c_j)$ 则是类别 c_j 中的总词频.

$$\overline{p(w_i)}=\frac{1}{c}(\sum_j p(w_i,c_j)) \tag{3}$$

是词 w_i 的平均出现概率.

同理, 定义词 w_i 的类间文档分布方差为:

$$\mathrm{var}_D(w_i)=\sqrt{\frac{\sum_j p_D(w_i,c_j)-\overline{p_D(w_i)}}{c}} \tag{4}$$

其中,

$$p_D(w_i,c_j)=\frac{D(w_i,c_j)}{D(c_j)} \tag{5}$$

式(5)是特征词 w_i 的文档概率, $D(w_i,c_j)$ 为类别 c_j 中出现 w_i 的文档数量, $D(c_j)$ 为类别 c_j 中的总词数.

$$\overline{p_D(w_i)}=\frac{1}{c}(\sum_j p_D(w_i,c_j)) \tag{6}$$

是词 w_i 的平均文档概率.

用特征的类间概率分布方差和文档分布方差乘积的对数去修正 TF-IDF 来计算特征权重,

文档 j 中词 w_i 的权重是:

$$weight(w_i,j)=TF_{i,j}*IDF_i\log(\mathrm{var}_D(w_i)*\mathrm{var}(w_i)) \tag{7}$$

其中 $TF_{i,j}$ 为 w_i 在文档 j 中出现的次数, IDF_i 为 w_i 在训练语料上的逆文档频率值.

$$TF_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}} \tag{8}$$

$$IDF_i=\log\frac{|D|}{\sum_j(w_i\in d_j)} \tag{9}$$

文档分布方差可以选出在不同类别文本间特征词分布差异大的特征, 这样的特征是有类别代表性的特征, 文档分布方差大的特征往往对于一些类别来说是正相关特征, 对于某些类别来说是负相关特征, 正相关和负相关的特征都可以选择出来. 正相关特征有利于提高文本分类的准确率, 负相关特征有利于提高文本分布的召回率, 词的概率分布方差则可以修正文档分布方差的低频词缺陷. 通过概率分布方差和文档分布方差乘积的对数作为一个因子修正 TF-IDF 的特征评价权值, 可以提高所选特征的质量.

特征选择的过程就是对于语料库中的每篇文本, 根据权重评价函数(7)计算这篇文本中每个特征的权重, 根据权重从大到小排序, 选出排序靠前的指定数量的特征作为文本的特征词.

2.2 词向量的训练原理

Google 的 Word2vec 是对神经网络概率语言模型的实现, 是概率语言模型与神经网络的组合. Word2vec 在训练语言模型的过程中将词映射到一个指定维数的向量空间上面, 每个词向量是向量空间中的一个点.向量表示语义语法的准确程度和训练词向量所用语料的类型及语料库的大小密切相关[13-15]. 一般来说, 训练数据达到几百兆的情况下, 可以认为得到的词向量比较精确, 两个词向量的余弦距离就可以表示词之间的相似度[16]. 例如, 660M 新闻文件训练出来的词向量中计算得到与“银行”的词向量距离最近的八个词向量代表的词分别是“分行”, “总行”, “农行”, “浦发”, “工行”, “建行”, “招行”, “中行”, 词向量已经相对准确.

Word2vec 实现了 continuous bag of words 和 skip-gram 两种训练方式, continuous bag of words 训练方式是根据词的上下文去预测目标词, skip-gram 训练方式是根据目标词去预测它的上下文[17].

两种训练模型均包括输入层, 投影层, 输出层. 除此之外, Word2vec 提供了两种优化方法来提高词向量的训练效率, 分别是 Hierarchy Softmax 和 Negative Sampling, 训练模型和优化方法组合可以得到四种训练框架. 本文主要是对 CBOW+Hierarchy Softmax (HS)[18]训练框架进行改进.

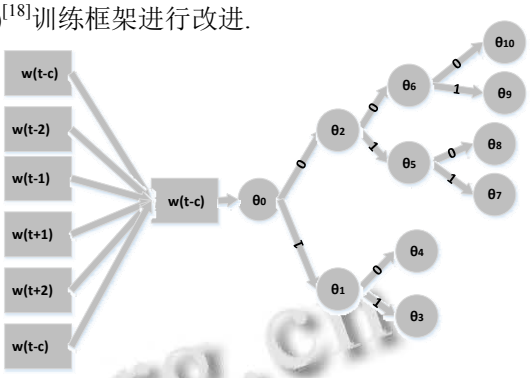

图 1 CBOW+HS 训练框架

CBOW+HS 框架是在给定一个目标词 w_t 及其上下文 $context(w_t)$ 的情况下预测 w_t. 其输入层是 $context(w_t)$, 上下文 $context(w_t)$ 的定义是:

$$context(w_t)=w_{t-c},.....,w_{t-1},w_{t+1},.....w_{t+c} \tag{10}$$

c 是 w_t 前后应该考虑的词的个数, 即窗口长度.

原本的 CBOW+HS 训练框架认为 $context(w_t)$ 中的词对于目标词 w_t 的贡献大小随着二者距离变化是线性衰减的, 距离目标词越近的词对目标词 w_t 的贡献越大. 原框架采用在 *0* 到 *c* 之间随机选取数字作为当前窗口长度的方式来实现统计意义上上下文中的词对预测目标词贡献的线性衰减. 假设当前的随机窗口长度是 b, 那么投影层就是:

$$X_{w\_origin}=\sum_{i=t-b,i\neq 0}^{i=t+b}v(w_i) \tag{11}$$

$v(w_i)$ 指 w_i 的词向量.

本文则认为上下文中词对预测目标词的作用随着距离变化是呈指数衰减的, 根据 w_{t-i} 距离 w_t 的词的数量 i 构造出指数衰减的权重因子 $e^{-d_{w_{t-i}}}$.

这时候投影层是 $context(w_t)$ 中所有词向量的叠加.

$$X_w = \sum_{i=t-c, i\neq 0}^{i=t+c} e^{-d_{w_{t-i}}} v(w_i) \tag{12}$$

其中,

$$d_{w_{t-i}} = \frac{i}{2} \tag{13}$$

输出层是一个二叉树, 它是以训练语料中出现的所有词为叶子节点, 以词频为权值构造出的一棵 Huffman 树. 由 Huffman 树的构造规则可知叶子节点共有 $|D|$ 个, 每个对应词典中的一个词, 非叶子节点有 $|D|-1$ 个.

在用 $context(w_t)$ 预测 w_t 时, 假设根节点到叶子节点的路径为 p^w, 路径 p^w 上包含的节点个数为 l^w, $p_1^w, p_2^w, ..., p_{l^w}^w$ 是路径 p^w 上的节点, 其中 p_1^w 代表跟节点, $p_{l^w}^w$ 代表叶子节点, $d_2^w, d_3^w, ..., d_{l^w}^w$ 代表词 w_t 的 Huffman 编码, $\theta_1^w, \theta_2^w, ..., \theta_{l^w}^w$ 代表 p^w 上非叶子节点对应的词向量. 将 Huffman 树中每一个分支看做一个二分类, 编码为 *1* 的作为负类, 编码为 *0* 的作为正类, 即向左的分支作为负类, 向右的分支作为正类. 每个节点进行二分类的时候, 采用逻辑回归计算其分到正类或者负类的概率, 那么分到正类的概率是:

$$p(d_{i+1}^w \mid X_W, \theta_i^w) = \delta(X_W, \theta_i^w) = \frac{1}{1+e^{-X_w^T \theta_i^w}} \tag{14}$$

分到负类的概率是:

$$1 - \delta(X_w, \theta_i^w) \tag{15}$$

沿着路径将所有的概率相乘随后取对数就是所要优化的目标:

$$\begin{aligned} &p(w_t \mid context(w_t)) = \prod_{i=0}^{l^w-1} p(d_{i+1}^w \mid X_w, \theta_i^w) \\ &= \prod_{i=0}^{l^w-1} [\delta(X_w, \theta_i^w)]^{1-d_{i+1}^w} [1-\delta(X_w, \theta_i^w)]^{d_{i+1}^w} \end{aligned} \tag{16}$$

将条件概率带人似然函数, 得到最终优化目标:

$$\begin{aligned} L &= \sum_{w\in c} \log\left(p(w_t \mid context(w_t))\right) \\ &= \sum_{w\in c} \sum_{i=1}^{l^w-1} (1-d_{i+1}^w) \log(\delta(X_w, \theta_i^w)) \\ &+ d_{i+1}^w \log(1-\delta(X_w, \theta_i^w)) \end{aligned} \tag{17}$$

为了方便计算, 可以单独计算累加运算符里面的各项:

$$\begin{aligned} L(X_w, \theta_i^w) &= (1-d_{i+1}^w) \log(\delta(X_w, \theta_i^w)) \\ &+ d_{i+1}^w \log(1-\delta(X_w, \theta_i^w)) \end{aligned} \tag{18}$$

分别求 $L(X_w, \theta_i^w)$ 对的下降梯度:

$$\frac{\partial L(X_w, \theta_i^w)}{\partial X_w} = (1-d_{i+1}^w - \delta(X_w, \theta_i^w))\theta_i^w \tag{19}$$

$$\frac{\partial L(X_w, \theta_i^w)}{\partial \theta_i^w} = (1-d_{i+1}^w - \delta(X_w, \theta_i^w))X_w \tag{20}$$

非叶子节点的向量更新过程:

$$\theta_i^w \longleftarrow\!\!- \theta_i^w + \alpha(1-d_{i+1}^w - \delta(X_w, \theta_i^w))X_w \tag{21}$$

其中 α 是学习率.

对于上下文中的每一个词的词向量 $w_i \in context(w_t)$ 对应的叶子节点向量的更新过程是:

$$v(w_j) \longleftarrow\!\!- v(w_j) + \alpha(1-d_{i+1}^w - \delta(X_w, \theta_i^w))\theta_i^w \tag{22}$$

其中 $v(w_j)$ 代表 w_j 的词向量, α 是学习率.

word2vec 训练出的语言模型把每个词映射为指定维数的向量, 结合 VAR-TF-IDF 的特征选择方法, 把选出的文本特征的词向量叠加作为文本的特征向量, 这样的特征向量包含了文档的语义语法信息并可以指定维数, 把原来孤立的多维特征浓缩到指定的维数, 解决了文本特征向量的高维和稀疏的问题, 降低了分类器的负担, 又能更好的表示文本特征.

3 实验与结果分析

本实验分两部分, 第一部分证明 VAR-TF-IDF 的特征选择方法相对于常用特征选择方法的优势, 比较他们的分类准确率、召回率. 第二部分证明: (1)相同特征选择算法下用词向量进行特征提取时, 分类效果可以提高; (2)不同特征选择算法用词向量进行特征提取时, VAR-TF-IDF 特征选择算法分类效果仍然是最好的.

3.1 用 VAR-TF-IDF 进行特征选择

分类所用语料是从互联网各个新闻门户和新闻网站所爬取来的新闻. 包括体育 11123 篇、军事 4719 篇、健康 7752 篇、教育 13150 篇、女人 3470 篇、房产 6871 篇、文化 33801 篇、汽车 5468 篇、科技 14129 篇、财经 31424 篇、国际 2433 篇、娱乐 26705 篇、美食 4498 篇、旅游 9586 篇、社会 5468 篇共 15 个新闻分类频道的共 188960 篇新闻. 训练数据和测试数据的划分是 5:1, 训练集和测试集不相交, 对比了 python 的机器学习模块 sklearn 中的各个机器学习算法后, 这里采用逻辑回归进行分类, 模型参数为 C=1, tol=0.1, L1 正则化

方式. 评价方法采用准确率, 召回率和 F1 值.

对于每篇新闻中所有在最终特征集合里面的特征, 按照权重评估函数(7)由高到低排序筛选出的所有特征, 权重最大的前 100 个作为该新闻的特征词, 由此完成了特征选择过程. 对比 VAR-TF-IDF 的特征选择算法与 TF-IDF, 卡方检验(Chi2), 互信息(Mutul_Info)等方法, 查看它们在文本分类中准确率和召回率的区别, 实验结果如表 1.

表 1 不同特征选择方法下文本分类的效果(%)

	Precision	Recall	F1_score
VAR-TF-IDF	81.31	80.12	80.68
TF-IDF	81.35	74.38	77.37
Info_Gain	77.50	70.28	73.08
Chi2	76.49	73.91	75.11
Mutual_Info	79.72	71.82	74.94

可以看出 VAR-TF-IDF 的特征选择方法明显提高了分类的准确率, 召回率, 准确率平均提高了 2.54 个百分点, 召回率平均提高了 7.52 个百分点, F1 值平均提高了 5.56 个百分点. 召回率的提高相对较多, 是因为方差改进后的特征选择在选出正相关特征的基础上, 也倾向于选择与类别负相关的特征, 负相关特征有利于提高分类的查全率. 因此召回率相对于准确率有了更多提高.

3.2 词向量进行特征提取

本文所使用的 Word2vec 的词向量是由 17G 汉语语料训练而成, 其中包括了各大新闻门户网站爬取来的 2015.06 到 2015.12 的汉语新闻, 搜狗语料等开源语料库. Word2vec 词向量的训练采用 CBOW+HS 的训练框架, 上下文长度为 5, 权重衰减因子, 分别训练出 10 维, 50 维, 100 维, 150 维, 200 维, 300 维的词向量.

用 VAR-TF-IDF 已经选出好的特征的情况下, 为了进一步提高分类效果, 将特征词的词向量叠加作为文本的向量, 进行分类. 这里采用和第一个实验相同的数据, 相同的分类器及其参数, 不同的是特征提取过程, 即特征的最后的表达方式, 本实验分别对比了词向量是 10, 50, 100, 150, 200, 300 维的条件下分类准确率、召回率的变化情况.

表 2 不同特征选择方法, 不同词向量下的分类效果(%)

Method	Vecor length	Precision	Recall	F1_score
VAR-TF-IDF	10	76.52	70.84	72.90
	50	83.52	81.77	82.57
	100	84.60	82.84	83.66
	150	84.93	83.30	84.06
	200	84.92	83.72	84.27
	300	84.51	83.60	84.02
TF-IDF	10	77.55	71.93	74.11
	50	81.56	79.67	80.47
	100	82.33	80.63	81.37
	150	82.63	81.04	81.74
	200	82.10	80.94	81.42
	300	83.08	81.58	82.23
Info_Gain	10	73.23	66.37	68.69
	50	78.94	75.97	77.28
	100	80.02	77.03	78.36
	150	80.40	77.50	78.79
	200	80.46	78.05	79.12
	300	80.64	7813	7925
Chi2	10	78.30	74.72	76.18
	50	83.22	81.50	82.25
	100	83.33	81.98	82.56
	150	83.55	82.34	82.87
	200	84.33	82.73	83.45
	300	84.58	83.00	83.72
Mutual_Info	10	65.48	56.32	59.52
	50	76.79	73.02	74.63
	100	78.63	74.92	76.49
	150	78.94	75.70	77.10
	200	78.11	75.17	76.45
	300	79.24	76.17	77.52

实验结果表明: (1) 相同的词向量下, 改进的特征选择方法确实提高了分类的准确率,召回率, F1 值, 准确率平均提高了 3.5 个百分点, 召回率平均提高了 4 个百分点. (2) 采用基于深度学习的词向量进行特征提取, 词向量的维度为 200 时, 分类的准确率, 召回率和 F1 值的提高最多, 但是同 100-300 之间其他维度的词向量相比优势不明显几乎可以忽略. 因此 100-300 之间的维度都可以取得较好的分类效果. 维度小于 100 或者大于 300 时, 分类精度慢慢降低, 维度高于 300 还会导致计算复杂度的增加, 维度为 10 或者更小时分类准确率下降明显. (3) 采用 VAR-TF-IDF 的特征选择方法

和词向量的特征提取方法时, 分类效果最好.

所以基于深度学习词向量的特征提取方法由于其考虑了特征词之间的语义语法关联, 这样的特征向量可以更好的表示文本, 提高文本分类的效果.

4 结语

本文探讨了文本分类中的特征选择和特征提取问题. 针对现有特征选择方法的不足, 本文提出了 VAR-TF-IDF 特征选择的算法, 本文认为类别之间词的概率分布, 文档概率分布差异大的特征有助于分类精确度的提高. 特征提取方面, 为了使文本的特征向量包含特征之间的语义语法关联, 降低向量空间的维度以降低求解的复杂度, 这里提出用改进的 Word2vec 的 CBOW+HS 的训练框架去训练指定维度词向量, 文本特征词的词向量的叠加得到的向量就是文本的特征向量, 实验证明这样的特征向量可以进一步提高文本分类的准确率, 召回率.

参考文献

1 唐焕玲,孙建涛,陆玉昌.文本分类中结合评估函数的 TEF-WA 权值调整技术.计算机研究与发展,2005,42(1): 47–53.

2 陈雨杰.文本分类中的特征选择算法研究[硕士学位论文].哈尔滨:哈尔滨工业大学,2015.

3 赵世奇,张宇,刘挺,陈逸恒,黄永光,李生.基于类别特征域的文本分类特征选择方法.中文信息学报,2005,19(6):21–27.

4 张玉芳,万斌候,熊忠阳.文本分类卷,第中的特征降维方法研究.计算机应用研究,2012,29(7):2541–2543.

5 张玉芳,彭时名,吕佳.基于文本分类 TF-IDF 方法的改进与应用.计算机工程,2006,32(19):76–79.

6 邱云飞,王威,刘大有,邵良彬.基于方差的 CHI 特征选择方法.计算机应用研究,2012,29(4):1304–1306.

7 Zhang M, Zhang Y, Che W. Chinese parsing exploiting Characters. ACL, 2013: 125–134.

8 Sun Y, Lin L, Yang N. Radical-enhanced chinese character embedding. Neutral Information Processing. Spring Internatiional Publishing, 2014: 279–286.

9 Socher R, Perelygin A, Wu JY. Recursive deep models for semantic compositonality over a sentiment treebank. Proc. of Conference on Empirical Methods in Natural Language Processing. 2015. 1631–1642.

10 江大鹏.基于词向量的短文本分类方法及研究[硕士学位论文].杭州:浙江大学,2015.

11 Hu B, Tang B, Chen Q, et al. A novel word embedding learning model using the dissociation between nouns and verbs. Neurocomputing, 2016, 171: 1108–1117.

12 邱云飞,王威,刘大有,邵良杉.基于方差的 CHI 特征选择方法.计算机应用研究,2012,29(4):1304–1306.

13 Wang P, Xu B, Xu J, et al. Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. Neurocomputing, 2016, 174: 806–814.

14 Nalisnick E, Mitra B, Craswell N, et al. Improving document ranking with dual word embeddings. Proc. WWW. International World Wide Web Conferences Steering Committee. 2016.[to appear].

15 Mikolov T, Yih IW, Weig GZ. Linguistic regularities in continuous space word representations. NAAL-HLT, 2013: 746–751.

16 Wei I, Lai K, Liu LH, Xu JZ. How to Generate a good word embedding. Computer Science. Computer and language, 2015.

17 Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space. arXiv Preprint arXiv:1301.3781, 2013.

18 Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems, 2013: 3111–3119.