ISSN 1673-9418 CODEN JKYTA8
Journal of Frontiers of Computer Science and Technology
1673-9418/2016/10(09)-1299-11
doi: 10.3778/j.issn.1673-9418.1509018

E-mail: fcst@vip.163.com
http://www.ceaj.org
Tel: +86-10-89056056

文本分类中基于熵的词权重计算方法研究*

陈科文+,张祖平,龙 军
中南大学 信息科学与工程学院,长沙 410083

Research on Entropy-Based Term Weighting Methods in Text Categorization*

CHEN Kewen+, ZHANG Zuping, LONG Jun
School of Information Science and Engineering, Central South University, Changsha 410083, China
+ Corresponding author: E-mail: kewencsu@csu.edu.cn

CHEN Kewen, ZHANG Zuping, LONG Jun. Research on entropy-based term weighting methods in text categorization. Journal of Frontiers of Computer Science and Technology, 2016, 10(9): 1299-1309.

Abstract: As the volume of textual data has become very large and is still increasing rapidly, automatic text categorization (TC) is becoming more and more important. Term weighting or feature weight calculation is one of the hot research topics in TC to improve the classification accuracy. It is found that entropy-based weighting (EW) methods are usually more effective than others. However, there are still some problems with the existing EW methods, e.g., they may perform worse than the traditional TF-IDF (term frequency & inverse document frequency), for TC on some text corpora. So this paper proposes a new term weighting scheme called LTF-ECDP, which combines logarithmic term frequency and entropy-based class distinguishing power as a new weighting factor. In order to test LTP-ECDP and compare it with other weighting methods, a considerable number of TC experiments using support vector machine (SVM) have been done on three popular benchmark datasets including a Chinese corpus, TanCorp, and two English corpora such as WebKB and 20 Newsgroups. The experimental results show that LTF-ECDP outperforms the other five entropy-based weighting methods and two famous methods such as TF-IDF and TF-RF (term frequency & relevance frequency). Compared with the other term weighting methods, LTF-ECDP can further improve the accuracy of TC while keeping good performance on different datasets consistently.

Key words: term weighting; entropy-based weighting; text categorization; class distinguishing power

* The National Natural Science Foundation of China under Grant No. 61379109 (国家自然科学基金); the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant No. 20120162110077 (高等学校博士学科点专项科研基金).
Received 2015-07, Accepted 2015-09.
CNKI网络优先出版: 2015-10-13, http://www.cnki.net/kcms/detail/11.5602.TP.20151013.1655.006.html

摘 要:随着文本数据量变得很大且仍在迅猛增加,自动文本分类变得越来越重要。为了提高分类准确率,作为文本特征的词的权重计算方法是文本分类领域的研究热点之一。研究发现,基于信息熵的权重计算方法(熵加权)相对于其他方法更有效,但现有方法仍然存在问题,比如在某些语料库上相比TF-IDF(term frequency & inverse document frequency),它们可能表现较差。于是将对数词频与一个新的基于熵的类别区分力度量因子相结合,提出了LTF-ECDP(logarithmic term frequency & entropy-based class distinguishing power)方法。通过在TanCorp、WebKB和20 Newsgroups语料库上使用支持向量机(support vector machine,SVM)进行一系列文本分类实验,验证和比较了8种词权重计算方法的性能。实验结果表明,LTF-ECDP方法比其他熵加权方法和TF-IDF、TF-RF(term frequency & relevance frequency)等著名方法更优越,不仅提高了文本分类准确率,而且在不同数据集上的性能更加稳定。

关键词:特征词权重;熵加权;文本分类;类别区分力

文献标志码:A **中图分类号:**TP391

1 引言

随着计算机应用的普及和互联网规模的不断发展,文本数据量变得非常庞大且仍在迅猛增加,比如每天都有大量的以文本内容为主的电子文献、网页、消息和邮件在不断地产生。因此,作为文本组织与挖掘的基本技术手段之一,自动文本分类(text categorization,TC)变得越来越重要。为了进一步提高文本分类的性能,研究人员主要从两个方面开展研究:一是改善分类算法(或学习模型);二是改善文本数据表示模型。众所周知,在文本分类领域,通常采用向量空间模型(vector space model,VSM)来表示文本,就是在分类之前把每个文本文档都表示成由一定数量的特征词的权重值所组成的向量。这种表示法涉及到特征词的选择和权重计算两方面。其中特征选择的主要目的是降低文本特征维度,以提高分类速度,同时又保持较高准确率。特征选择必须考虑文本中不同词条的重要性,往往又依赖于权重计算。而特征词的权重计算是否合理则直接影响到文本分类的准确率。因此,特征词权重计算方法成为文本分类领域的研究热点之一。

特征词权重计算(或权重分配)可简称为词加权(term weighting),在后面的叙述中,这几个术语可以互换。众所周知,最常用的文本特征词权重计算方法是TF-IDF方法[1],即根据词频与反文档频率(term frequency & inverse document frequency)来计算特征词的权重。这种方法起源于信息检索领域,并在文本分类和聚类领域也得到了广泛应用。实际上,TF-IDF方法在文本分类领域并不是最有效的,因为它在计算特征词的权重时没有考虑文本的类别。于是,研究人员一直在努力改进TF-IDF,并提出了一些新的权重计算方法。其中很多方法都有一个共同特点,就是利用已知的文本类别信息,因此这些方法统称为有监督词加权(supervised term weighting,STW)[2]。很多STW方法只利用了特征词在正反两类文本上的分布[2-3],也有一些方法考虑了特征词在多个类别上的分布,比如基于信息熵的权重计算方法(简称为熵加权)[4-8]。尽管某些方法已在特定数据集上的文本分类实验中被证明是有效的,但是至今没有人对它们在不同数据集上的性能作进一步的验证并与更多的方法比较。本文对各种特征词权重计算方法进行了系统的研究,发现基于熵的权重计算方法相对而言一般更加有效,但是现有研究工作仍然存在一些问题或不足,于是提出了一种新的熵加权方法,并通过在不同数据集上的大量实验来比较它与其他多种典型的权重计算方法的性能,实验结果充分证明了它的优越性。

本文组织结构如下:第2章分析几种典型的特征词权重计算方法及其局限性;第3章介绍新的熵加权方法;第4章详细介绍一系列文本分类实验,包括实验数据集的选择及其预处理、实验步骤和具体方法,以及最终的实验结果,并对结果进行了分析和讨论;第5章总结全文。

2 相关研究工作的分析

下面将介绍几种典型的特征词权重计算方法，以便于比较。

2.1 传统的TF-IDF方法

最流行的特征词权重计算方法就是传统的TF-IDF。根据TF-IDF方法，一个特征词 t_k 在某个文档中的权重 $w(t_k)$ 不仅取决于它在该文档中出现的次数，即词频（term frequency，TF），表示为 tf_k，而且还取决于整个语料库中包含它的文档数目，即文档频率（document frequency，DF），表示为 df_k。尽管研究人员提出了TF-IDF的多个变种，但通常使用式（1）表示的标准形式[1,9]。

$$w(t_k)=tf_k\times \mathrm{lb}\left(\frac{N}{df_k}\right) \tag{1}$$

其中，N 表示语料库中的总文档数。因为局部因子 tf_k 受文档长度的影响，所以通常还要采用所谓的“余弦归一化（cosine normalization）”方法[9]对同一文档中所有特征词 $t_i(i=1,2,\cdots,n)$ 的权重作归一化处理：

$$\bar{w}(t_k)=\frac{w(t_k)}{\sqrt{\sum_{i=1}^{n}w(t_i)^2}} \tag{2}$$

其中，n 表示不同特征词的数目；$\bar{w}(t_k)$ 就是归一化后的最终权重。

众所周知，自动文本分类是利用已经分好类的训练文本集来对待分类的新文本的类别进行预测，但是TF-IDF方法并没有利用已知的文本类别信息。例如，假设有两个特征词 t_1 和 t_2，其文档频率相同 $df_1=df_2$，所不同的是，t_1 在多个类别的文本中出现，而 t_2 只在单个类别的文本中出现。显然 t_2 的类别区分力比 t_1 大，但是它们用反文档频率（inverse document frequency，IDF）表示的全局权重因子是相同的。因此，TF-IDF权重不能充分反映特征词在文本分类中的重要性。

2.2 有监督的TF-RF方法

为了克服TF-IDF方法在文本分类中的不足，研究人员提出了有监督词加权的概念[2]，即利用已知的文本类别信息来计算特征词的权重。很多STW方法都采用文本分类中的特征选择指标，比如卡方统计量（Chi-square）、信息增益、互信息量等，以取代传统的IDF因子或者作为附加的全局权重因子[2-3]。也有一些研究人员提出了新的STW方法[3,10-12]，其中典型代表就是TF-RF（term frequency & relevance frequency），它在多个场合比TF-IDF等其他方法更加优越[3,11]。根据TF-RF方法，特征词 t_k 在属于类别 c_j 的某个文档中的权重 $w(t_k,c_j)$ 计算方法如下：

$$w(t_k,c_j)=tf_k\times \mathrm{lb}\left(2+\frac{df_{kj}}{\max(df_k-df_{kj},1)}\right) \tag{3}$$

其中，df_{kj} 和 df_k-df_{kj} 分别表示特征词 t_k 在正类（c_j 类）和反类（非 c_j 类）文本集中出现的文档频率；tf_k 和 df_k 分别为 t_k 的词频和总文档频率。显然有 $df_k=\sum_{j=1}^{m}df_{kj}$，其中 m 为类别数。

然而，上面有关STW方法的研究工作大多数都只考虑特征词在正反两类文本上的粗粒度分布，并且实验结果都是从两类分类实验中得到的，即使使用了多类别数据集，也是以一对余（one-against-rest）的方式进行多次正反两类分类实验。因此，这些权重计算方法对于两类以上的多类别文本分类不一定是最优的。

2.3 基于熵的权重计算方法

为了进一步提高文本分类的性能，在为特征词分配权重时，就有必要考虑它在多个文本类别上的细粒度分布。根据其分布特性来判断特征词的类别相关性，从而为它分配合适的权重。特征词在文本集中的分布特性可以用香农（Shannon）的信息熵理论来分析。在文本分类领域，文献[4]较早将信息熵理论用于特征词权重计算，并通过理论推导提出了一种新的权重计算方法：

$$w(t_k,c_j)=-\left(\frac{df_{kj}}{df_k}\times\frac{N_j}{N}\right)\times \mathrm{lb}\left(\frac{df_{kj}}{df_k}\times\frac{N_j}{N}\right) \tag{4}$$

其中，$w(t_k,c_j)$ 表示特征词 t_k 与类别 c_j 相关的权重；N_j 表示类别 c_j 中的文档数；N 表示训练集中的总文档数；df_{kj} 和 df_k 的含义与式（3）相同，分别表示特征词的类别文档频率和总文档频率。

然而，这种方法存在严重的问题。首先，论文中

理论分析有错，比如作者在用Bayes定理进行推导时错误地将以 c_j 为条件的 t_k 的概率 $P(t_k|c_j)$ 表示为 df_{kj}/df_k，实际上这个比值应该是条件概率 $P(c_j|t_k)$。概念错误最终导致结论错误。其次，由于原文没有给出实验结果，用这种方法在TanCorp语料库上做了文本分类实验（具体实验方案见第4章），得到的实验结果如表1所示，其中EWdiao就是文献[4]提出的权重计算方法。表1给出了当选择不同特征数时两种方法所对应的用微平均 F_1 值（$micro\text{-}F_1$）表示的文本分类准确率。很明显，用式（4）表示的EWdiao方法的性能比TF-IDF差得多。

Table 1 Performance comparison between two term weighting methods

表1 两种特征词权重计算方法的性能比较

特征数	文本分类准确率 $micro\text{-}F_1$	
	TF-IDF	EWdiao
500	0.931 1	0.843 5
1 000	0.942 6	0.851 0
2 000	0.947 1	0.842 0
4 000	0.952 3	0.816 2
6 000	0.950 5	0.814 8

特征词在不同类别的文本中出现具有一定的不确定性，这种不确定性可用熵（entropy）来度量。对于类别相关的特征词，不确定性小，则熵小，应分配大的权重；而对于类别无关的特征词，不确定性大，则熵大，应分配小的权重。因此，特征词的权重与熵的大小是相反的关系。基于这种思想，近几年研究人员提出了几种新的基于信息熵的特征词权重计算方法，统称为熵加权（entropy-based weighting，EW）方法。文献[5]和[6]都提出了在TF-IDF权重中引入信息熵因子的方法，并且这种权重因子是根据特征词 t_k 的类间分布熵 $H(t_k)$ 的倒数 $1/H(t_k)$（简称为反熵）来计算的。两者的主要区别有两点：一是权重归一化处理顺序不同，文献[5]是先将TF-IDF权重按式（2）进行余弦归一化后再乘以信息熵因子，而文献[6]是先将TF-IDF权重乘以信息熵因子后再进行余弦归一化。二是信息熵因子的表示略有不同，文献[5]使用反熵的对数 $\mathrm{lb}(1/H(t_k)+1)$，而文献[6]直接用反熵 $1/H(t_k)$ 作为权重因子。此外，为了避免分母变为0，两者都附加了一个相似的非零函数值，即用 $H(t_k)+\varphi(df_k)$ 来代替 $H(t_k)$，其中 $\varphi(df_k)$ 是特征词 t_k 的文档频率 df_k 的函数。但是文献[7]与上面两种方法不同，为了改进TF-IDF他们提出了用信息熵因子取代IDF因子的做法，并且把信息熵因子表示为 $h-H(t_k)$，其中 h 是一个比 $H(t_k)$ 大的常数，但原文并未明确其取值为多少。应当指出，在上面3种方法中，$H(t_k)$ 都是根据特征词 t_k 在不同文本类别 $c_j(j=1,2,\cdots,m)$ 中出现的概率 $P(t_k,c_j)$ 来计算的，但是类别概率 $P(t_k,c_j)$ 的计算方法不同，分别为 df_{kj}/df_k[5]、$df_{kj}/(df_k+1)$[6] 和 df_{kj}/N[7]（这里 N 为总文档数）。

除了上述根据特征词的类间分布熵来计算权重的方法外，也有一些研究人员提出将特征词在每个类别内部的分布信息熵也引入权重计算中，比如文献[8，13-14]。第4.6节将讨论这些引入类内分布熵的方法的有效性。

尽管上面提到的一些方法在特定语料库的文本分类实验中已被证明是有效的，但是至今没有人对这些方法在其他不同语料库上的性能做进一步的验证并与更多方法进行比较，尤其是没有将几种不同的熵加权方法的性能做比较。而且，通过实验也发现，上述方法在不同语料库上的性能不稳定，有时表现得比传统的TF-IDF方法更差。鉴于此，通过反复研究，提出了一种新的熵加权方法，并在不同数据集上做了大量文本分类实验，验证了它的有效性和优越性。

3 新的熵加权方法

3.1 特征词的类别区分力

特征词的权重应当根据它在文本分类中的重要性来分配，而特征词的重要性体现在它的类别区分力（class distinguishing power，CDP）的大小，因为类别区分力大的词更有助于区分不同类别的文本。显然，一个只与单类相关的特征词具有比与多类相关的特征词更大的类别区分力。类别区分力大的特征词往往集中出现在单个或少数类别中，它们在多个类别上的分布表现出高度不均匀性。这种不均匀性

可以用特征词的类间分布熵来度量，比如类别文档频率（DF）分布熵，表示如下：

$$E_{df}(t_k)=-\sum_{j=1}^{m}\left(\frac{df_{kj}}{df_k}\right)\times \mathrm{lb}\left(\frac{df_{kj}}{df_k}\right) \tag{5}$$

其中，$E_{df}(t_k)$ 为特征词 t_k 的类别DF分布熵；df_{kj} 和 df_k 分别为 t_k 在类别 $c_j(j=1,2,\cdots,m)$ 和训练集中的文档频率，m 为类别数，$df_k=\sum_{j=1}^{m}df_{kj}$，而 $df_{kj}/df_k=P_{kj}$ 为 t_k 在类别 c_j 中出现的概率。

当特征词只出现在单个类别的文本中时，它的类别区分力最大，而熵 $E_{df}(t_k)$ 最小且为0。当特征词在所有类别 $c_j(j=1,2,\cdots,m)$ 中均匀分布时，它的类别区分力最小，而熵 $E_{df}(t_k)$ 达到最大值 $E_{\max}=\mathrm{lb}(m)$。因为特征词的类别区分力与类别DF分布熵是相反的关系，所以可这样来度量：$CDP(t_k)=1-E_{df}(t_k)/\mathrm{lb}(m)$，也就是说，用归一化熵来度量特征词 t_k 的类别区分力，显然有 $0\leqslant CDP(t_k)\leqslant 1.0$。

3.2 LTF-ECDP方法

为了给特征词分配合适的权重，定义了一个基于类别区分力的全局权重因子，即 $G(t_k)=1+\alpha\times CDP(t_k)$，其中系数$\alpha$的值可针对不同语料库来调节，一般取值为5~7比较合适。至于特征词权重中的局部因子，一般用特征词在文档中的词频 (tf_k) 来表示。但是，一个在文档中出现20次的特征词的重要性并不是仅出现1次的特征词重要性的20倍，因此要适当降低高频词的局部词频因子，可使用对数词频 $\mathrm{lb}(tf_k+1)$ 来代替原始词频 tf_k[15]。综上所述，特征词 t_k 在某个文档中的权重 $w(t_k)$ 可以用式（6）来计算。

$$w(t_k)=\mathrm{lb}(tf_k+1)\times\left(1+\alpha\times\left(1-\frac{E_{df}(t_k)}{\mathrm{lb}(m)}\right)\right) \tag{6}$$

当然，最终同一文档中所有特征词的权重 $w(t_k)$ $(k=1,2,\cdots,n)$ 也要按照式（2）进行余弦归一化。本文把这种新的熵加权方法称为LTF-ECDP（logarithmic term frequency & entropy-based class distinguishing power），即对数词频与基于熵的类别区分力度量因子相结合的特征词权重计算方法。

3.3 新方法的两个变种

为了便于比较，引入了LTF-ECDP方法的两个变种。第一个变种称为TF-ECDP，其局部因子仍为原始词频，即 tf_k，而不是对数词频 $\mathrm{lb}(tf_k+1)$。第二个变种称为TF-ECDP-EIC，即在第一个变种中引入了特征词的类内信息熵（entropy in a class，EIC）因子，其理由是：一个具有类别代表性的特征词应当被分配更大的权重，它通常在某个类别的所有文档内分布比较均匀，而类内分布的均匀性同样可用信息熵来度量[13-14]。特征词 t_k 在类别 c_j 内的信息熵 $E_{ic}(t_k,c_j)$ 可根据它在类内各文档中的词频分布来计算：

$$E_{ic}(t_k,c_j)=-\sum_{i=1}^{|c_j|}\left(\frac{tf_k(d_i)}{tf_k(c_j)}\right)\times\mathrm{lb}\left(\frac{tf_k(d_i)}{tf_k(c_j)}\right) \tag{7}$$

其中，$tf_k(d_i)$ 为特征词 t_k 在属于类别 c_j 的文档 $d_i(i=1,2,\cdots,|c_j|)$ 中出现的频率；$|c_j|$ 表示 c_j 类中的文档数；$tf_k(c_j)=\sum_{i=1}^{|c_j|}tf_k(d_i)$，为 t_k 在 c_j 类中的总词频。如果特征词在某个类别中的TF分布熵越大，意味着它在这个类别内分布比较均匀，则它的类别代表性越大，它在该类别的所有文档中都应当被分配更大的权重。可见，引入类内TF分布熵因子以后，特征词的权重大小是与特定类别有关的。但是这种方法不一定有效，将在第4.6节讨论。

4 实验结果与分析

4.1 数据集及其预处理

本文实验使用了3个具有不同特点的公开数据集，包括一个中文语料库TanCorp和两个英文语料库WebKB和20 Newsgroups。前两个非平衡语料库的各类文档数不相等，第三个平衡语料库的各类文档数基本相等。3个语料库的文本来源也不同。

TanCorp语料库[16]有多个版本，选择其中预处理格式的TanCorp-12语料库，共有14 150篇中文文档，分为12类，各类别规模差别大，无异类重复文档，所有文本预先已用中文分词器ICTCLAS分词，并去掉了数字与标点符号。从中提取出72 601个不同词条构成初始特征词表，并把语料库按类别随机分割为训练集（占66%）和测试集（占34%）。

原始WebKB语料库[17]包含大约8 300个英文网

页,分为7大类。只选择其中最常用的4大类,包括student、faculty、course和project类别,共有4 199个文档。这个被称为WebKB-4的文本子集又进一步按2:1的比例被随机分割为训练集和测试集。通过删除停用词、单字符和非字母符号,并把字母转换为小写,提取词根(stemming)等预处理后,从训练集文本中共提取出7 287个不同的初始特征词。此外,为了提高实验的可靠性,移除了部分重复文档,最终训练集和测试集各剩下2 756和1 375个文档。

20 Newsgroups语料库包含20个类别的英文消息文本。本文所用的20 News-bydate版本[18]共有18 846篇文档,预先已按日期排序并分割为训练集(包含11 314篇文档)和测试集(包含7 532篇文档),所有重复文档和某些消息头部已被删除。通过与WebKB语料库类似的预处理后,从20 News-bydate的训练集文本中共提取出35 642个不同的初始特征词。

4.2 实验步骤与方法

对数据集进行预处理后,再按顺序经过特征选择、特征词权重计算、分类器训练及分类测试、性能评估等步骤开展文本分类实验。

特征选择采用流行的卡方统计量(Chi-square或χ^2)指标。特征词t_k关于类别c_j的卡方统计量$\chi^2(t_k,c_j)$可用下式来计算:

$$\chi^2(t_k,c_j)=\frac{N\times\left(df_{kj}\times N_x-df_{knj}\times(N_j-df_{kj})\right)^2}{df_k\times(N-df_k)\times N_j\times(N-N_j)} \tag{8}$$

其中,$N_x=N-N_j-df_{knj}$;$df_{knj}=df_k-df_{kj}$;N为总文档数;N_j为c_j类中的文档数;df_k和df_{kj}分别为特征词t_k在训练集和类别c_j中的文档频率。t_k的总评分为$score(t_k)=\max\{\chi^2(t_k,c_j)\,|\,j=1,2,\cdots,m\}$。根据$score(t_k)$,从初始特征词表中选择部分评分较高的词作为文档特征集。分批次为TanCorp-12语料库选择了{200,500,1 000,2 000,4 000,6 000}个特征,为WebKB-4语料库选择了{50,100,250,500,1 000,1 500,2 500,4 000}个特征,为20 Newsgroups语料库选择了{500,1 000,2 000,4 000,6 000}个特征。

为了比较性能,尝试了前面介绍的8种特征词权重计算方法,分别是LTF-ECDP、TF-ECDP、TF-ECDP-EIC、EWzhou、EWguo、EWxue、TF-RF和TF-IDF,其中第4~6个分别代表文献[5]、[6]和[7]提出的熵加权(entropy-based weighting,EW)方法。开头3种采用了基于熵的类别区分力(ECDP)度量因子的方法中,参数α均设为7.0。因为用TF-ECDP-EIC和TF-RF方法[3]计算的特征词权重都与文档类别有关,如式(7)和(3)所示,而待分类的文档的类别是未知的,所以对于测试集文档中的每个特征词,用其与各类别相关的权重的最大值作为它的权重。当一个文档中所有特征词的权重都已得到,再按照式(2)进行余弦归一化。但EWzhou方法[5]例外,它是先对所有词的TF-IDF权重进行归一化,再乘以熵加权因子。通过权重计算,每个文档都被转换成特征词权重向量。

为了实现文本分类,采用性能优良的支持向量机(support vector machine,SVM)作为分类器。具体做法是:在TanCorp和20 Newsgroups语料库上使用软件包LibSVM分类器[19-20],并设置线性核和默认参数;在WebKB语料库上使用LibLINEAR分类器[20],其参数也是默认的。LibLINEAR是对带有线性核的LibSVM进行优化后的分类器,性能更好。先用训练集文档特征向量来训练SVM分类器,再用SVM分类器对测试集文档特征向量进行分类。

最后的性能评估使用微平均F_1值($micro\text{-}F_1$)和宏平均F_1值($macro\text{-}F_1$)两个指标来度量所有类别的总体分类准确率,其定义分别为式(9)和(10)。

$$micro\text{-}F_1=\frac{2P\times R}{P+R} \tag{9}$$

$$macro\text{-}F_1=\frac{1}{m}\sum_{j=1}^{m}F_{1j} \tag{10}$$

其中,P为整个测试集分类结果的精确率;R为整个测试集被正确分类的召回率;$F_{1j}=2P_j\times R_j/(P_j+R_j)$为第$j$类($j=1,2,\cdots,m$)的分类性能,$m$为类别数,$P_j$和$R_j$分别为第$j$类文本分类精确率和召回率。

4.3 在TanCorp-12上的实验结果分析

首先用带线性核的LibSVM分类器对TanCorp-12语料库里的中文文本进行分类,用微平均F_1值和宏平均F_1值所度量的总体分类准确率如图1所示。图中每条曲线代表一种特征词权重计算方法,水平坐标轴显示不同特征数。

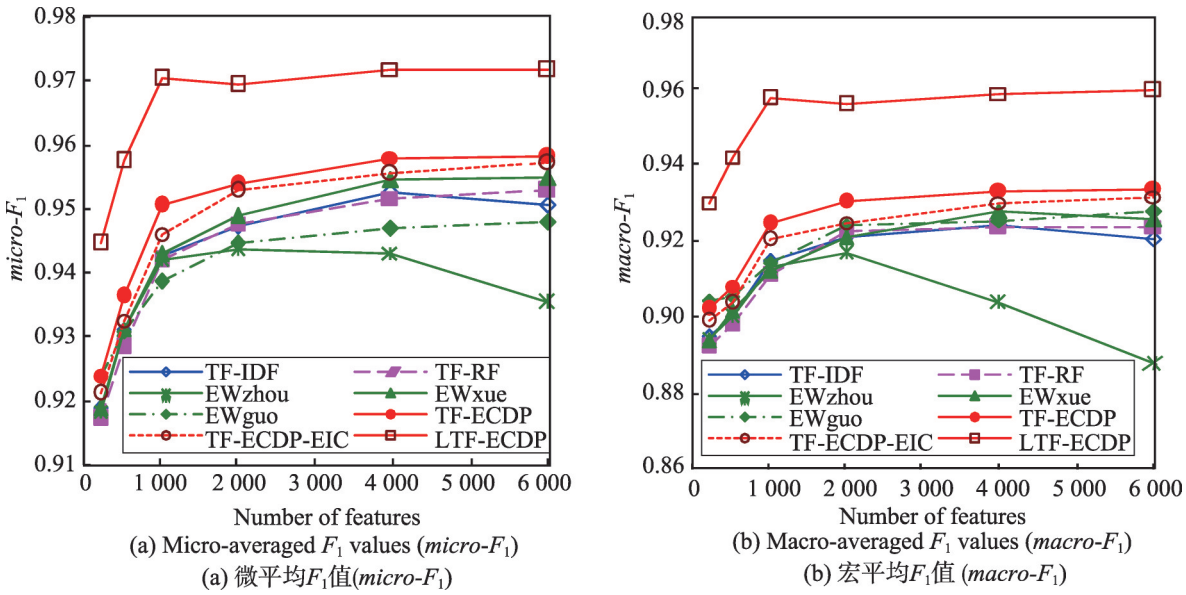


Fig.1 Accuracies of text categorization using different term weighting methods on TanCorp-12 corpus

图1 在TanCorp-12语料库上使用不同特征词权重计算方法的文本分类准确率

从图1中可以看出，3种新的特征词权重计算方法LTF-ECDP、TF-ECDP和TF-ECDP-EIC的性能都比其余方法更好。特别是，性能最好的LTF-ECDP方法具有明显的优势。就 $micro\text{-}F_1$ 和 $macro\text{-}F_1$ 而言，LTF-ECDP超越TF-IDF分别约2.8%和4.3%。引入特征词类内分布熵因子的TF-ECDP-EIC的性能略低于TF-ECDP。至于文献中的3种熵加权方法，EWxue的性能表现是最好的，略好于TF-RF。而EWzhou表现最差，明显不如TF-IDF，特别是在数据集特征维度较高时。EWguo则表现不同，就 $micro\text{-}F_1$ 而言，它比TF-IDF差；但就 $macro\text{-}F_1$ 而言，它比TF-IDF略好。而TF-RF的性能与TF-IDF相当。

4.4 在WebKB-4上的实验结果分析

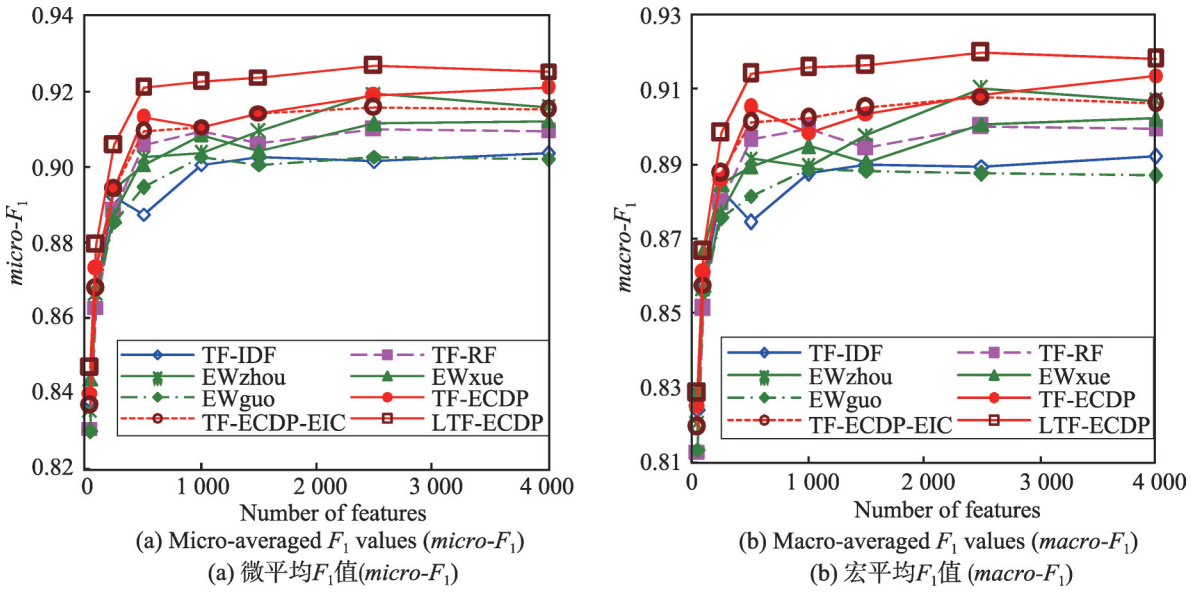
然后用性能更好的LibLINEAR分类器对WebKB-4语料库里的英文网页进行分类，分别用微平均 F_1 值和宏平均 F_1 值所度量的总体分类准确率如图2所示，图中各项的含义与图1相同。

从图2中可以看出，3种新的特征词权重计算方法LTF-ECDP、TF-ECDP和TF-ECDP-EIC的性能表现总体上仍然比其余方法更好，并且LTF-ECDP还是最好的。就 $micro\text{-}F_1$ 和 $macro\text{-}F_1$ 而言，它超越TF-IDF分别约3.3%和4.0%。TF-ECDP和TF-ECDP-EIC两者的性能不相上下。但是文献中的3种熵加权方法的关系发生了变化：EWzhou由最差变为最好，EWguo变为最差，而EWxue居中。EWzhou、EWxue和TF-RF的性能都比TF-IDF更好。但是EWguo的性能与TF-IDF相当，或比后者略差。

4.5 在20 Newsgroups上的实验结果分析

最后仍用LibSVM分类器对20 Newsgroups语料库里的英文消息文本进行分类，总体分类准确率如图3所示，图中各项的含义与图1相同。

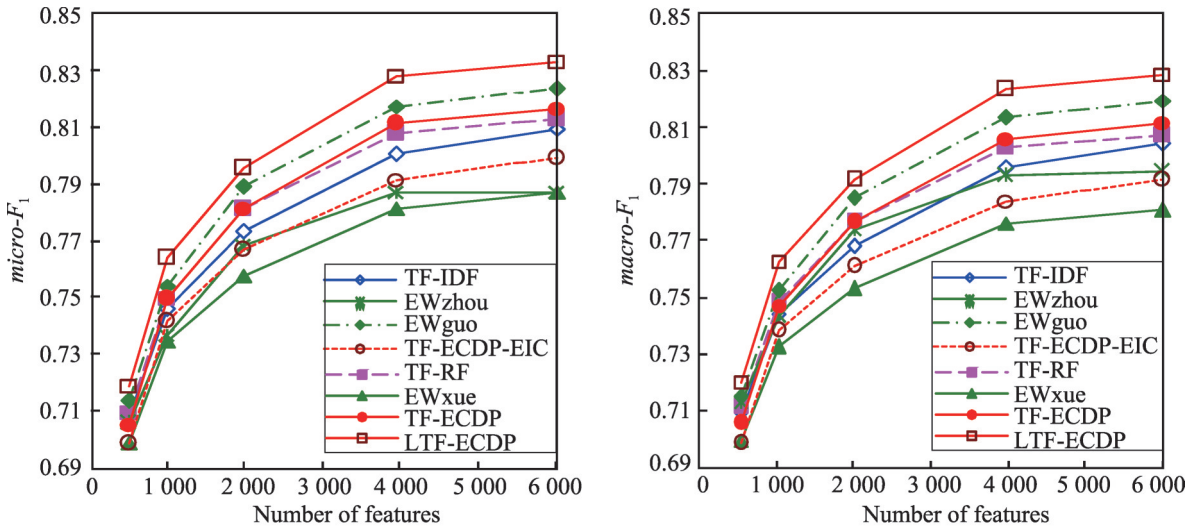
从图3中可以看出，3种新的特征词权重计算方法LTF-ECDP、TF-ECDP和TF-ECDP-EIC在20 Newsgroups上的性能差别较大，其中LTF-ECDP的性能最佳。就 $micro\text{-}F_1$ 和 $macro\text{-}F_1$ 而言，它超越TF-IDF达2.8%左右。而TF-ECDP胜过其余5种方法，只有1种例外。但是TF-ECDP-EIC的性能比较差。文献中的3种熵加权方法的关系发生了戏剧性的变化：前面表现最差的EWguo变为最好的，前面一直表现好的EWxue变为最差的，而EWzhou居中。EWzhou、EWxue和TF-ECDP-EIC熵加权方法都表现得比TF-IDF更差。在平衡语料库20 Newsgroups上，TF-RF和EWguo都表现比较好，胜过TF-IDF，这与文献[3]和[6]的实验结果是一致的。



(a) Micro-averaged F_1 values (*micro-*F_1)
(a) 微平均F_1值(*micro-*F_1)
(b) Macro-averaged F_1 values (*macro-*F_1)
(b) 宏平均F_1值 (*macro-*F_1)

Fig.2 Accuracies of text categorization using different term weighting methods on WebKB-4 corpus

图2 在WebKB-4语料库上使用不同特征词权重计算方法的文本分类准确率



(a) Micro-averaged F_1 values (*micro-*F_1)
(a) 微平均F_1值(*micro-*F_1)
(b) Macro-averaged F_1 values (*macro-*F_1)
(b) 宏平均F_1值 (*macro-*F_1)

Fig.3 Accuracies of text categorization using different term weighting methods on 20 Newsgroups corpus

图3 在20 Newsgroups语料库上使用不同特征词权重计算方法的文本分类准确率

4.6 关于实验结果的讨论

上面的实验结果是在3个具有不同特点的公共测试语料库上得出的。实验结果表明,LTF-ECDP和TF-ECDP方法不仅有效,而且比其他熵加权方法和著名的TF-RF、TF-IDF方法更好。这两种新的特征词权重计算方法不仅提高了分类准确率,而且在不同语料库上的性能表现稳定。尤其是LTF-ECDP方法的表现一直是最好的,并且具有明显的优势。而其余4种熵加权方法在不同语料库上的性能表现波动性比较大,跟TF-IDF方法相比,它们的表现有时好有时差。另外,TF-RF方法[3]的优越性也再次得到验证,它的性能也比较稳定,不比TF-IDF差,而且有时

更好。但是，TF-RF 的性能还是不如本文提出的 LTF-ECDP 和 TF-ECDP 方法。

在所有实验中，LTF-ECDP 表现得比 TF-ECDP 更优越，这再一次通过实验证实了特征词在文本分类中的重要性与其词频一般不是成正比的，因此有时不要对高频词在文本分类中的作用寄予太大的期望。当然，类别相关的高频词例外。一个特征词的重要性或对文本分类的贡献度主要取决于它的类别区分力。一个类别区分力大的词不一定是高频词，而主要体现在它在不同文本类别上的分布很不均衡。

上述实验结果还显示了新方法的另一个变种 TF-ECDP-EIC 的性能并没有预期的那么好，它不但没有在 TF-ECDP 的基础上进一步提高文本分类的性能，有时反而降低了分类准确率。引入特征词的类内分布熵的目的是给具有类别代表性的词分配更大的权重，因为代表某一类别的词在该类别各文档上的分布比其他非代表性的词更加均匀，对应的类内分布熵更大。这听起来似乎有理，但是忽视了一个事实：代表整个类别（尤其是大类）的词毕竟是少数，而大多数类别区分力大的词只能代表其中一个小的子类。比如："古筝"属于"艺术"类但不能代表"艺术"。一篇文章中如果出现"古筝"，很容易被判断为跟"艺术"有关。可见"古筝"一词具有较大的类别区分力，应当被分配较大的权重。但是"古筝"在整个"艺术"类中出现频率较低，一旦引入类内分布熵，它的权重将明显降低。而能够代表整个艺术类的词汇很少。只有当语料库的各类别规模较小或各类别代表性词汇较多时，在特征词权重中引入类内分布熵才会有效。但是在一般情况下，引入类内分布熵很可能会失效。

最后应当指出，所有文本分类实验都是用带有线性核的支持向量机（简称为线性 SVM）来实现的，并且尝试了在数据集的多个不同特征维度上进行分类测试。之所以选择线性 SVM，是因为它对文本分类的性能很好。尽管一些研究人员在努力改进其他分类算法，比如朴素贝叶斯算法、k 近邻（k nearest neighbors，kNN）分类器、中心点（centroid）分类器、决策树算法、神经网络等[9]，但它们对文本分类的性能还是难以超越 SVM。上述实验结果再次证明了通过改进特征词权重计算方法和调节特征维度，可以进一步提高 SVM 文本分类性能。由于篇幅的限制，本文没有给出使用其他分类器的实验结果。事实上，本文提出的特征词权重计算方法 LTF-ECDP 也能明显提高 k 近邻分类器的文本分类性能。而 k 近邻分类器更易于在分布式的云计算环境中实现。本文提出的 LTF-ECDP 方法即使在特征维度较低时也能获得较好的分类准确率，更适合大规模文本分类应用。

5 结束语

相比于其他有监督词加权方法而言，基于信息熵的特征词权重计算方法（简称为熵加权）更加有效，因为前者通常只利用了特征词在正反两类上的粗糙分布信息，而后者考虑了特征词在所有类别上的精细分布。但是，现有的熵加权方法用于不同语料库的文本分类时效果变化比较大，有时表现得比传统的 TF-IDF 方法更差。本文提出了一种新的熵加权方法 LTF-ECDP（对数词频-基于熵的类别区分力）以及它的两个变种 TF-ECDP 和 TF-ECDP-EIC。在 TanCorp、WebKB 和 20 Newsgroups 这 3 个具有不同特点的语料库上使用支持向量机进行文本分类的实验结果表明，LTF-ECDP 和 TF-ECDP 方法不仅有效，而且它们的性能优于其他熵加权方法以及 TF-IDF 和 TF-RF 等著名方法，不仅进一步提高了文本分类准确率，而且性能更加稳定。尤其是 LTF-ECDP 具有明显的优势。同时也发现，虽然 LTF-ECDP 和 TF-ECDP 都只利用了特征词的类间分布熵，但是引入特征词的类内分布熵在大多数情况下并没有进一步改善文本分类的性能。与前两者对比，TF-ECDP-EIC 的表现稍差。

未来将把 LTF-ECDP 方法用于文本特征降维和某些 Web 数据分析任务（比如情感分析）中，并且开展更广泛的实验研究。

References:

[1] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval[J]. Information Processing and Manage-

ment, 1988, 24(5): 513-523.

[2] Debole F, Sebastiani F. Supervised term weighting for automated text categorization[C]//Proceedings of the 2003 ACM Symposium on Applied Computing, Melbourne, USA, Mar 9-12, 2003. New York, USA: ACM, 2003: 784-788.

[3] Lan Man, Tan C L, Su Jian, et al. Supervised and traditional term weighting methods for automatic text categorization[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(4): 721-735.

[4] Diao Qian, Wang Yongcheng, Zhang Huihui, et al. A Shannon entropy approach to term weighting in VSM[J]. Journal of the China Society for Scientific and Technical Information, 2000, 19(4): 354-358.

[5] Zhou Yantao, Tang Jianbo, Wang Jiaqin. Improved TFIDF feature selection algorithm based on information entropy[J]. Computer Engineering and Applications, 2007, 43(35): 156-158.

[6] Guo Hongyu. Research on term weighting algorithm based on information entropy theory[J]. Computer Engineering and Applications, 2013, 49(10): 140-146.

[7] Xue Wei, Xu Xinshun. Three new feature weighting methods for text categorization[C]//LNCS 6318: Proceedings of the 2010 International Conference on Web Information Systems and Mining, Sanya, China, Oct 23-24, 2010. Berlin, Heidelberg: Springer, 2010: 352-359.

[8] Li Ran, Guo Xianjiu. An improved algorithm to term weighting in text classification[C]//Proceedings of the 2010 International Conference on Multimedia Technology, Ningbo, China, Oct 29-31, 2010. Piscataway, USA: IEEE, 2010: 1-3.

[9] Sebastiani F. Machine learning in automated text categorization[J]. ACM Computing Surveys, 2002, 34(1): 1-47.

[10] Liu Ying, Loh H T, Sun Aixin. Imbalanced text classification: a term weighting approach[J]. Expert Systems with Applications, 2009, 36(1): 690-701.

[11] Hakan A, Zafer E. Analytical evaluation of term weighting schemes for text categorization[J]. Pattern Recognition Letters, 2010, 31(11): 1310-1323.

[12] Nguyen T T, Chang K, Hui S C. Supervised term weighting centroid-based classifiers for text categorization[J]. Knowledge and Information Systems, 2013, 35(1): 61-85.

[13] Yi Junkai, Tian Likang. A text feature selection algorithm based on class discrimination[J]. Journal of Beijing University of Chemical Technology: Natural Science, 2013, 40 (S1): 72-75.

[14] University of Electronic Science and Technology of China. A method of text classification based on feature selection and weight calculation: China, CN102930063A[P]. 2013-02-13.

[15] Dumais S. Improving the retrieval of information from external sources[J]. Behavior Research Methods, Instruments, and Computers, 1991, 23(2): 229-236.

[16] Tan Songbo, Cheng Xueqi, Ghanem M M, et al. A novel refinement approach for text categorization[C]//Proceedings of the 14th ACM International Conference on Information and Knowledge Management, Bremen, Germany, Oct 31-Nov 5, 2005. New York, USA: ACM, 2005: 469-476.

[17] CMU text learning group. The 4 universities data set (WebKB corpus) [EB/OL]. (1998-01-11)[2015-06-30]. http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/.

[18] Ken Lang, Rennie J. The 20 Newsgroups data set[EB/OL]. (2008-01-14) [2015-06-30]. http://people.csail. mit.edu/jrennie/20Newsgroups/, http://qwone.com/~jason/20Newsgroups/.

[19] Chang C C, Lin C J. LIBSVM: a library for support vector machines[J]. ACM Transactions on Intelligent Systems and Technology, 2011, 2(3): 1-27.

[20] Chang C C, Lin C J. LIBSVM—a library for support vector machines[EB/OL]. [2015-06-30]. http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html.

附中文参考文献：

[4] 刁倩, 王永成, 张惠惠, 等. VSM中词权重的信息熵算法[J]. 情报学报, 2000, 19(4): 354-358.

[5] 周炎涛, 唐剑波, 王家琴. 基于信息熵的改进TFIDF特征选择算法[J]. 计算机工程与应用, 2007, 43(35): 156-158.

[6] 郭红钰. 基于信息熵理论的特征权重算法研究[J]. 计算机工程与应用, 2013, 49(10): 140-146.

[13] 易军凯, 田立康. 基于类别区分度的文本特征选择算法研究[J]. 北京化工大学学报: 自然科学版, 2013, 40(S1): 72-75.

[14] 电子科技大学. 一种基于特征项选择与权重计算的文本分类方法: 中国, CN102930063A[P]. 2013-02-13.

CHEN Kewen was born in 1970. He is a Ph.D. candidate in computer application technology at Central South University, and the member of CCF. His research interests include machine learning, text mining and information fusion, etc.

陈科文(1970—),男,湖南湘潭人,中南大学计算机应用技术博士研究生,CCF会员,主要研究领域为机器学习,文本挖掘,信息融合等。

ZHANG Zuping was born in 1966. He received the Ph.D. degree in computer application technology from Central South University in 2005. Now he is a professor and Ph.D. supervisor at Central South University, and the senior member of CCF. His research interests include information fusion and information system, parameter computing and biology computing, etc.

张祖平(1966—),男,湖南湘乡人,2005年于中南大学获得计算机应用技术博士学位,现为中南大学教授、博士生导师,CCF高级会员,主要研究领域为信息融合与信息系统,参数计算,生物计算等。

LONG Jun was born in 1972. He received the Ph.D. degree in computer application technology from Central South University in 2011. Now he is a professor and Ph.D. supervisor at Central South University, and the senior member of CCF. His research interests include service computing, Internetware, software engineering methods to solve scientific problems in big data, etc.

龙军(1972—),男,安徽安庆人,2011年于中南大学获得计算机应用技术博士学位,现为中南大学教授、博士生导师,CCF高级会员,主要研究领域为服务计算,网构软件,面向大数据的软件工程方法等。