Web 文本特征选择算法的研究*

冯长远，普杰信
（河南科技大学 电子信息工程学院，河南 洛阳 471003）

摘 要：以向量空间模型作为 Web 文本的表示方法，结合 Web 文本的结构特征对向量空间模型中的特征选择算法进行了分析并加以改进。在改进的算法中，体现出了特征词在 Web 文档结构中的位置信息；引入了信息论中熵的概念，用词的熵函数对权值进行调整，从而更加准确地选取有效的特征词。实验验证了改进算法的可行性和有效性。

关键词：文本表示；向量空间模型；特征选择；熵
中图法分类号：TP393 **文献标识码**：A **文章编号**：1001-3695(2005)07-0036-03

Research about Algorithm of Web Text Feather Selection

FENG Chang-yuan, PU Jie-xin
(Collgege of Electronic Information Engineering, Henan University of Science & Technology, Luoyang Henan 471003, China)

Abstract: This paper uses vector space model as the description of the Web text, analyses the feather selection algorithm and brings forward an improved algorithm in view of the construct character of the Web text. The new algorithm describes the situation information of the feather terms in Web text, introduces the concept of entropy and adjusts the weighting by the entropy-function of the words, thus it can select feather terms more effectively. The experiment shows the feasibility and the validity of this method in feather selection.

Key words: Text Represents; Vector Space Model; Feather Selection; Entropy

随着 Internet 上信息资源的迅猛增加，以及人们对能够从 Web 上快速、有效地发现资源和信息的工具的迫切需要，大大促进了信息检索技术的发展，尤其是 Web 信息检索技术的发展。这主要表现在近年来更多的研究者关注于面向特定问题的解决方法的研究，如针对信息检索领域中的文本分类、聚类和自动文摘等算法的提出和改进，以及信息内容的智能化过滤等。文本的表示及其特征项的选取是信息检索的一个基本问题，它把从文本中抽取出的特征词进行量化来表示文本信息。这些特征词作为文档的中间表示形式，用来实现文档与文档、文档与用户目标之间的相似度计算[1]，对文本内容的过滤和分类、聚类处理以及用户兴趣模式发现等有关信息检索方面的研究都有非常重要的影响。目前有关文本表示的研究主要集中于文本表示模型的选择和特征词选择算法的选取上[2~4]，但是多数研究算法是针对普通文本结构的，对具有 HTML 结构特征 Web 文本，这些算法在准确表示文本信息的程度上仍然存在一些不足。

本文利用向量空间模型作为 Web 文本的表示方法，讨论了特征词选择和词权重算法的实现，对 TFIDF 算法进行了深入分析，给出了一种改进的 Web 文本特征获取算法，提高了 Web 文档的可计算性和可操作性。

1 Web 文本的表示

对于 Web 文本的表示方法有许多种，一般的方法是首先对 Web 文档中的信息进行预处理，提取出 HTML 结构中各个标记符中的文本信息，然后按照普通文本的处理方式表示。文本表示的模型有多种，近年来研究学者提出的模型有向量空间模型（Vector Space Model, VSM）[2]。VSM 把文档看作是由一组正交词条矢量所组成的矢量空间，每个文档表示为其中的一个范化特征矢量。布尔逻辑模型是 VSM 模型的一种简化，是一种严格匹配向量模型，其实现简单可用于快速检索；此外还有概率模型和混合模型。目前应用最多且效果较好，并被广泛接受的是 VSM。

在 VSM 中，从文本中提取其特征词组成特征向量，并计算出特征词的权重。例如文档可以表示为$(t_1, t_2, \cdots, t_N)$，其中 t_i $(1 \le i \le N)$ 是特征词。根据特征词的不同重要程度，可以赋予不同的权重 W_i 来进行量化，这样文档也可表示为$(W_1, W_2, \cdots, W_N)$，其中每一项 W_i 与相应的特征词 $t_i(1 \le i \le N)$ 对照。在 VSM 中，不考虑特征词在文中出现的先后顺序，只保证特征词的唯一性，然后把 $t_1, t_2, \cdots, t_N$ 看成一个 N 维的坐标系，则相应的 $W_1, W_2, \cdots, W_N$ 为文档在坐标系中的坐标值，一个文档就可以被表示成一个 N 维空间中的向量。这样就把文档以向量的形式定义到实数域中，使得机器学习或其他领域中成熟的计算方法得到应用，大大提高了文档的可计算性和可操作性。

2 Web 文本特征选取的方法

特征提取是指从文档中选取一部分能反映其内容信息的词语及其权重的计算。在对文档进行特征提取之前，需要先进行文本信息的预处理，包括 Web 文本信息的提取和分词处理。从文本中有意义地抽取特征词条是一项非常重要的技术，也是

收稿日期：2004-07-11；修返日期：2004-09-03
基金项目：国家教育部科研基金重点项目(200305)；河南省自然科学基金资助项目(0311012600)

文本处理的基本要求和前提条件。一个有效的特征词条集, 必须具备以下三个特征[5]: ①完全性, 特征词条能够确实表示目标内容; ②区分性, 根据特征矢量能将目标文档与其他文档相区分; ③精练性, 特征矢量的维数应该尽可能地小。

2.1　文档特征的选择方法

目前文本特征词的获取一般是通过一些分词算法和词频统计方法, 从文档中选出尽可能多的词、词组和短语, 由它们来构成文档矢量。但是如采用这种方法来表示文档, 文档矢量的维数将非常巨大, 这将给后续的文档处理带来巨大的计算开销, 使整个处理过程的效率非常低下。因此需要采取一定的方法进行文档矢量的降维。目前主要采用的方法是对文本特征进行选择, 选出最能代表文档信息的特征子集以降低文档矢量的维数。

针对文本特征选择, 国内外的研究学者已经提出了很多方法, 一种主要的方法就是采用某种评估函数对每一个特征词进行计算, 然后按照计算结果的高低排列, 数值大于预先设定的阈值的特征词被选取。在文本处理过程中主要的评估函数有文档频数(DF) 、信息增益(IG) 、期望交叉算熵、互信息(MI) 、文本证据权和 X^2 统计法等[4, 6]。另外一种方法是采用潜在语义索引(LSI) 的方法构造出文本词频矩阵, 利用单值分解技术来减少频数矩阵并保留最重要的行, 这样就可以减去原来文本词频矩阵中那些不重要的信息, 选取出有效的特征词[7, 8]。

2.2　特征权重的计算

特征词(关键词) 权重的计算方法相应于特征选择方法也有多种不同的算法, 常应用的有布尔值法、词频法和 TFIDF 法。在向量空间模型中特征词的权重计算方法一般采用 TFIDF 算法。

TFIDF[2] 法是以特征词在文档 d 中出现的次数与包含该特征词的文档数之比作为该词的权重, 即

$$W_i = \frac{TF_i(t, d)\log\left(\frac{N}{DF(t)} + 0.01\right)}{\sum_k TF_i^2(t, d)\log^2\left(\frac{N}{DF(t)} + 0.01\right)} \quad (1)$$

其中, W_i 表示第 i 个特征词的权重, $TF(t, d)$ 表示词 t 在文档 d 中的出现频率, N 表示总的文档数, $DF(t)$ 表示包含 t 的文档数。用 TFIDF 算法来计算特征词的权重值是表示当一个词在这篇文档中出现的频率越高, 同时在其他文档中出现的次数越少, 则表明该词对于表示这篇文档的区分能力越强, 所以其权重值就应该越大。

2.3　TFIDF 算法的分析

TFIDF 算法是建立在这样一个假设之上的: 对区别文档最有意义的词语应该是那些在文档中出现频率高, 而在整个文档集合的其他文档中出现频率少的词语, 所以如果特征空间坐标系取 TF 词频作为测度, 就可以体现同类文本的特点。另外考虑到单词区别不同类别的能力, TFIDF 法认为一个单词出现的文本频数越小, 它区别不同类别文本的能力就越大。因此引入了逆文本频度 IDF 的概念, 以 TF 和 IDF 的乘积作为特征空间坐标系的取值测度, 并用它完成对权值 TF 的调整, 调整权值的目的在于突出重要单词, 抑制次要单词。但是在本质上 IDF 是一种试图抑制噪音的加权[9, 10], 并且单纯地认为文本频数小的单词就越重要, 文本频数大的单词就越无用, 显然这并不是完全正确的。IDF 的简单结构并不能有效地反映单词的重要程度和特征词的分布情况, 使其无法很好地完成对权值调整的功能, 所以 TFIDF 法的精度并不是很高。

此外, 在 TFIDF 算法中并没有体现出单词的位置信息, 对于 Web 文档而言, 权重的计算方法应该体现出 HTML 的结构特征。特征词在不同的标记符中对文章内容的反映程度不同, 其权重的计算方法也应不同。因此应该对于处于网页不同位置的特征词分别赋予不同的系数, 然后乘以特征词的词频, 以提高文本表示的效果。

3　改进的算法

3.1　TF 算法的改进

目前的信息检索主要是针对于 Web 信息的检索, Internet 上的文本信息大多是 HTML 结构的, 对于处于 Web 文本结构中不同位置的单词, 其相应的表示文本内容或区别文本类别的能力是不同的, 所以在单词权值中应该体现出该词的位置信息。

在文献[11] 中作者对网页中不同位置的信息对表现文本内容的能力进行了测试, 结果表明网页不同标记符中的信息对于文本信息的检索有不同的影响。其中 Meta 标记中的信息最具有代表力, 如果使用 Meta 和 Title 标记符中的信息来表示文档, 其文本分类结果要好于仅用 Body 标记符中的信息来表示 Web 文本。由此可以看出在 Web 文本中不同位置上的信息在表达文档信息方面的贡献是不同的, 因此在我们的算法中对词频统计 TF 进行了改进, 特征词的权值用下面的式子表示:

$$LTF(t_i, d_j) = \sum_s a(s) \cdot TF(t_i, d_j, s) \quad (2)$$

其中, $LTF(t_i, d_j)$ 表示特征词 t_i 在 Web 文档 d_j 中各个标签位置中的权值之和; s 表示特征词出现在 Web 文档中的不同 HTML 标记中, 如 < title > , < meta > , < body > 和 < p > 等; $a(s)$ $(a(s) \geq 1)$ 表示对相应的 HTML 标记位置上的信息所赋予的权值系数, 其数值的大小可以通过试验来确定; $TF(t_i, d_j, s)$ 表示单词在网页中的某个位置出现的次数, 对于处在重要位置的信息赋予较大的系数, 处于普通或是无关紧要位置的信息给予较小的系数, 体现出 Web 文档的结构特征。在式(1) 中, 如果对于一般文本文档, 不考虑特征词的位置信息, 则可令 $a(s) = 1$, 则式(1) 就等同于 TFIDF 算法中的 TF 函数。

3.2　信息熵的引用

熵(Entropy) 在信息论中是一个非常重要的概念[12, 13], 它是不确定性的一种度量。信息熵方法的基本目的是找出某种符号系统的信息量和多余度之间的关系, 以便能用最小的成本和消耗来实现最高效率的数据储存、管理和传递。信息熵是数学方法和语言文字学的结合, 其定义为: 设 X 是取有限个值的随机变量, 各个取值出现的概率为

$$P_i = P\{X = x_i\} \quad i = 1, 2, \cdots, n$$

则 X 的熵为

$$Entropy(X) = -\sum_{i=1}^{n} P_i \log_a P_i \quad (3)$$

其中, 底数 a 可以为任意正数, 并规定当 $P_i = 0$ 时, $P_i \log_a P_i = 0$。在式(3) 中, 对数底 a 决定了熵的单位, 如 $a = 2$、e、10, 熵的单位分别为 Bit, nat, Hartley。在我们的研究论文中, 均取 $a = 2$。熵具有最大值和最小值[14], 由熵的定义公式可以看出, 当

每个值出现的概率相等时,即当 $P_1 = P_2 = \cdots = P_n$ 时

$$\text{Entropy}(X) = -\sum_{i=1}^{n} P_i \log_2 P_i = -\log_2 \frac{1}{n} = \log_2 n \qquad (4)$$

这时熵函数达到最大值 $\log_2 n$,记为最大熵 $E_{\max}$。其中 $P_i \geq 0$,并且 $\sum_{i=1}^{n} P_i = 1$。而当 $P_i = 1$, $P_j = 0 (j = 1, 2, \cdots, i-1, i+1, \cdots, n)$ 时,熵值最小,$\text{Entropy}(X) = 0$。

在信息论中,熵函数的值解释为信息不确定的度量。对于一组选好的特征词$(t_1, t_2, \cdots, t_n)$,取任意两个特征词 t_i 和 t_j,如果 $\text{Entropy}(Pt_i) > \text{Entropy}(Pt_j)$,那么我们可以说在表明一个事件方面,$t_i$ 比 t_j 好,因为 t_j 的不确定性度量要比 t_i 小[15]。

由此,我们将信息论中的熵原理引入到信息检索中的特征词权重的计算中。在信息检索中,文档中的特征词可以解释为熵理论中的特征,文档可以解释为事件,特征的熵值表示的其不确定性度量可以对特征词的有效性权重度量进行评估。为了便于特征词之间的比较,我们用 $E/E_{\max}$ 来规格化表示词的不确定度量,其中 E 表示特征词 t 的熵值,$E_{\max}$ 为最大熵值。但在研究中需要的是一个特征词对于一篇文档的确定性度量,因此我们用下面的式子作为特征词的确定性度量计算方法:

$$w_i = 1 - \frac{E(t_i)}{E_{\max}} \qquad E(t_i) = -\sum_{j=1}^{n} P_{ij} \log_2 P_{ij} \qquad (5)$$

其中,$E(t_i)$ 是特征词 t_i 的熵,P_{ij} 是特征词 t_i 在文档 j 中出现的概率,n 为文档集中的文档数。$E_{\max}$ 是最大熵值,也可以写成 $\log_2 n$。这样一个特征词的权重调整系数 w_i 被定位在[0, 1]之间,如果某个词在所有文档中的出现概率相同,其 w_i 值为 0,表明该词不具有代表性;如果某个词仅在一篇文档中出现,则 w_i 值为 1,表明该词就表示该篇文档而言最具有代表性。由此,我们用式(4)来代替 IDF 函数对特征的权值进行调整。改进后的权重计算公式为

$$W_i = LTF(t_i, d_j) \cdot w_i = \sum_s a(s) \cdot TF(t_i, d_j, s) \cdot \left(1 - \frac{E_i}{\log_2 N}\right) \qquad (6)$$

考虑到文档长度不一样对词频的统计具有一定的影响,一个特征词在长的文档中比在短的文档中更有可能被提取出来,因此采用对每个特征词的权重进行归一量化的方法来消除这种影响。处理的方法如下:

$$W_i' = W_i / \sqrt{\sum_{i=1}^{m} W_i^2} \qquad (7)$$

其中,W_i' 为特征词 t_i 的最后权重值,m 为文档中的特征词个数。

4 验证与分析

为了验证特征词的熵函数在权值调整方面比原来的算法优越,我们通过下面一个简单的例子来证明。设有两篇文档 Doc1 和 Doc2,每篇文档经过处理后有四个词语,在每篇文档中单词的频数如表 1 所示,表 2、表 3 为按照 TFIPF 和改进算法得到的权重。

表 1 文档词频表

文档\单词词频	Term1	Term2	Term3	Term4
Doc1	15	10	8	0
Doc2	16	6	0	4

表 2 按照 TFIDF 的算法得到的权重

文档\单词权重	Term1	Term2	Term3	Term4
Doc1	0.02671	0.01780	0.99948	0
Doc2	0.05335	0.02134	0	0.99834

表 3 按照改进后的算法得到的权重

文档\单词权重	Term1	Term2	Term3	Term4
Doc1	0.00140	0.05686	0.99838	0
Doc2	0.00299	0.06818	0	0.99767

通过比较我们可以看出,如果按照 TFIDF 算法,Doc1 的单词权重排序为 Term3, Term1, Term2; Doc2 的为 Term4, Term1, Term2。但是我们可以发现,Term1 在两篇文档中的出现频数相近,对于区分两篇文档毫无意义,但是它却排在特征权重的前列,很有可能被选为特征词,这就显示出了该算法的弊端,无法体现特征词在区分文档时所提供的信息量。通过改进的算法计算后,Term1 的权重为 0,说明不能用来区分两篇文档,这样选取的特征词也能更加准确地表示文档的内容。

5 结论

本文着重研究了文本特征选取中权重的计算问题,提出了一种改进的权重算法。该方法着重于两个方面:①结合 Web 文档的结构特征,加入特征词在 HTML 结构中的位置信息;②引用信息论中的熵理论,提出了一种用特征词的熵函数来代替 IDF 对特征词的权值进行调整。但我们必须考虑到的一个问题是,在已有的特征选择算法和我们的特征选择研究中,都有一个前提条件:假设文本中的特征词之间是相互独立的。它给文档的计算和操作提供了极大的方便,但是这样却损失了大量词语的关联信息,对于有效地进行文本特征选择和文本处理是有影响的。在我们以后的研究中,如何考虑特征词之间的相互关系,并在这种内在关系之上进行特征选择将是我们下一步继续考虑和研究的问题。

参考文献:

[1] 涂承胜,鲁明羽,陆玉昌. Web 内容挖掘技术研究[J]. 计算机应用研究,2003,20(11):5-9.

[2] Salton G, Wong A, Yang C S. A Vector Model for Automatic Indexing[J]. Communication of the ACM, 1975, 18(11): 613-620.

[3] Keli Chen, Chengqing Zong. A New-Weighting Algorithm for Linear Classifier[C]. Beijing: International Conference on Natural Language Processing and Knowledge Engineering, 2003. 650-655.

[4] Y Yang, J O Pedersen. A Comparative Study on Feature Selection in Text Categorization[C]. The 14th International Conference on Machine Learning, San Francisco: Morgan Kaufmann Publishers, 1997. 412-420.

[5] 刘明吉,王秀峰. Web 文本信息的特征获取算法[J]. 小型微型计算机系统,2002,23(6):683-686.

[6] Monica Rogati, Yiming Yang. High-Performing Feature Selection for Text Classification [C]. CIKM'02, New York: ACM Press, 2002. 4-9.

[7] 朱明. 数据挖掘[M]. 合肥:中国科学技术大学出版社,2002. 184-186.

[8] S Deerwester, S T Dumais. Indexing by Latent Semantic Analysis [J]. Journal of the American Society of Information Science, 1990, 391-407.

[9] 陆玉昌,鲁明羽. 向量空间法中单词权重函数的分析和构造[J]. 计算机研究与发展,2002,39(10): 1205-1210.

[10] 李凡,鲁明羽,陆玉昌. 关于文本特征抽取新方法的研究[J]. 清华大学学报(自然科学版),2001,41(7): 98-101.

[11] Daniele Riboni. Feature Selection for Web Page Classification[C]. Shiraz, Iran: EURASIA-ICT 2002, Proceedings of the Workshop, Austrian Computer Society, 2002. 1-5.

(下转第 59 页)