计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(9):237−241 [doi: 10.15888/j.cnki.csa.008066]

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

基于类信息的 TF-IDF 权重分析与改进①

姚严志, 李建良

(南京理工大学 理学院, 南京 210094)
通讯作者: 李建良, E-mail: ljl6006@njust.edu.cn

摘　要: 经典的 TF-IDF 算法仅考虑了特征词频率和逆文档频率等, 忽略了特征词的类间、类内分布信息. 本文通过 TF-IDF 算法计算特征词在不同规模语料库中的权重, 分析特征词的类信息对权重的影响, 并进一步针对该影响提出一种新的衡量特征词的类间、类内分布信息的方法. 本文通过增加两个新的权值, 类间离散因子和类内离散因子, 将其与经典的 TF-IDF 算法结合, 提出了基于类信息的改进的 TF-IDF-CI 算法. 本文通过朴素贝叶斯模型对改进后的算法的分类性能进行了验证. 实验证明, 改进后的权重算法在测试数据集上的表现, 在准确率、召回率和 *F*1 值上均优于经典的 TF-IDF 算法.

关键词: TF-IDF 算法; 类信息; 权重分析; 文本分类

引用格式: 姚严志,李建良.基于类信息的 TF-IDF 权重分析与改进.计算机系统应用,2021,30(9):237–241. http://www.c-s-a.org.cn/1003-3254/8066.html

Feature Weight Analysis and Improvement of TF-IDF Based on Category Information

YAO Yan-Zhi, LI Jian-Liang

(School of Science, Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract: The classical TF-IDF algorithm only considers the feature term frequency, inverse document frequency, etc. but overlooks the distribution information of feature terms between and inside categories. In this study, we calculate the weights of feature terms through the TF-IDF algorithm in the corpus with different scales and analyze the impact of category information on weights. Based on this, a new method is proposed to measure the distribution information of feature terms between and inside categories. Furthermore, an improved TF-IDF-DI algorithm based on category information is proposed by adding two new weights and discrete factors between and inside categories to the classic TF-IDF algorithm. The Naive Bayes algorithm is used to validate the classification performance of the improved algorithm. Experiments show that the algorithm is superior to the classic TF-IDF algorithm in precision, recall, and *F*1 values.

Key words: TF-IDF algorithm; feature weight; weight analysis; text classification

随着网络的普及, 网络上时刻都在产生大量的文本信息, 为了满足用户面对海量文本时多样化的需求, 对文本信息进行有效的分类就显得至关重要. 在文本分类领域中, 用向量空间模型表示文本的方法应用尤为普遍. 用向量空间模型表示文本, 需经过分词、特征选择、权重计算等步骤, 而权重计算方法的优劣直接影响着分类算法的性能表现. 权重计算的方法多种多样, 常用的包括文档频率、信息增益、互信息、卡方分布、TF-IDF 等[1].

TF-IDF 算法自提出以来, 因其算法相对简单和有较高的准确率及召回率, 一直受到广泛应用[2]. 但该算法的权重计算仅考虑了特征词的词频和逆文档频率等, 仍还有许多可改进的空间. 因此, 很多学者分析 TF-IDF 的缺陷, 对其进行了相应的改进. How 等[2] 提出利

① 收稿时间: 2020-12-04; 修改时间: 2021-01-08; 采用时间: 2021-01-13; csa 在线出版时间: 2021-09-02

用 Category Term Descriptor (CTD) 来改进 TF-IDF, 考虑不同类别的文档数可能存在数量级的差距, 以改善类别数据集偏斜所引起的误差; 徐冬冬等[3] 引入逆类频率因子和类别比率因子用以修正 TF-IDF 权重算法, 得到基于类别描述的 TF-IDF-CD 方法, 叶雪梅等[4] 针对新词识别对分类结果的影响, 提出了基于网络新词的改进文本分类 TF-IDF 算法; 许甜华等[5] 通过引入去中心化词频因子和特征词位置因子以加强特征权重的准确性.

本文使用 TF-IDF 算法计算特征词权重, 对特征词在不同规模文档集中的权重加以比较, 具体分析了特征词的类信息对于权重的影响, 并在此基础上提出一种新的衡量特征词的类间、类内分布信息的改进方法. 改进方法增加两个新的权值, 类间离散因子和类内离散因子, 将其与经典的 TF-IDF 算法结合, 进而提出了改进的 TF-IDF-DI 算法. 改进的权重计算方法有效改善了 TF-IDF 算法对类信息不敏感的问题. 本文通过朴素贝叶斯模型对改进后的算法的分类性能进行验证. 实验证明, 改进后的权重算法在测试数据集上的表现, 在准确率、召回率和 $F1$ 值上均优于经典的 TF-IDF 算法.

1 经典的 TF-IDF 算法及权重分析

1.1 经典的 TF-IDF 算法

TF-IDF 算法作为计算特征项权重的算法, 在文本分类中的应用极为广泛, 其主要思想为: 在某一特定文档中, 某词语的出现频率越高, 且数据集中包含该词语的文档数越少, 说明该词语越是能标志文档内容的属性, 其权重自然也就越大[6–9]. 计算公式如下:

$$\begin{aligned} w(t_j,d_i) &= tf(t_i,d_j)\times idf(t_i,d_j) \\ &= tf(t_i,d_j)\times \lg\left(\frac{N}{n_j}+0.01\right) \end{aligned} \tag{1}$$

其中, $w(t_j,d_i)$表示特征词权重; $idf(t_j,d_i)$表示特征词在文档d_i中的出现频率; N表示文档集中的文档总数; n_j表示文档集中出现特征词t_j的文档数.

在使用时考虑到文档长度不同对权值计算的影响, 我们通常会对公式做归一化处理[10], 得到公式如下:

$$w(t_j,d_i)=\frac{tf(t_j,d_i)\times \lg\left(\frac{N}{n_j}+0.01\right)}{\sqrt{\sum_{k=1}^{n} tf(t_k)^2\times \lg^2\left(\frac{N}{n_k}+0.01\right)}} \tag{2}$$

1.2 TF-IDF 算法的权重分析

传统 TF-IDF 并不能很好的区分类间和类内分布所带来的影响. 类间分布指的是特征词在不同类别间的分布情况, 通常认为集中分布于某个类别的特征词, 相比于在各个类别均匀分布的特征词, 更能体现该类别的内容属性; 类内分布指的是特征词在某类别内的分布情况, 通常认为在某类别内各文档均普遍出现的特征词能够更好的表现该类别的内容属性, 反之对于仅出现于类别内一小部分文档的特征词, 往往特征词只是体现了该小部分文档的内容属性, 我们应适当降低其权重.

我们使用 IMDB 语料库进行实验来说明以上问题. IMDB 语料库收集了 50 000 条来自互联网的严重两极分化的电影评论, 我们从中分别随机抽取 200、500、1000 条评论, 根据式 (2) 计算特征词的 TF-IDF 权重, 并进一步计算特征词在正类评论、负类评论中的平均 TF-IDF 权重. 为保证实验的随机性, 我们重复以上实验多次, 并计算特征词的平均 TF-IDF 权重. 表 1 是部分特征词在不同文档集的权重.

表 1 部分特征词在不同文档集的平均 TF-IDF 权重

文档集容量	特征词	TF-IDF	
		正类评论	负类评论
200	fighting	9.9686×10^{-4}	4.2689×10^{-4}
	awkward	$1.486\,38\times10^{-3}$	9.2993×10^{-4}
	sincere	7.9704×10^{-4}	5.3264×10^{-4}
500	fighting	8.2402×10^{-4}	8.2483×10^{-4}
	awkward	$1.960\,61\times10^{-3}$	8.3979×10^{-4}
	sincere	4.1025×10^{-4}	2.0710×10^{-4}
1000	fighting	$1.163\,80\times10^{-3}$	4.1628×10^{-4}
	awkward	$1.529\,46\times10^{-3}$	$1.070\,24\times10^{-3}$
	sincere	7.0905×10^{-4}	7.9736×10^{-4}

在实验中我们发现大部分特征词在不同的文档集中使用 TF-IDF 算法计算的权重均有较大差别, 能够较好的体现特征词的内容属性, 如表 1 中的特征词“awkward”. 但是我们也发现部分特征词在有些文档集中的 TF-IDF 十分接近, 如特征词“fighting”在样本容量为 500 的文档集和特征词“sincere”在样本容量为 1000 的文档集中, 它们在正类和负类的评价中的 TF-IDF 权重都极为接近. 我们进一步统计分析了此类权重接近的特征词在正类评论和负类评论中的词频和文档频率. 表 2 从不同容量的文档集中选取了部分 TF-IDF

权重接近的特征词, 并分别比较了其在正类评论和负类评论中的词频、文档频率信息.

通过表 2 可以发现部分特征词的 TF-IDF 权重极为接近, 但其在不同类别的词频、文档频率却有着较大的差异. 这说明在该情况下 TF-IDF 算法并不能很好的反映特征词的类间、类内的分布信息, 因此提出一种新的衡量特征词的类间、类内分布信息的方法就显得尤为重要了.

表 2 部分 TF-IDF 权重接近的特征词在正类评论和负类评论中的词频、文档频率

文档集容量	特征词	正类评论			负类评论		
		词频(%)	文档频率(%)	TF-IDF	词频(%)	文档频率(%)	TF-IDF
200	price	4.0403	53.78	$1.968\,67\times10^{-3}$	3.0186	39.07	$1.921\,25\times10^{-3}$
200	forbidden	0.0063	2.13	4.2483×10^{-4}	0.0028	2.91	4.2373×10^{-4}
500	fighting	0.0003	2.05	8.2402×10^{-4}	0.0007	3.29	8.2483×10^{-4}
500	effects	7.7731	62.30	4.2483×10^{-4}	6.5829	36.93	4.2711×10^{-4}
1000	sincere	2.1075	56.33	7.0905×10^{-4}	0.0775	21.09	7.9736×10^{-4}
1000	waste	0.9701	17.49	5.4158×10^{-4}	1.7943	34.24	5.5239×10^{-4}

2 改进的 TF-IDF 算法

文献 [11] 提出了改进的 TF-IDF-DI 方法通过变异系数, 即特征词词频在类间、类内的分布标准差与均值之比来描述其类间、类内离散程度, 但仍有其缺陷: 当特征词在各类别中的平均出现频率或特征词在某类别中的各文档的平均出现频率较小, 以至趋近于 0 时, 即使微小的扰动也会导致也会对系数产生巨大的影响, 不利于准确描述特征词的类信息.

本文提出一种新的类间、类内离散程度的描述方法, 进而提出了改进的 TF-IDF-CI 算法. 我们引入特征词的类间离散度因子CI_{ac}和类内离散度因子CI_{ic}.CI_{ac}通过特征词在不同类别文档集的词频的分布标准差来描述特征词的类间分布信息; CI_{ic}通过特征词在类别c_k内的词频与类别c_k内实际包含该特征词的文档的词频之差描述特征词的类内分布信息. 通过类信息的引入, 改进的算法加强了区分特征词类别分布信息的能力. 下面分别给出衡量类间离散度CI_{ac}和类内离散度CI_{ic}的方法:

$$CI_{ac}(t_j)=2\arctan(S(t_j))/\pi \tag{3}$$

$$CI_{ic}(t_j,c_k)=2\arctan(s(t_j,c_k))/\pi \tag{4}$$

其中, $S(t_j)$指特征词t_j在各类别之间的词频的分布标准差; $s(t_j,c_k)$指特征词t_j在类别c_k的词频与类别c_k中实际包含该特征词的文档的词频之差, 计算方法如下:

$$S(t_j)=\sqrt{\left[\sum_{k=1}^{|C|}\left(TF(t_j,c_k)-\overline{TF(t_j)}\right)^2\right]\Big/(|C|-1)} \tag{5}$$

$$s(t_j,c_k)=TF(t_j,c_k)\cdot\frac{N(c_k)}{n(t_j,c_k)}-TF(t_j,c_k) \tag{6}$$

其中, $TF(t_j,c_k)$表示特征词t_j在类别c_k中的出现频率; $\overline{TF(t_j)}$表示特征词t_j在各类别中的平均出现频率; $N(c_k)$表示类别c_k中的文档数; $n(t_j,c_k)$表示类别c_k中包含特征词t_j的文档数; C为文档集的总类别数.

在式 (3)–式 (6) 中, 我们给出了类间离散因子CI_{ac}和类内离散度因子CI_{ic}的计算方法. 易发现特征词t_j在不同类别中的分布标准差越大时, 特征词t_j越能体现不同类别的内容属性, 分类能力越强; 特征词t_j在类别c_k中的词频与特征词t_j在类别c_k中实际包含该特征词的文档中的词频, 两者之差越大时, 说明特征词t_j是更突出表现了类别c_k中部分文档的内容属性而不是类别c_k的整体的内容属性, 分类能力越弱. 可见特征词的分类能力与CI_{ac}成正比, 与CI_{ic}成反比. 基于此我们得到了改进的 TF-IDF-CI 算法:

$$W(t_j,d_i,c_k)=w(t_j,d_i)\times\frac{1-CI_{ic}(t_j,c_k)}{1-CI_{ac}(t_j)} \tag{7}$$

其中, $W(t_j,d_i,c_k)$是改进的特征权重; $w(t_j,d_i)$为式 (2) 中计算所得的特征词t_j在文档 d_i中的权重.

同样采用表 1 中所使用的文档集进行实验, 表 3 给出部分特征词根据改进的 TF-IDF-CI 算法在不同文档集中计算所得的特征权重, 并与 TF-IDF 算法计算的权重进行对比.

通过表 3 的对比容易发现, 改进的 TF-IDF-CI 算法有效改善了 TF-IDF 算法并能很好的反映特征词类间、类内的分布信息的问题. 如特征词“fighting”在样本容量为 500 的文档集和特征词“sincere”在样本容量为 1000 的文档集中, 使用 TF-IDF 算法的计算的特征权重极为接近, 但使用 TF-IDF-CI 算法则得到

了有效的改善. 同时, 通过实验也可发现如“awkward”等使用 TF-IDF 算法可以很好区分的特征词, 在使用 TF-IDF-CI 算法计算特征权重时亦不会有很大的偏差.

表 3 部分特征词在不同文档集的 TF-IDF 权重与 TF-IDF-CI 权重对比

文档集容量	特征词	TF-IDF		TF-IDF-CI	
		正类评论	负类评论	正类评论	负类评论
200	fighting	9.9686×10^{-4}	4.2689×10^{-4}	8.1648×10^{-4}	4.1246×10^{-4}
	awkward	$1.486\,38\times10^{-3}$	9.2993×10^{-4}	$1.414\,76\times10^{-3}$	8.6429×10^{-4}
	sincere	7.9704×10^{-4}	5.3264×10^{-4}	9.7623×10^{-4}	3.1934×10^{-4}
500	fighting	8.2402×10^{-4}	8.2483×10^{-4}	$1.154\,78\times10^{-3}$	7.8298×10^{-4}
	awkward	$1.960\,61\times10^{-3}$	8.3979×10^{-4}	$1.719\,86\times10^{-3}$	7.1735×10^{-4}
	sincere	4.1025×10^{-4}	2.0710×10^{-4}	6.2148×10^{-4}	2.5171×10^{-4}
1000	fighting	$1.163\,80\times10^{-3}$	4.1628×10^{-4}	$1.346\,82\times10^{-3}$	2.0971×10^{-4}
	awkward	$1.529\,46\times10^{-3}$	$1.070\,24\times10^{-3}$	$1.773\,10\times10^{-3}$	7.9617×10^{-4}
	sincere	7.0905×10^{-4}	7.9736×10^{-4}	$1.107\,16\times10^{-3}$	5.8723×10^{-4}

3 实验分析

实验使用的语料库是搜狗新闻数据语料库, 该语料库包含来自搜狐新闻的健康、体育、社会、娱乐等 18 个频道的新闻数据. 实验选取了健康、教育、军事、汽车、体育 5 类共 5000 篇文档作为训练样本, 另选取 500 篇文档作为测试样本.

分词使用的是 Hanlp 的 StandardTokenizer 分词器. 同时还对分词后的数据集进行去体停用词的处理, 将常用的停用词 (的, 并不, 而且等) 进行过滤. 为验证改进的 TF-IDF-CI 算法对分类性能的影响, 实验分别采用经典的 TF-IDF 算法、TF-IDF-DI 算法、改进的 TF-IDF-CI 算法计算特征词的权重, 并使用朴素贝叶斯算法进行文本分类, 评估指标使用准确率 (Precision, P)、召回率 (Recall, R)、$F1$ 值 3 个指标[12]. 分类器在测试集上的分类性能分别如表 4 所示.

通过实验结果, 可以发现使用改进的 TF-IDF-CI 算法对特征词权重进行计算, 并使用朴素贝叶斯算法对文本进行分类, 准确率、召回率和 $F1$ 值都相比于经典的 TF-IDF 算法有了一定的提升, 其中类别“健康”的提升最为明显, $F1$ 值较 TF-IDF 提升了约 6.42%, 较 TF-IDF-DI 提升了约 3.23%. 这说明改进的 TF-IDF-CI 算法相比于 TF-IDF 算法, 较好的考虑了特征词的类间、类内的分布信息, 能很好的分辨出集中分布于某类别且在该类别内相对均匀出现的特征词, 从而达到了提升分类性能的效果.

表 4 不同权重算法的分类性能对比 (%)

类别	TF-IDF			TF-IDF-DI			TF-IDF-CI		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
健康	82.12	81.27	81.69	84.20	84.46	84.32	87.71	87.67	87.69
教育	81.66	82.76	82.21	83.26	85.56	84.39	87.45	86.69	87.07
军事	85.56	85.44	85.5	87.81	87.04	87.42	91.81	90.38	91.09
汽车	81.27	80.97	81.11	83.56	81.73	85.26	85.67	85.72	85.69
体育	76.71	80.35	78.48	78.42	81.99	82.50	77.01	80.21	78.57

4 总结与反思

本文以特征词权重的计算方法为研究对象, 总结了现有的一些方法, 并着眼于使用相对广泛的经典的 TF-IDF 算法, 对国内外研究者在 TF-IDF 算法的研究成果进行了介绍. 本文对 TF-IDF 算法在不同的文档集中的表现做了具体的分析对比, 针对 TF-IDF 算法未能很好区分特征词类间、类内分布的问题, 做了详细的研究. 基于此本文提出了一种新的衡量特征词类间、类内分布信息的方法, 提出了基于类信息的改进的 TF-IDF-CI 算法. 最后通过朴素贝叶斯模型对改进后的算法的分类性能进行验证. 实验发现, 改进的 TF-IDFCI 算法不论在准确率、召回率、$F1$ 值上, 均优于经典的 TF-IDF 算法, 由此证实了改进算法的有效性.

当然本文仍有不足之处: 首先本文的实验均在均

衡的数据集上进行实验, 改进的 TF-IDF-CI 算法在数据集偏斜时的表现还需要进一步实验, 以验证其性能[2]; 同时 TF-IDF-CI 算法仍还有改进空间, 如将特征词在文本内的分布信息, 即其位置信息进一步纳入特征权重的考虑范畴, 这也是笔者今后要研究的内容.

参考文献

1 李鹏鹏, 范会敏. 文本分类中特征权重算法改进研究. 计算机与现代化, 2018, 2(2): 66–70. [doi: 10.3969/j.issn.1006-2475.2018.02.014]

2 How BC, Narayanan K. An empirical study of feature selection for text categorization based on term weightage. Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence. Beijing, China. 2004. 599–602.

3 徐冬冬, 吴韶波. 一种基于类别描述的 TF-IDF 特征选择方法的改进. 现代图书情报技术, 2015, 31(3): 39–48.

4 叶雪梅, 毛雪岷, 夏锦春, 等. 文本分类 TF-IDF 算法的改进研究. 计算机工程与应用, 2019, 55(2): 104–109, 161. [doi: 10.3778/j.issn.1002-8331.1805-0071]

5 许甜华, 吴明礼. 一种基于 TF-IDF 的朴素贝叶斯算法改进. 计算机技术与发展, 2020, 30(2): 75–79. [doi: 10.3969/j.issn.1673-629X.2020.02.016]

6 段国仑, 谢钧, 郭蕾蕾, 王晓莹. Web 文档分类中 TFIDF 特征选择算法的改进. 计算机技术与发展, 2019, 29(5): 49–53. [doi: 10.3969/j.issn.1673-629X.2019.05.010]

7 张玉芳, 彭时名, 吕佳. 基于文本分类 TFIDF 方法的改进与应用. 计算机工程, 2006, 32(19): 76–78. [doi: 10.3969/j.issn.1000-3428.2006.19.028]

8 黄磊, 伍雁鹏, 朱群峰. 关键词自动提取方法的研究与改进. 计算机科学, 2014, 41(6): 204–207. [doi: 10.11896/j.issn.1002-137X.2014.06.040]

9 周炎涛, 唐剑波, 王家琴. 基于信息熵的改进 TFIDF 特征选择算法. 计算机工程与应用, 2007, 43(35): 156–158, 171. [doi: 10.3321/j.issn:1002-8331.2007.35.047]

10 Salton G, Buckley B. Term-weighting approaches in automatic text retrieval. Information Processing & Management, 1998, 24(5): 513–523.

11 徐凤亚, 罗振声. 文本自动分类中特征权重算法的改进研究. 计算机工程与应用, 2005, 41(1): 181–184, 220. [doi: 10.3321/j.issn:1002-8331.2005.01.056]

12 黄勇, 罗文辉, 张瑞舒. 改进朴素贝叶斯算法在文本分类中的应用. 科技创新与应用, 2019, 5(5): 24, 27.