面向信息内容安全的文本过滤系统研究

张刚　刘挺　秦兵　车万翔　李生

哈尔滨工业大学信息检索组　150001

{ bert, tliu, qinb, car, sli }@ir.hit.edu.cn

摘要： 本文设计并实现了一个面向信息内容安全的汉语文本过滤器。该系统能够通过对正例、反例的学习，来提高自身的过滤性能。并给出了一个过滤器阈值选择的算法，使阈值的选择更为合理。详细描述了能够实现高效过滤的数据结构。最后给出了对大规模网页进行过滤的实验结果。

关键词： 特征词 向量空间模型 相似度 正例 反例 阈值反馈

Content Security Oriented Filtering System

Zhang Gang　Liu Ting　Qin Bing　Che Wangxiang　Li Sheng

Information Retrieval Group　Harbin Institute of Technology

{bert,tliu,qinb,car,sli}@ir.hit.edu.cn

Abstract: In this paper, a content security oriented filtering system is designed and implemented. Relevance feedback technology was developed in the system. A dynamic adjusting threshold algorithm was given. Efficient data structure was described in detail. A large amount of Web Pages were used to test the system.

Keywords: feature-word　vector space model　similarity　positive sample　negative sample　threshold feedback

1 前言

近年来，我国的因特网应用进入大发展阶段，随之而来的信息安全问题也日益突出，并逐渐成为社会性问题。不仅金融部门重视，企事业单位和个人也都日益重视起来。信息安全中除了信息系统的安全以外，还有一个迫切需要解决的难题，就是信息内容的安全问题。所谓信息内容的安全问题就是如何自动侦测出信息中的有害内容，并将其屏蔽掉。所谓有害信息主要指的是和色情、暴力有关内容，以及垃圾邮件等。这对于维护社会稳定以及保护青少年的身心健康具有极其重要的意义。

所谓信息过滤是指计算机根据用户提供的一个检索需求(Profile)，从动态变化的信息流（比如 Web）中自动检索出满足用户需求的信息。应用到信息内容安全领域就是根据用户提交的关于某种有害信息的特征描述，对网上的信息进行甄别，将满足这些特征的网页找出来。

2 系统概述

整个系统的流程如下图所示：

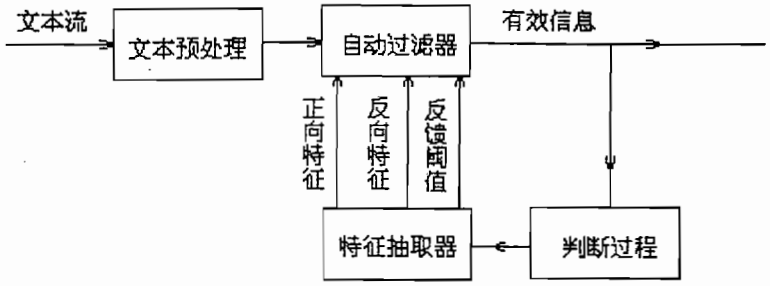

图 1 系统流程图

文本流首先要经过一个预处理的过程，在预处理阶段要对文本进行分词，同时去掉大部分对分类无用的信息。由于词是负载语义的基本单位，因此分词的效果好坏会直接影响到后面的其他处理过程。我们的分词模块能够解决大部分的组合歧义和交集型歧义，同时还有较强的未登录词的识别功能，能够很好的对人名、地名进行识别，这就为进一步的处理提供了基础。文本流经过预处理后进入自动过滤器模块，自动过滤模块根据用户配置文件来判断流入的信息是否为用户需求信息，经过自动过滤模块后，无关信息被过滤掉，流出的信息是系统认为有效的信息，鉴于系统的过滤准确率并不能达到 100%，这些信息中有的是用户需求相关的也有些是用户需求不相关的，经过用户判断后，我们把与用户需求相关的部分称为“正例”，与用户需求不相关的部分称为“反例”。正例、反例都被送入特征抽取器，在特征抽取器中分别抽取正例和反例的特征，这些特征被添加到用户配置文件中，从而使配置文件能够更好的描述用户的需求，同时调整过滤器的阈值，提高整个系统的正确率和召回率。

3 过滤器实现

3.1 方法描述

过滤器在实现时采用经典的向量空间模型，和向量的余弦值的方法来计算每一篇文档与用户的需求之间的相似度。用户配置文件分为两部分，一部分为特征词，另一部分为特征词对应的权重，权重可以为正数也可以为负数，取值范围在(-1，1)，在我们的用户接口中分为“非常相关”、“相关”、“比较相关”、“比较不相关”、“不相关”、“非常不相关”等几类，分别对应着(-1，1)间的不同值，正的权重表示该特征词为用户所需的文档信息的特征，而负的权重则表示该特征词不是用户所需的文档特征，我们称为反向特征词。我们把用户的配置文件作为一个向量，同时把用户的配置文件中的词在文档中的出现频率与该文档的总词数的比

值作为另一个向量，用余弦值的方法计算这两个向量的相似度。

$$\text{Similarity}(d_j, q) = \frac{\sum_{i=1}^{n}(td_{ij} \times tq_i)}{\sqrt{\sum_{i=1}^{n} td_{ij}^2 \times \sum_{i=1}^{n} tq_i^2}}$$

td_{ij} 为第 j 个文档向量的第 i 个词权重

tq_i 为用户配置文件向量的第 i 个词的权重

我们给出一个阈值来判断该文档是否为用户需求的信息，即当相似度大于阈值时，认为是用户需求的信息，反之，则不是。初始阶段阈值为零，在阈值反馈后重新进行调整。

3.2 过滤模块的实现

由于要过滤的文本可能会很多，当用户的配置文件中的特征词数量比较大时，如何快速统计特征词在文档中的出现次数，成为影响文本过滤器效率的一个重要因素。为了加快统计速度，我们以每个特征词的首字为索引，将特征词散列在 6763 个汉字组成的散列表中，对于首字相同的特征词，我们用链表将它们组织起来。如下图所示：

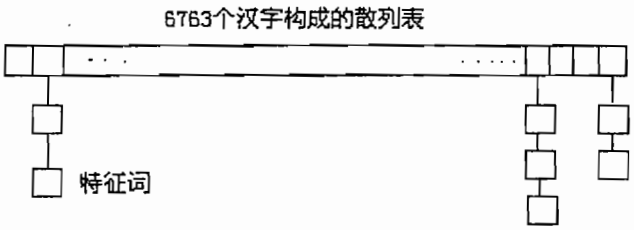

图 2 特征词索引图

在图 2 中特征词节点可以记录该词在文档中出现的次数，这样极大的提高了统计速度。

4 系统学习过程

4.1 特征抽取

一般来说，用户不可能在初始阶段就写出一个能很好表达自己需求的配置文件，因此系统应具有自动学习的功能，不断完善用户配置文件，以更贴近用户的过滤需求。在用户对过滤后的文档进行确认之后，这些文档被分为正确分类和错误分类两部分，它们都被送到特征

抽取器来抽取正面和反面特征。我们尝试了下面几种特征抽取的方法：

(1)TF（term frequency）方法

TF 的方法就是依据词汇在文档中出现的频率来描述文档的特征。一般的，文档的特征词在文档中的出现频率相对来说应该是比较高的，但大量的停用词出现频率也很高，影响特征提取。

(2) TF*IDF 方法

在信息检索中最常用的确定一个词在文档中重要性的方法是 TF*IDF 的方法。TF 即该词在一篇文档中出现的频率，IDF 称为反文档频率，一个词在越多的文档中出现它的 IDF 就越小，反之就越大。这种方法在信息检索的关键词提取中是非常有效的，但用在这里却有点不太适合，因为文档的特征词应该是出现在大部分的文档中的，因此，特征词汇的反文档频率是比较小的，从这方面来说，特征词与停用词有着一些相似的特点。

(3)DF(document frequency)方法

DF 的方法是根据词汇在多少文档中出现作为衡量该词重要性的标志，由于要提取的特征应是大部分文档都具有的词汇，这样刚好适合用 DF 的方法进行评价，存在的主要问题依然是会把一些停用词提取出来，我们通过一个停用词汇表来过滤停用词。经过试验比较，采用 DF 的方法并且过滤掉停用词来抽取特征词效果最好。在实验中发现，采用 DF 方法时是有一定的条件的，它要求训练的文档应该比较多，一般要取得比较好的效果的训练文档数量要在 400-500 篇以上。

系统学习的过程可以看作是由两部分组成：特征词的提取、特征词权重的调整。以上只是讨论了特征词的抽取，在提取出特征词后我们要给出一个权重来评价该词刻画文本特征的能力，这里我们用 DF 除以训练文档的总数作为该词的权重，如果该特征词已经在用户配置文件中出现，则把对应的权重相加后取平均值，作为该特征词的新的权重。

4.2 阈值的反馈

阈值选择应在文本流不断变化中，动态的取最优值。在系统初始状态给定一个初始阈值零，在经过对正例、反例的学习之后，用新的特征词完善用户的过滤配置文件，再计算每一个正例、反例与新的用户配置文件之间的相似度，以正例相似度的均值和反例相似度均值的中点作为新的阈值。可以按照下面的公式计算反馈阈值：

$$Threshold = \frac{\frac{\sum_{i=1}^{m} Simp_i}{m} + \frac{\sum_{j=1}^{n} Simn_j}{n}}{2}$$

其中：$Simp_i$、$Simn_j$ 分别为第 i 个正例，第 j 个反例与新的用户配置文件的相似度

m,n 分别为正例和反例的数量

在一系列有阈值反馈和无阈值反馈的对比实验中，有阈值反馈的正确率和召回率都要优于无阈值反馈的结果。

4.3 学习方法的实现

系统学习所需要的主要数据是文挡中词语的出现次数统计，以及某个词在多少个文档中出现，由于要统计文档数量较多，因此统计的效率是关键。为了能进行高效的统计我们设计了如下图所示的数据结构：

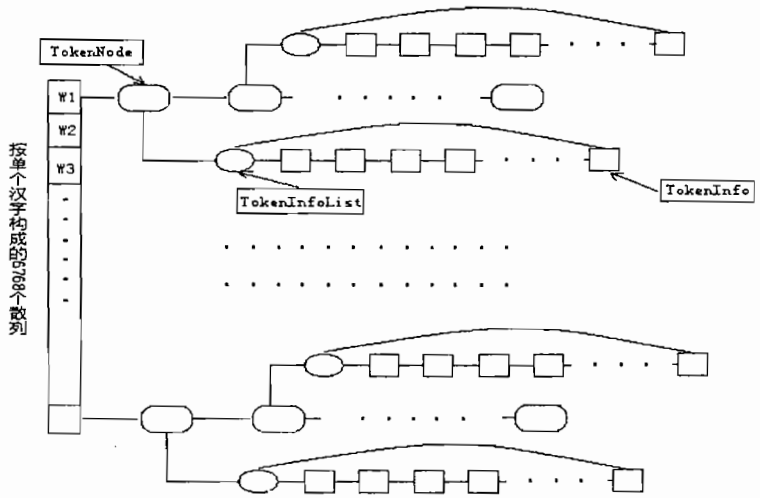

图 3 学习算法数据结构图

如图 3 所示，首先对输入的文档过滤掉停用词，然后按照 6763 个汉字建立散列表（由于在汉字表中有 5 个空缺位置，因此散列表共有 6768 个节点），对于文档中出现的词语按照其首字填入散列表中，对于具有相同首字的词语我们用由 TokenNode 构成的链表结构把它们连接起来，我们称这个链表为词语链表，同时每个词语在不同文档中出现又构成一个由 TokenInfo 构成的链表，我们称其为词语信息链表，因为它记录了该词语都出现在哪些文档中并且记录了在该文档中的出现次数。词语信息链表又由 TokenInfoList 与词语链表相连接，TokenInfoList 记录了链表结构的首尾地址，同时还纪录了这个链表结构共有多少个节点，这个节点个数就是该词在多少个文档中出现，即前面所提到的 DF。

5 试验结果

为了能较为方便地得到训练和测试语料，我们从互联网上收集了一些含有有害信息和与有害信息相关（包括批判和反对）的文档，另外在新浪网(http://www.sina.com.cn)下载了大量的体育、娱乐、教育等文章进行过滤试验，由于网站已经对文章进行了分类，这为进一步的实验提供和很多方便。

实验 1

在实际的应用情况下，有害信息常和一些反对和批判有害的信息混杂在一起，为了能尽可能的模拟真实的情况，在我们收集的语料中有害信息（包括一些色情、暴力等）文档数为216篇，另外还有一些是反对和批判这些有害信息的文档共516篇，其他包括足球1337篇、篮球968篇、娱乐668篇、经济822篇、教育579篇、生活897篇、科技713篇、汽车111篇。由于在反对有害信息的文档中也有一些有害信息的特征词，因此有害信息，无害但却相关的信息，以及无害不相关信息的分布关系可用下面的示意图中（1）近似的表示

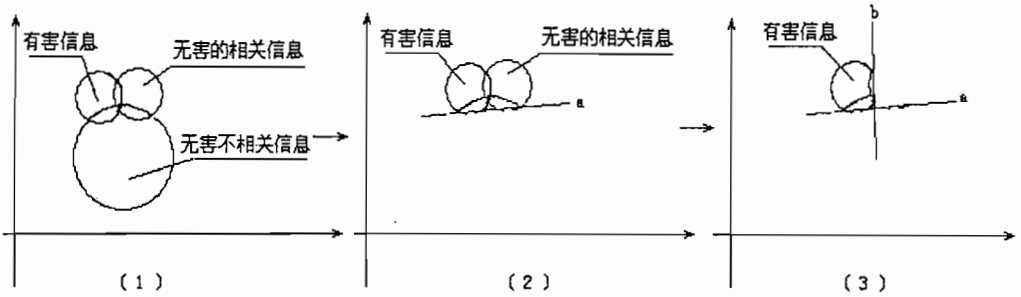

图4 分阶段过滤示意图

从上图（1）可以看出很难一次就把有害信息过滤出来，因此我们采用分两个阶段来过滤的方法。第一阶段是把和有害信息相关的文档（包括有害文档和反有害信息的无害文档）过滤出来相当于示意图中（2）用直线a把无害不相关信息划分出去，第二阶段再通过反馈学习把有害信息从相关的信息中过滤出来亦相当于示意图中（3）用直线b把无害相关信息划分出去。如果只进行第一阶段的过滤系统的正确率和召回率如下：正确率：33.50% 召回率：94.90% 由于有大量的无害相关信息混入，因此系统的正确率较低。在进行了相关反馈的学习和第二阶段的过滤后系统的正确率和召回率可以达到：正确率：63.09% 召回率：92.59%

可见，在召回率下降不大的情况下，系统的正确率提高了近一倍。为了能更全面的测试系统我们有做了如下实验。

实验 2

在汽车、足球混合实验中我们用915篇汽车、16369篇足球混合的测试语料中过滤汽车类的文章。实验结果如下：

只采用用户的配置文件，阈值取为零时的实验结果为：正确率：83.34% 召回率：43.19% 当加入相关反馈和阈值反馈后：

表1 试验1结果评价

正确率%	召回率%	DF/DOCNUM范围
49.38	93.78	[0.25, 0.5]
58.37	94.90	[0.2, 0.5]
65.90	96.39	[0.15, 0.5]
71.70	97.38	[0.1, 0.5]
73.59	97.76	[0.05, 0.5]
73.39	98.13	[0.05, 0.55]

其中 DF 为文档频率，DOCNUM 为训练的文档总数，文档频率在表 1 的 DF/DOCNUM 范围内的词语被作为特征词加入用户配置文件中，当 DF/DOCNUMD 取不同范围时会直接影响到相关反馈的特征词汇集。可见在经过相关反馈后系统的性能有较大的提高。

实验分析

在上面的实验中可以看出 DF/DOCNUM 的取值对实验结果有很大的影响，DF/DOCNUM 取值是多少时才能是系统达到最优，还值得进一步讨论。另外，在阈值反馈时，每次计算反馈的阈值并没有考虑原来阈值作用，过于强调了本次反馈的结果，可以通过一个常量 λ $(0 \le \lambda \le 1)$ 来调整新阈值与原阈值的比例：

$$Threshold = \lambda Threshold_{old} + (1-\lambda) Threshold_{new}$$

这样可以防止由于某此反馈的偏差对系统造成巨大的影响。

6 结束语

本文针对随着互连网的发展而带来的一系列有关信息内容安全的问题，提出了面向信息内容安全的过滤系统，系统运用了信息过滤和信息抽取的技术，并提出了一些改进的方法。系统虽然是面向信息内容安全的，但它同时也可以用来收集用户感兴趣的有益信息，因此有着广泛的应用前景。

参考文献

[1] Willian B. Frakes, Ricardo Baeza-Yates. Information Retrieval Data Structures & Algorithms. Prentice Hall PTR, New Jersey, 1992

[2] David D. Lewis. Feature Selection and Feature Extraction for Text Categorization. In speech and Natural Language Workshop,1992

[3] Ricardo Baeza-Yates Berthier Ribeiro-Neto. Modern Information Retrieval. ACM Press, New York, 1999