文本分类中基于类别概念的特征选择方法①

A New Feature Selection Method Based on Class-Concept in Text Categorization

杨奋强 刘玉贵 (中国科学院 研究生院 信息科学与工程学院 北京 100049)

摘 要: 传统的 TFIDF 公式常被用于信息检索各种计算特征项权重的场合，但在文本分类任务下，TFIDF 忽略了特征项的类别信息，且较易产生一些不合理的低频高权特征，一定程度上影响了最终分类的准确性。本文提出一种基于类别概念的 TFCW 特征选择方法，该方法避免了 TFIDF 的上述缺陷。实验表明该方法用于文本分类中优于目前常见的 TFIDF 改进算法。

关键词: 文本分类 特征选择 TFCW 类别

1 前言

文本分类是信息化时代人们组织和管理信息资料、并从中快速准确地获取知识的有效手段。在互联网信息量不断急速膨胀的情况下，文本自动分类技术已成为信息检索领域非常活跃的一个研究方向。从 90 年代至今，各种机器学习及统计方法不断应用于文本分类中，其中以 KNN 和 SVM 为代表，分类效果得到了较大幅度的提高。文本分类的最终效果，受到分词、特征选择及分类算法等因素的限制。TFIDF 因其简单有效，成为目前最常用的一种特征选择方法。之前很多人从词的角度，对 TFIDF 公式做了不同程度的改进，但始终没有摆脱 IDF 在分类任务下的局限性：容易产生不合理的低频高权词；TFIDF 本身没有考虑特征项的类别因素，尽管有一些改进算法考虑到类别因素做了补充改进，但同时也给计算带来了更多的复杂性。本文综合词与类别的因素，从类别词的权重计算的角度提出一种专门适用于文本分类任务的特征选择方法 TFCW，这种方法相对之前的一些 TFIDF 改进方法更为简洁有效。最后通过实验证实了 TFCW 方法的有效性。

2 TFIDF算法及其改进算法

2.1 TFIDF 算法

目前特征选择较常用的方法主要有 TFIDF、IG (Information Gain)、MI(Mutual Information)、CHI、ECE(Expected Cross Entropy)等[1]。在英文文本分类中，文献[2]通过实验客观地比较了各种特征选择方法，得出 IG、CHI 方法表现最好，TFIDF 稍次的结论[2]。而在中文文本分类中，文献[3]则发现 TFIDF 特征选择效果要好于 IG、CHI 方法[3]。又因为 TFIDF 的简洁易用性，因此 TFIDF 公式成为目前文本分类中最常用的一种特征选择方法。

TFIDF 公式主要从文档特征项的频度及稀疏性两个方面来考虑的：

TF 指的是特征项的出现次数(Term Frequency)。特征项可以是字、词或者是短语。如果一个特征项在当前文档中出现的频率较高，TF 方法则认为该词更能代表该文档的特征。

IDF 是反文档频度，如果一个特征项越普遍地在各文档中出现，则 IDF 认为该词的重要性越小，其公式表示为[4]：

$$IDF(t) = \log(N / n(t))$$

其中，N 为所有的文档数。表示在所有训练集文档中，含有特征项 t 的文档个数。

目前，常用的 TFIDF 归一化公式表示为[1]：

$$TFIDF(t,d) = \frac{\log(tf(t,d)+1.0)\times\log(N/n(t)+L)}{\sqrt{\sum_t[\log(tf(t,d)+1.0)\times\log(N/n(t)+L)]^2}}$$

① 基金项目:国家自然科学基金(69983007)
收稿时间:2009-01-13

其中，$\log(tf(t,d)+1.0)$ 即 TF 部分，$tf(t,d)$ 表示特征项 t 在文档 d 中出现的次数；$\log(N/n(t)+L)$ 是 IDF 部分，L 根据实验结果确定，其他参数具体含义同上。

2.2 TFIDF 存在的问题及其改进算法

IDF 基于以下假设：如果某个特征词越稀少，那么其重要性越大；反之越小。而根据训练语料库的不同，一些偏僻词随机出现在各个类中，如一些地名，人名，生僻字等，这类词比如“乌拉圭”、“邦臣”、“豔”等，往往对分类没有太大作用，但 IDF 赋予其很高的权重，这样一定程度上降低了最终分类的精度。另一方面，某些在个别类中出现频度较高，对分类具有较好的区分度，如“司令员”、“教育”、“历史”等，而 IDF 却认为这些词是相对不重要的。总之，IDF 仅仅是从特征项的生僻程度的角度去考虑其重要性，而没有考虑该词是否对分类具有更好的区分度。

考虑到词在各类中的分布，不少研究者对 TFIDF 公式加入了离散度的因素[5, 6]。表示特征项类间分布的离散度常用方差来计算。公式如下：

$$DI(t)=\frac{\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(tf_i(t)-\overline{tf}(t))^2}}{\overline{tf}(t)}$$

其中，$tf_i(t)$ 表示特征项 t 在第 i 类中出现的次数。

$$\overline{tf}(t)=\frac{1}{n}\sum_{i=1}^{n}tf_i(t)$$

$DI(t)$ 越大，表明特征项 t 在类间分布不均匀，即在某一个或者几个类中出现的频度较高，这类词对分类的区分度较高。更多地还可以考虑特征项在类内的分布特点：特征项类内分布越均匀，则该特征项的区分能力越高，同样也用方差来计算其分布特点。因此，加入离散度因素后的 TFIDFDI 公式为：

$$TFIDFDI(t,d)=TFIDF(t,d)\times DI_{\text{inter}}(t)\times(1\text{-}DI_{\text{inner}}(t))$$

其中，$DI_{\text{inter}}(t)$ 为类间离散度，$DI_{\text{inner}}(t)$ 为类内离散度。

另外，由于 TFIDFDI 方法容易产生一些低频高权特征(LFHW)，这些低频高权特征项需要另外挖掘出来，形成 VSM 后再单独处理。

3　TFCW算法

传统的 TFIDF 方法简单易用，但没有考虑到分类任务中的特殊性。改进后的 TFIDFDI 算法一定程度上提高了特征项权重计算的效果，但同时也带来了一定的复杂性。除此之外，TFIDFDI 算法没有根除 IDF 算法带来的缺陷，对于 TFIDF 产生的一些不合理的低频高权特征项，其采取了“事后处理”的补救措施，这些低频高权词处理又要通过和具体的实验相关的阈值来选择。除此之外，DI 模型同 TF 之间也存在一定的耦合性：若某个特征项在各类之间 TF 分布不均匀，则 TF 因素会将最终分类结果导向含有该特征项 TF 最大的类，而 DI 模型又一次重复了这个效果。

为了克服这些缺点，综合词频与类别的因素，从类别词的权重计算角度，本文提出一种专门适用于文本分类任务的特征选择方法 TFCW。该方法从另一个角度出发，不用考虑离散度，而是基于以下一个事实：如果一个特征项在某个类类内频度越高，而在类外频度越低，那么认为这个特征项更能体现出该类的特点。这里用类别词的权重(CW：Class Weigh)来描述这个词对类的体现程度：若一个特征项在某个类下的权重越高，则该特征项更倾向于该类。为了避免 CW 同 TF 之间的耦合性，CW 从 DF 的角度来计算类别词权重，公式如下：

$$CW(t,c_i)=\log\left(\frac{\alpha\times inner\_df(t,c_i)}{outer\_df(t,c_i)+L}+1\right)$$

其中，$inner\_df(t,c_i)$ 表示特征项 t 在类别类内的文档频度，$outer\_df(t,c_i)$ 表示特征项 t 在类别 c_i 以外的文档频度。α 为调节因子，它与类别数有关，即 $\alpha\propto(class\_num-1)$。L>0 为低频词惩罚因子，如果一个特征项只在个别文档中出现，那么这个特征项 CW 值越低，L 越大，对低频词的惩罚度越大。

在 CW 公式中，一方面体现了特征项的类间分布特点：如若 $inner\_df(t,c_i)$ 越大，$outer\_df(t,c_i)$ 越小，则 $CW(t,c_i)$ 越大，其结果更倾向于 c_i 类，即该特征项的类间分布越不均匀，类别表决能力越强。另外一方面，CW 也描述了特征词的类内分布，如若有特征项 t_1 和特征项 t_2 满足 $inner\_df(t_1,c_i)>inner\_df(t_2,c_i)$，则有 $CW(t_1,c_i)>CW(t_2,c_i)$，表明特征项在类内文档分布越均匀，其对类的表决能力越强，t_1 更能将结果导向 c_i 类。

根据上面所述，TFCW 公式综合 TF 与 CW 两个方面的因素考虑，最终训练样本特征项的 TFCW 归一化公式为：

$$TFCW_{train}(t,d,c_i)=\frac{\log(tf(t,d)+1.0)\times CW(t,c_i)}{\sqrt{\sum_t[\log(tf(t,d)+1.0)\times CW(t,c_i)]^2}}$$

待测样本特征项的 CW 值取其在所有类别中的最

大值，其公式表述为：

$$TFCW_{test}(t,d)=\frac{\log(tf(t,d)+1.0)\times \max CW(t)}{\sqrt{\sum_{t}[\log(tf(t,d)+1.0)\times \max CW(t)]^2}}$$

其中：$\max CW(t)=\max(CW(t,c_i)|_{i=1}^{n})$。

采用余弦距离计算文档之间的相似度时，TFCW 特征项匹配公式为：

$$\cos(D_{test},D_{train})=\sum_{j=1}^{m}TFCW_{test}(t_j,D_{test},c_i)\times TFCW_{train}(t_j,D_{train},c_i)$$

其中：$D_{train}\in c_i$，$t_j|_{j=1}^{m}\in D_{test}\cap D_{train}$，$D_{train}$ 为训练集文档，D_{test} 为待测文档。

传统的特征选择算法中，每个特征项的权重是固定不变的。每一个特征项在分类算法未执行前保持“中立”状态，在分类算法结束后才能获取最终的分类结果。特征匹配是一个被动的过程。而在 TFCW 算法中，每个特征项的权重因类别而异，这个权重代表该特征项对这个类的贡献度。如“学位办”在教育类的权重较高，则“学位办”对教育类的贡献度较高；而在医疗类中较低，说明其对医疗类的贡献度低。分类过程中，每一个训练集中的特征向量对测试样本有一个“拉”的作用，如果测试样本的某个特征项更偏向于哪个类，则该特征项将测试样本“拉”向某个类。最终分类的结果，看各种“拉”力的“合力”。整个分类的特征匹配过程是一个主动的过程。

4 实验数据及分析

实验语料由搜狗实验室提供的中文文本分类语料库及复旦大学语料库整理而成。搜狗语料库规模较大，但文档属类较乱；复旦大学语料库经过人工整理，属类情况较好，但是选取文档的质量不高，且规模较小。整理后的语料库分为教育、汽车、医疗、历史、旅游、军事、职业、信息、体育、经济等十个类约 5000 篇文档。整个语料库又分为训练集和测试集两部分。训练集中每个类 200 篇文档，一共 2000 篇文档，形成一个均匀语料库。测试集每类约 300 篇文档，一共约 3000 篇文档。

在实验中，所有文档经过 ICTCLAS 工具分词，采用目前最常用的空间向量模型 VSM(Vector Space Model)来表示一个文档[7]，并对每一个特征项标注了词性。特征选择的过程中，根据汉语的特点，我们只保留了名词、动词、形容词、字符串、时间词等实体含义较清晰、类别区分能力较强的词，如“学校”、“聘任”、“美轮美奂”、“Linux”、“古代”等。而像一些助词，介词，叹词等意义不大的词，都被过滤掉，这样就避免了采用过滤词表方式，人工收集禁用词的过程。

我们使用余弦距离计算文档之间的相似度。设测试样本的 VSM 表示为 $\overrightarrow{D_{test}}(W_1,W_2,...,W_n)$，训练样本的 VSM 表示为：$\overrightarrow{D_{train}}(w_1,w_2,...,w_n)$，特征项经过归一化处理后，余弦距离公式表示为：

$$\cos(\overrightarrow{D_{test}},\overrightarrow{D_{train}})=\overrightarrow{D_{test}}\bullet\overrightarrow{D_{train}}=\sum_{i=1}^{n}W_iw_i$$

KNN 和 SVM 是目前文本分类中效果最好的两类分类器[7,8]。本实验采用 KNN 分类器。公式如下：

$$score(c_i|\overrightarrow{D_{test}})=\sum_{d_j\in c_i,d_j\in KNN}sim(D_{test},d_j)$$

sim 相似度公式采用上面所述的余弦距离公式，根据训练集的规模及类别数，实验中 K 取 15。最终文本将被分到 score 计算结果最大的类中。

最终的评估指标采用最常用的 F1 评估方法[1, 6]，即

查准率 P(Precision)=分类正确的文本数 / 实际分类文本数

查全率 R(Recall)=分类正确的文本数 / 类内应有文本数

兼顾查全率及查准率，F_1 定义为：

$$F_1=\frac{2PR}{P+R}$$

本实验综合比较了传统 TFIDF、加入离散度模型改进后的 TFIDFDI 方法、TFCW 三种方法，分类统计结果如表 1 所示。

从图 1 可以看出，整体分类效果上，TFCW>TFIDFDI>TFIDF。对于一些特征较为明显的类，如体育，军事等，这些类中文档的特征项分布较为集中，因此分类的效果最好。而对于一些类，因为类和类之间存在某些关联，这些类的分类效果较低。如教育类和就业类，难以区分一些类似大学生就业等主题的文档；而对于旅游和历史类，一些描述参观历史博物馆、游览名胜古迹的文档往往容易产生误分，这些类与类之间关联性一定程度上影响了分类结果。总的来说，一个类同其他类的关联性越小，独立性越强，其分类的准确度越高。

表1　三种特征选择方法在KNN分类器下的结果比较

	TFIDF			TFIDFDI			TFCW		
	R(%)	P(%)	F1(%)	R(%)	P(%)	F1(%)	R(%)	P(%)	F1(%)
教育	81.96	81.74	81.47	82.28	86.96	84.55	91.61	91.30	91.46
经济	81.15	81.43	81.29	87.02	88.50	87.76	90.08	90.50	89.26
农业	79.87	78.55	79.20	77.67	72.54	75.02	86.56	85.41	85.98
军事	87.51	86.04	86.77	90.33	92.15	91.23	92.89	94.77	93.82
历史	85.16	82.01	82.58	85.27	82.91	84.07	87.98	92.80	90.30
旅游	75.86	75.36	75.61	82.03	84.71	83.34	86.41	88.10	87.25
汽车	90.86	87.04	88.84	89.24	93.09	91.12	93.81	95.68	94.74
体育	97.30	95.68	96.48	96.32	95.68	96.00	98.48	99.14	98.81
信息	80.21	83.35	81.75	79.78	77.79	78.77	92.27	92.60	92.43
医疗	78.68	83.42	81.51	88.11	93.89	91.44	91.89	91.27	91.58

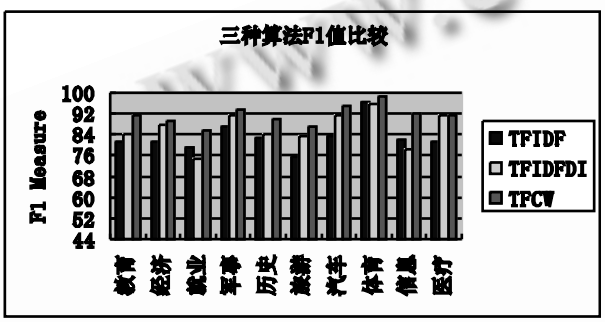

图 1　三种算法比较图

TFCW 模型更多地考虑了分类任务下类别的因素，因此相比 TFIDF 等方法来说，不适用于非分类任务下的权值计算，如文本聚类，搜索引擎系统中检索词权重计算等，有其一定的局限性。除此之外，对训练集有以下要求：计算 CW 值时须保证训练集为均匀语料库，即每个类中的文档数基本相同，这样使得 TFCW 计算结果不会倚重于任何类。一般情况下，为了提高 CW 计算的准确性，可以先选用一个较大规模的均匀语料库，使得每一个特征项的分布具有更高的稳定性，离线计算出所有特征项的 CW 值，然后可以在分类时适当缩减语料库，以提高分类算法的效率，特别是对 KNN 这种时间复杂度和语料库规模相关的算法来说。当然，语料库的规模越大，相对来说分类的准确度越高。

5　结语

本文综合比较传统的 TFIDF，基于离散度模型的 TFIDFDI 方法，以及本文提出的 TFCW 方法，最后发现 TFCW 算法在文本分类任务下，特征选择的效果相对较好。传统的特征选择算法，包括 CHI、IG、ECE 等，往往每个特征项相对于各个类别来说，都保持“中立性”，即在各个类中的权重都是相同的，不会“偏向”某个类。在这些特征选择方法中，特征项的权值大小体现了自身的重要性，最终分类依赖于分类器的结果。而 TFCW 方法从类别的角度出发，计算出的特征项权重因类而异，一个特征项在某个类中权值越大，则该特征项将分类结果更“拉”向该类，每个特征项都有其不同的类别色彩。TFCW 方法模型简洁，最终的实验结果表明该做法相对传统的 TFIDF 等方法是更为行之有效的。

参考文献

1 谭松波.高性能文本分类算法研究[博士学位论文].北京:中国科学院计算技术研究所, 2006.

2 Yang Y, Pedersen JO. A comparative study on feature selection in text categorization. Proc. of ICML-97, 14th International Conf. on Machine Learning, 1997.

3 代六岭,黄河燕,陈肇雄.中文文本分类中特征抽取方法的比较研究.中文信息学报, 2004,18(1):26－32.

4 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.

5 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,(1):181－184.

6 刘海峰,王元元,张学仁.文本分类中一种改进的特征选择方法.情报学报, 2007,25(10):1534－1537.

7 Manning CD, Raghavan P, Schütze H. Introduction to Information Retrieval. Cambridge University Press, 2008:234－318.

8 Yang YM, Zhang J, Kisiel B. A scalability analysis of classifiers in text categorization. Proc. of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. 2003.