Blog 检索中的关键问题研究

摘 要

本文针对当前互联网环境及对文本情感分析技术的需求，研究了 Blog 检索中的网页信息抽取和文本情感分析问题，主要创新工作和成果如下：

第一，提出了一套高效、健壮的网页文本抽取算法。

该算法克服了主流的基于 DOM 模型的网页文本抽取算法性能的性能缺陷，首次以 SAX 接口实现了对页面框架结构信息的利用。提出了基于全局噪声信息去重的方式提取页面正文的方法。

该方法被应用在 TREC Blog06 数据集上，在将文档集规模压缩 87.5%的同时，提高相关性检索性能指标 52.5%以上。

第二，提出了基于统计模型的情感分析中的几组关键特征。

对情感分析中词汇的 N-gram 特征及其各种权重计算方法、词性特征、否定词特征和同义词扩展特征在当前情感分析领域的应用和效果进行了分析。通过词级别和句子级别的情感极性分类实验，分析了几种特征及其各种组合的应用效果，发现词性、否定词等高级文本特征在用于词级别情感分析时需要与位置信息结合，同时这些高级特征在使用基于统计的分类模型进行句子级别情感分类时效果不如单纯使用词的 Unigram 特征。

使用本文发现的特征组合，词级别情感极性分类准确率达到 88.6%，句子级别情感极性分类准确率达到 83.9%。

第三，实现了一套 Blog 观点发现系统。

该系统引入网站全局噪声信息净化网页，创造性的结合段落和篇章全文级别的检索结果，从而大幅度提高了话题相关性检索性能。在 2008 年的 TREC Blog 测试中，该系统由于表现出色被列为后续任务的基准系统。

关键词： Blog 检索 网页分割 正文提取 情感分析 特征选择

RESEARCH ON SEVERAL KEY ISSUES IN BLOG SEARCH

ABSTRACT

Facing the ever growing complexity of web pages and the increasing need of more intelligent content analysis technology, we investigated two key problems in the blog information retrieval system, which include web page content extraction and text sentiment analysis. The main innovations of this thesis are stated below:

Firstly, to eliminate the space and computational overhead of the state-of-the-art DOM based web page content extraction method, we purposed a SAX style algorithm. This fast and robust algorithm utilizes the page level template structure and site level noise block dedupe to extract the content of pages.

While being tested on TREC Blog06 dataset, it reduces the dataset to 12.5 percent of its original size and gets a 33 percent improvement on MAP.

Secondly, the feature selection problem in statistical sentiment analysis is investigated. Features including term n-gram, part-of-speech, negation and synonym expansion are tested. We found that, in word leve sentiment polarity analysis, all these advanced features need to be accompanied by position information in order to take effect, and the statistical classification model work well enough when using only unigram feature on sentence level sentiment polarity analysis. The best features combination achieved an 88.6% precision word polarity classification and an 83.9% in sentence level.

At last, we introduced a blog opinion retrieval system we developed with the techniques above. We got great improvement in topic relevance when we innovatily use the site-level noise removing technique to extract the web page content and combine the document\paragraph level relevant score. In TREC 2008 Blog Track, our system is chosen as one of the baselines for our good relevance retrieval performance.

KEY WORDS: blog retrieval, page segmentation, content extraction, sentiment analysis, feature selection

独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：2009.03.18

关于论文使用授权的说明

学位论文作者完全了解北京邮电大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属北京邮电大学。学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存、汇编学位论文。（保密的学位论文在解密后遵守此规定）

保密论文注释：本学位论文属于保密在__年解密后适用本授权书。非保密论文注释：本学位论文不属于保密范围，适用本授权书。

本人签名：_____ 日期：_____

导师签名：_____ 日期：2009.03.18

第一章 绪论

1.1 研究背景及意义

1991年8月6日，在互联网的核心技术诞生20年之后，Tim Berners-Lee发布了他的WWW项目以及第一个互联网站Info.cern.ch[126]，从此人类正式进入互联网时代。又过了20年，互联网跨越了Web 1.0和Web 2.0时期，正在向即将到来的Web 3.0迈进。互联网自身，以及整个人类社会，都发生了巨大的变化。

Web 1.0是互联网的"只读"时代。由于信息发布的门槛仍然很高，内容主要由网站集中的采集、编辑和发布。由机构和专业作者扮演作者的角色，而个人用户只作为阅读者存在。这一时期的互联网以静态的HTML网页为主，格式简单。

从1994年成立的最初以分类收藏夹形式提供网站索引服务的Yahoo![64]，到1996年上线的现代意义上的搜索引擎Google[58]，这一时期的网络信息处理系统都是以静态网页为处理对象，对网页进行主要基于相关性的Ad Hoc检索。在这一阶段，网页本身作为一个信息单元存在。

Web 2.0是互联网的"全民创作"时代。2002年，Blog和RSS开始流行。简单的编辑和发表，丰富的用户间互动使得曾经沉默的网民群体爆发出巨大的创造力。内容发布的主体从网站编辑变成了普通网民，他们开始大量发表带有强烈个人色彩的文章。至2008年8月，全球Blog数量已经过亿，每天发表在Blog上的文章超过百万[49]。

更加丰富的内容也伴随着更加丰富的信息载体。传统的HTML网页已经不足以满足用户的需求，CSS层叠样式库、JavaScript脚本语言和在此基础上出现的能实现网页局部动态刷新的AJAX技术大大增强了网页的表现力，但同时也使得网页的内容更加复杂和难于处理。

Web 2.0在带来海量数据的同时，也带来了大量新的应用需求，如：

- 基于用户评论分析的新型产品聚合网站：如Epinion[57]、IMDB[59]、豆瓣[56]这样的网站汇聚了大量的用户评论，这些评论对于了解用户对各类产品、服务或事件的评价有很大的价值。为了更好地组织利用这些评论，网站需要能自动的提供用户评论中的核心观点摘要，也需要有办法将用户的感性评论转化成量化的产品评分。
- 商业智能和电子政务：对于商业公司来说，相对于周期长、耗资大的目标

人群调研和市场调查，通过分析在线评论获得的用户对其商品的反馈意见和目标用户的信息显得更加高效和可靠。这就需要能够自动的抽取出评论中观点的持有者和被评论的对象，也需要对评论的情感倾向进行分类。而对政府来说，除了和公司一样要了解公民对其提供的公共服务的态度外，还可以通过分析潜在危险社区的言论来帮助提高社会安全保障水平。

- 传统服务的情感分析模块：电子邮件及其他在线通讯服务一直受到以Spam方式发送的各种侮辱性、攻击性的骚扰信息困扰；在线广告商希望更加准确智能的把广告投放到对其产品感兴趣的用户的屏幕上；问答类社区在充分进化后，也希望能区分寻求意见的观点型问题和寻求答案的信息型问题……

在实现这些应用的过程中，面临的最大挑战，就是高效的处理正变得日益复杂的网页，并将对文本的理解深入到语义情感层次。网页分析和文本情感分析技术正是解决这两个问题的关键武器。如果不能高效的对大量网页进行自动净化和信息抽取，则后续的信息处理则无从谈起。如果没有文本情感分析技术的支持，则无法自动的从句子、文档中发现作者的主观情感倾向、无法发现文档中观点的持有者和对象，那么就不可能满足上述的应用需求。

1.2 网页分析技术的研究现状

1.2.1 网页分析

目前多数面向Web的信息检索系统都将网页作为检索的最小单位。但网页作为一个信息载体，往往包含着许多诸如导航栏、广告、互动模块之类与网页主题无关的信息。同时，单个网页中也常包含多个不同主题的内容，应按主题切分成独立的文档被检索系统收录。未解决这一问题，网页分析技术应运而生，它综合的利用页面的HTML标记语言、网页视觉框架信息、网页文本的语义信息对页面进行分析，去除噪声，分割信息块。

1997年，为了加快网页浏览、实现关键词检索，一些研究者如Ashish [8]等人开始尝试设计Wrapper以抽取网页中的有效内容，结合数据库系统使用。这一时期的Wrapper系统往往表现为一组正则表达式的形式，针对每个不同的页面结构都需要进行重新构造。这类系统是网页分割技术的雏形。

2001年前后，链接分析（Link Analysis）技术开始在Web搜索中得到日益广泛的应用。而很快人们就发现，网页中大量的导航、推荐、广告类链接会严重影响HITS[77]和PageRank[102]之类基于链接的权威度计算算法的效果，同时也会

加重搜索引擎爬虫系统的工作负担，另外加上手持设备对小屏幕下网页展示的需求，这些都对网页分割技术的性能和效果提出了更高的要求。在这一时期，Chakrabarti[21][22][23]以分割后的网页块作为检索系统的基本单位，证明了更细粒度的检索系统可以在不严重影响召回率的同时提供更高的准确率。

1.2.2 全站模板抽取

网页模板就是网站在设计网页过程中反复使用的框架结构，例如新闻网站的新闻页面、论坛的文章、Blog 网站的日志等等都会有全站共享的导航栏、友站链接、广告推送、站长资料等内容。2003 年，Gibson[30]注意到在全站范围内大量使用同一模板生成网页的网站逐渐增多，Bar-Yossef 和 Rajagopalan[146]提出通过分析网页的文档对象模型（Document Object Model）树获取网页模板（Template），之后利用模板从网页中去除噪声的方法，在此之后基于 DOM 的方法成为网页分析技术的主流。Lan Yi[83]使用 HTML 节点的 metadata 信息建立网页的 Style Tree 进行模板提取。

1.2.3 网页内容块抽取

针对网站的模板抽取技术往往以在全站范围内净化网页噪声为主要目的。而垂直检索系统、手持设备显示系统往往要求不仅去除噪声，更要对网页上的内容进行某种筛选和重新排列。Kao 等人[42]最早开始对单个网页分别进行模板抽取，他们使用贪婪算法处理网页的结构信息，但是模板提取步骤仍然需要比较多个网页。在网页内容块分类方法问题上，Debnath 等人[112]训练分类器对 HTML 网页中的部分标签进行分类，Kao 等人[42]使用 DOM 树中节点的信息熵和链接数量作为特征，Davison[9]使用决策树分类 DOM 节点并移除噪声。

1.2.4 基于视觉信息的信息抽取

2004 年，微软亚洲研究院的 Deng Cai 等人[35]提出了利用网页的视觉信息进行网页分割的 VIPS 算法。VIPS 通过获得网页中视觉区域的相对大小、颜色、字体等视觉信息，并生成块的层次结构（Hierarchy Structure）以此为基础通过定义一系列启发式规则判断块的类别属性。该算法在对网页进行分割和属性标注上取得了良好效果，但是由于需要对网页进行部分渲染（render），导致了很大的计算开销，故而至今没有得到广泛应用。

1.3 文本情感分析技术的研究现状

文本的情感分析，近年来受到了广泛关注，其目的是判断给定文本片段所体现的作者的情感倾向和作者对特定观点所持的态度。对文本的情感分析的研究可以追溯到文献[144]，他们采用语言学特征对整个文本的整体情感进行判断。近年来越来越多的研究开始集中到文本情感分析上，主要的研究内容包括词语的语义倾向性识别、文本的主观性分析、文本的情感极性分类、观点提取等。这些研究工作可以归纳为以下几个领域：

1.3.1 主客观分类

主客观分类是将文本分成作者的主观性评论和客观性叙述两类。Finn 等人[34]论证了引入词性（Part-Of-Speech POS）特征的分类器效果比单纯使用词汇特征的方法效果好，本文稍后将对此进行试验。Wiebe 等人[136]对人工标注的语料从短语、句子和篇章的层次进行研究，主客观性的定义随标注者的不同有很大差异。近年来 TREC 测试的 Blog 检索项目一直把主客观分类作为核心任务之一，对该领域的研究起到了很大的推动作用。

1.3.2 词的极性分类

Turney 和 Littman[128]使用 AltaVista 搜索引擎[53]的“近邻”运算符获得到两个词在文本中的共现频率，以这种近邻共现关系来量度两个词的相关性。利用这种词相关性，Turyney 提出了一种类似聚类的方法，通过计算词与手工收成的极性词汇种子列表的距离关系来进行词的极性标注。此后他们引入了两种利用词语与具有明显语义倾向的种子词语之间的统计关系[128]，来自动识别词语语义倾向。

Esuli 等人[33]使用词典中对词语的解释进行训练和分类，从而判断其它词语的语义极性。Hatzivassiloglou 等人[131]提出了一种有监督学习算法，根据连接词对形容词的语义倾向的语法约束关系，由已知词语的极性，推测通过连接词与其关联的其它词语的语义倾向。Kennedy 和 Inkpen[4]设计了 General Inquirer[63]系统来查找网络评论中的情感词，并提出了 negations（否定）、intensifiers（强化）和 diminishers（弱化）三种算子来计算情感词的强度。Andreevskaia 和 Bergler[6]使用情感标签抽取程序（Sentiment Tag Extraction Program，STEP）方法，利用 WordNet 中词的同义、反义等关系和词的解释，从中提取情感词。

1.3.3 基于情感词标注的文本情感分析

Liu[87]等人使用Open Mind Commonsense数据库[48]为选择的语言特征赋予情感值，并将其归纳为六个基本类别（高兴、悲伤、愤怒、恐惧、厌恶和惊奇），通过分析带有情感色彩的词语特征来判断文本的极性。Subasic 和 Huettner[124]手工建立了一个情感类别词典，标注了词的强度（表达情感的力度）和向心度（与类别的相关程度）。Das 与 Chen[28]同样也使用了手工构造的情感词字典，识别出其中的倾向性词语，并将这些倾向词的极性累加（正面为 1，负面为-1，中立为 0）得到整个文本的极性，然后据此对文本进行情感分类（乐观/悲观/中立）。

1.3.4 基于机器学习的文本情感分析

基于机器学习方法的文本情感分析是本文的主要研究方面，下面对近年来这方面的主要研究工作做简要叙述。

Pang 等人[103]最早利用机器学习方法来解决文本情感分析问题，以在线电影评论作为语料，采用了不同的特征选择方法，应用朴素贝叶斯、最大熵、SVM 对电影评论进行分类。Ni 等人[98]将情感分类视为二分类问题，使用了朴素贝叶斯、SVM 和 Rocchio's 算法，并采用了 CHI 方和信息增益（Information Gain）进行特征选择。

在 Minqing Hu 和 BingLiu 的一系列工作中[65][66]，针对消费者对产品的评论，使用关联规则的方法抽取和挖掘产品特征，利用 1.3.2 节中提到的基于字典的方法进行词的极性标注，进而计算整个句子的语义倾向性，最后对产品评价进行分类和概述。

Bruce、Wiebe 等[14][137]利用贝叶斯分类器对句子的主客观性进行分类。Yi 等人[37]使用模式匹配的方法进行句子级的文本情感分类。Whitelaw 等人[135]提取含有形容词的词组及其修饰语作为特征，使用向量空间模型表示文档，并采用 SVM 进行分类，来区分带有正面和负面评论的文档。Lin 等人[86]把观点判别作为一个分类问题来考虑，提出了一种基于统计模型的学习方法，通过词汇特征对观点进行分类。Goldberg 和 Zhu[36]针对电影评论的等级推理问题，提出了基于图的半监督算法，较之以往采用多分类模型的方法，性能有了较大提高。Mei 等人[93]对 Blog 上主题情感分析的问题提出了一个新的 Topic-Sentiment Mixture（TSM）概率模型，同时捕捉主题和情感信息。该模型可以发现一个 Blog 数据集中潜在的主题，不需要任何领域的先验知识，该模型就可以从任何一个 ad-hoc 查询中抽取情感模型。

1.4 本文的工作及内容安排

论文的第一章介绍了本文的研究意义，并对相关研究领域的主要研究工作加以介绍，最后提出了本文的研究动因及本文的主要工作。

第二章提出了基于网页分析的Blog文本抽取算法

第三章着重分析了基于统计模型的文本情感分析中，特征选择方法对分析效果的影响。

第四章在前两章的研究成果基础上，围绕TREC Blog评测提出了一套完整的Blog观点检索系统。

第五章对全文工作进行总结，并提出下一步的工作展望。

第二章 基于网页分析的Blog文本抽取

2.1 引言

网页是互联网上信息的基本载体，它通常由 HTML 写成，通过链接（HyperLink）提供导航和对其他网页的引用功能。最初的HTML语言是标准通用标记语言（SGML）语言的一种扩展[125]，被设计来用来结构化信息，例如标题、段落和列表等。

随着互联网的快速发展，简单的 HTML 不能满足人们对页面设计越来越高的要求，微软和网景公司在浏览器之争中不断扩展 HTML 以使其更多的承担描述文档外观的功能，这令HTML标准变得相当混乱。浏览器对各种非良序（well formed）HTML的强大兼容性也使得各种不标准的写法广泛存在。

另外，过去单纯作为在线文档存在的网页几近绝迹。网站的界面设计正变得日趋复杂，页面中存在着大量诸如导航栏，广告、链接列等会大大影响信息处理精度的噪声信息。

本章讲探讨如何以一种快速、健壮的方式解析网页，并去除页面中的各种噪声信息的方法。后续内容组织如下：

第2节，对网页分析中的一些相关技术，如DOM模型等进行简单的介绍；

第3节，解释网页分析中需要完成的主要任务，这些任务的难点以及当前的主流方法；

第4节，介绍一种基于全局内容块去重的网页文本信息抽取算法

2.2 网页分析的相关技术概念

2.2.1 DOM

文档对象模型[78]（Document Object Model）是一种平台、语言无关的对象模型，它最初被用来对 XML 和 HTML 建模，同时作为一种 API 它也支持对这两种标记语言的查询（Query），遍历（Traverse）和修改（Manipulate）操作。

在 HTML 处理中，DOM 模型可以以<html>标签为根节点，将 HTML 网页按照标签间的包含关系，以每个页面标签为节点将页面解析成如图 2-1 所示的树形结构。

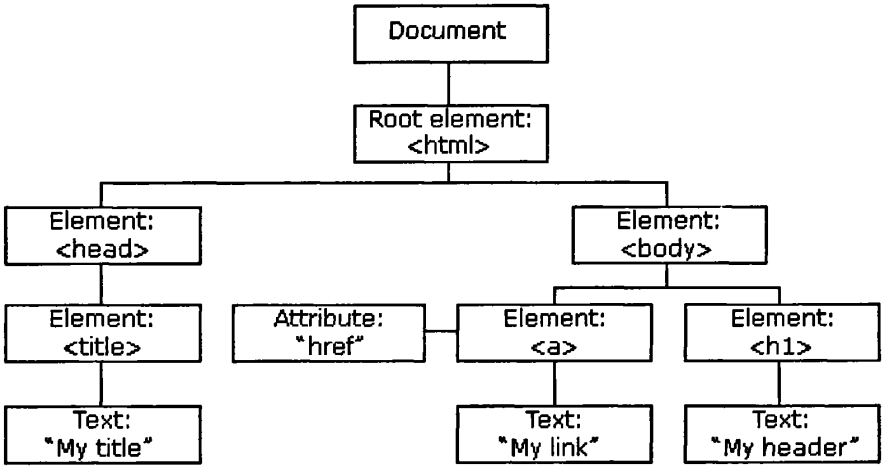

图 2-1 典型 DOM 树结构

借助 DOM，人们可以真正的将 HTML 作为一种半结构化文档进行处理。利用其层次结构，将网页分割为一个个单独的节点，其后利用各节点的标签类型、包含的文本内容等特征对其进行处理。

DOM 模型的问题在于目前大部分网页都无法满足它的良序性要求，同时对网页进行 DOM 建模的计算复杂度很高，许多功能需要借助于浏览器的页面渲染引擎实现，这使得以 DOM 为基础的算法很难应用在海量数据处理场合。

2.2.2 SAX

简单 XML 接口[133]（Simple API for XML）是一种事件驱动（Event Driven）的 XML 解析接口。与 DOM 的层次建模不同，它将 XML 文件视为一个字符串流进行解析。

相比 DOM，由于不需要在内存中建立整个 HTML 树型结构，SAX 只需要占用小得多的内存，并提供非常快的解析速度。假设文档中共有 N 个节点，则 DOM 模型的解析算法复杂度不可能优于O(N log N)，而 SAX 模型的复杂度仅为 O(N)。

SAX 模型的问题在于它无法支持对节点的多次遍历访问，也无法提供类似 DOM 中 XPath 式的节点查询操作。

2.2.3 CSS

层叠样式库（Cacscading Style Sheets）是由 W3C 定义的一种用来为如 XML 和 HTML 这样的结构化文档添加字体、间距、颜色等样式的计算机语言。它的主要目的是将文档的内容（以 HTML 或 XML 写成）与文件的显示（CSS）分割开来。这将有利于提高文件的可读性，使得文档的结构更加灵活，令作者和读者分别定义文档的显示，并且大大的简化文档的结构。

CSS 简化了 HTML 网页的内容，以一种更加清晰的方式对页面的显示进行了描述。很多基于页面视觉信息的页面分割算法如 VIPS[35]等都利用了页面的 CSS。

网页的 CSS 信息被包含在<style>标签中，与 JavaScript 的<script>标签一样，这种包含存在着很多中写法。某些写法会给网页解析造成很大的困难。

2.3 网页分析系统的主要任务

2.3.1 HTML 标签去除

网页通常被保存为一个 HTML 文件，为了提取出网页中的正文信息就需要去除网页中的大量 HTML 标签。如下图所示

I have a bike: ☐
I have a car: ☐
I have an airplane: ☐

```
<form>
I have a bike:
<input type="checkbox" name="vehicle" value="Bike">
<br>
I have a car:
<input type="checkbox" name="vehicle" value="Car">
<br>
I have an airplane:
<input type="checkbox" name="vehicle" value="Airplane">
</form>
```

图 2-2 HTML 页面及对应代码

在过去，要去除标签只需要简单的用正则表达式去掉所有“<”和“>”之间的内

容即可。但是由于浏览器对各种非良序的标签提供了良好的容错，导致各种错误格式的 HTML 大行其道。这不仅使得标签的去除变得非常困难，也增大了对网页进行 DOM 解析的难度。

2.3.2 语种识别

网页正文的语言识别是网页文本信息处理中很关键，又很容易被忽视的问题。

其关键之处在于，虽然基于统计的自然语言处理模型与语言无关，但是在特征抽取阶段，不同语言的 Tokenization 方法完全不同，例如拉丁语系语言可能需要 Stemming，True Casing 等处理，而东亚语系语言则可能需要分词。

语言识别容易被忽视的原因是，一来多数情况下信息处理系统处理的数据集都较小，语种也比较单一；二来 HTML 网页的 header 中本来就有 Language 字段来标识网页语种。但这两个假设在现实的网络应用环境中都不成立，大数据量下可能遇到各种语言，很多网页中根本没有标注 Language 字段或者标注错误。

这时就需要对经过净化的文本进行语言识别。80 年代初，密码学家 Konheim[79]提出了一种以 K-gram 字符序列为特征的识别算法，此后出现了很多与此类似的利用 K-gram 特征，基于统计模型的识别算法[20][37]。同时期，Beesley[13]通过寻找关键的功能性词汇进行语种识别，取得了很好的效果。

2.3.3 Spam 检测

对于 Spam 有两种定义。一种指主动的、无差别的大量向网络用户发布电子信息的行为，广泛存在于电子邮件、手机短信、在线论坛和 Blog 服务中。发布的内容通常以广告、诈骗和钓鱼攻击为主。另一种指为提高在搜索引擎中的权威度排名，以非人工手段制造内容的行为。这种 Spam 有时也被成为 SEO（Search Engine Optimization）。

考虑到这两种 Spam 都会产生大量与文档主题无关的噪声信息甚至噪声文档，这里将其放在一起讨论。对网页文本信息处理影响最大的 Spam 有三种：

- Post Spam：此类 Spam 属于第一类，通常发布在网页的留言本，评论栏或者论坛上。内容通常明显的与网页主题无关，有时候只有一行链接
- Link Stuffing：此类 Spam 属于第二类。由于 Google 的成功，以 PageRank[12]为代表的链接分析技术得到了广泛的关注。链接分析的假设是一个网站被越多的网站引用，则他就越重要。为了提高网站的搜索排名，有些人开始在网页内添加巨量的链接。这些链接通常位于网页底部，全都指向 Spammer 拥有的其他网站，链接的内容通常都是与网页主题无关的流行关

键词，链接以一种尽量不引起用户注意的方式存在。

- Keyword Stuffing：此类 Spam 属于第二类。通过在网页中把流行关键词重复很多遍的方式提高与该关键词的相关性以期提高与该关键词的搜索相关性。这些关键词通常不可见，被安放在宽度为 0 的 div 标签中，或者颜色被设为与网页背景色相同。

Jindal[97]等人将 Post Spam 和 Link Sutffing 的识别看作经典的二分类有监督学习任务，Fetterly[28]和 Henzinger[88]则利用了信息去重的技术识别这类 Spam。针对各类 Link Spam，Wu[10][11]的 TrustRank 和 Bencz´ur[2]的 SpamRank 算法试图通过提高链接分析的健壮性的角度消除其负面影响。

2.3.4 正文抽取

现代网页已经不同于最初以在线文档形式存在的简单风格，特别是在 Web 2.0时代的Blog网站中，页面上的正文信息甚至只会占到页面文本的极小一部分。图 2-3 是一个典型的 Blog 页面，我们可以清楚的看到，页面的绝大部分空间被网站的各类导航，广告信息占据。网页中可以提取出文本 6324 字节，而正文只有 3072 字节，占总长度的 48%。

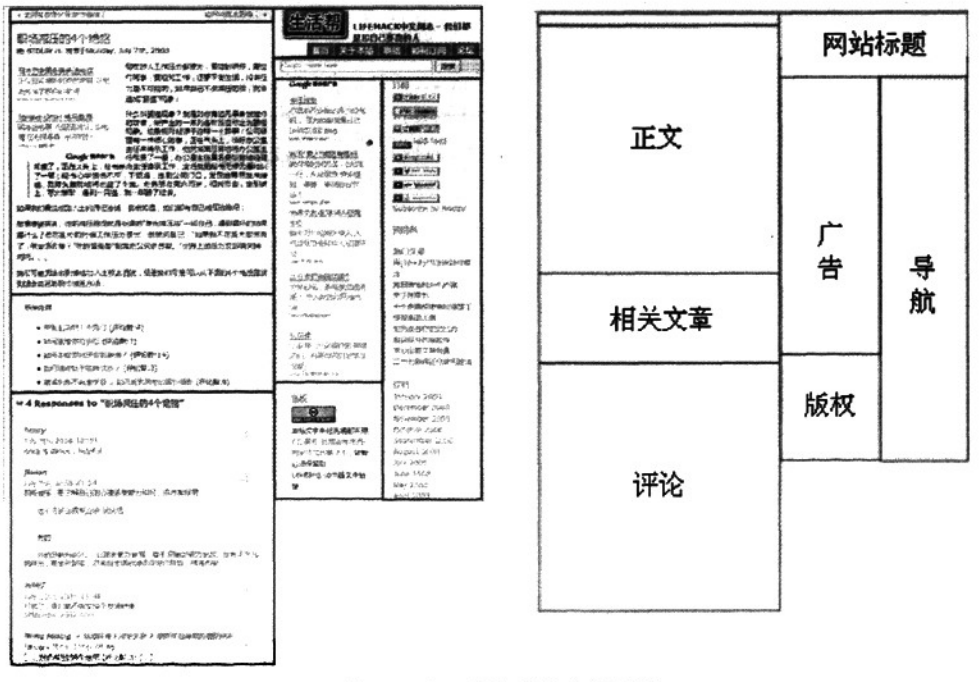

图 2-3 Blog 页面及其内容框架

正文抽取的任务即是去除与无关的冗余信息，只保留正文。Gibson[30]，Bar-Yossef[146]等人通过分析网页结构，抽取出网页间共通的自动框架结构，以此帮助噪声过滤；Chakrabarti[21][22][23]和 Deng Cai[35]等人利用 DOM 模型或视觉信息

将网页组织成节点树，再利用启发式规则或分类器方法对节点进行分类。

2.4 HTML 文本信息抽取算法研究

本节提出了一套高效的基于启发式规则的网页信息预处理流程，对我们在TREC 测试中中使用的语种识别、Spam 过滤、Blog 页面分割以及基于内容块去重的高效文本信息抽取算法进行了说明。

算法流程如图 2-4 所示，以下各节将对算法具体内容进行说

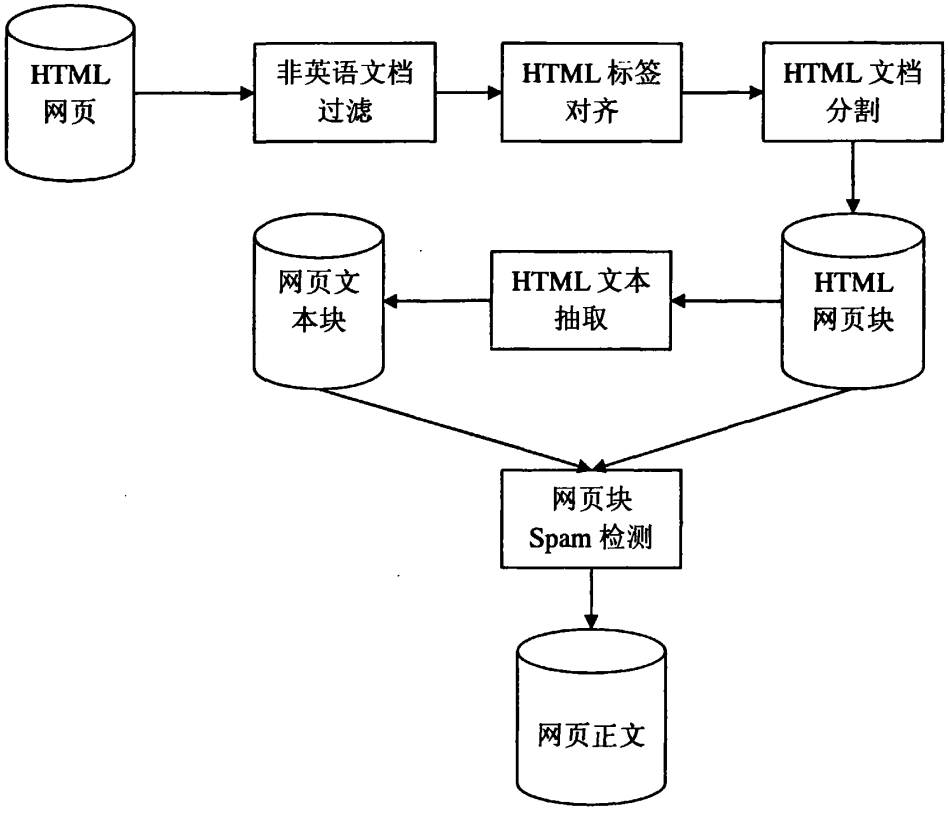

图 2-4 网页文本抽取算法流程

2.4.1 非英语文档过滤算法

尽管大量研究已经表明，以 K-gram 字符序列作为特征训练的分类器可以非常准确的判断文档的语种，但是这种分类器的困难在于训练数据集的收集。而在现实的网络环境中，可能遇到的语言又不可能限定在某个集合中。

为了解决我们在 TREC 测试中遇到的非英语文档过滤问题，我们利用了这样

一个事实：尽管用词会随着作者的不同千差万别，但是每种语言的停用词表在相应语言的文档中的出现率都会相对稳定。

我们对 Reuters-RCV1[85]文档集进行了统计。该数据集包含 1GB 英文新闻文本，共 810000 篇由 Reuters 发表的新闻文章。在统计前去除文档中所有数字字符，对全文进行 Case Folding，以空格和换行作为分隔符将全文 Tokenize。

经统计，文档中共有 179258204 个 Token，其中最常见的 30 个停用词共出现了 57400379 次，占全文的 32%。于是我们定义了这样的规则：

“若文档正文中出现停用词占全文比例低于 25%，则判断其为非英文文档”

其实现如下，

```
输入：字符串 S
输出：布尔量 IsEnglish

1  定义 StopWordList = {W0,W1,…W29}，包含英语中最常见的 30 个词
2  设单词 Wi 在字符串 S 中的出现次数为 Ci，S 中的总单词数为 N
3  Ratio = (∑_0^29 Ci) / N
4  If (Ratio > 0.25)
5      Return True
6  Else
7      Return False
```

2.4.2 HTML 标签对齐算法

为了净化 HTML 标签，提取文本信息，首先需要矫正文档中存在的不规范的 HTML 代码。常见的 HTML 不规范包括：

- 标签不闭合，如<i>word
- 包含关系不明确，如<a>
- 不规范的标签写法
  - 大小写错误，如<SCRIPT>
  - 自闭合标签未闭合， 如
  - 成对标签写成未闭合形势，如<STYLE …./>

所有这些问题中，犹以 CSS 和 JavaScript 造成的标签不对齐问题最为常见且难于处理。目前虽然存在一些开源的 HTML 净化程序如 HTML Fix[55], Html Tidy[50]等，但是他们都没法完全解决海量处理时遇到的千奇百怪的不规范格式，特别是<style>和<script>的对齐问题。为此我们提出了如下的算法：

以一个 Stack 记录 Html 中<style>和<script>标签的对齐情况，如配对不正常则在非正常开始的标签后最近的新标签处插入结束标签。定义算法 MatchTag 如下。

```
    输入：Html，TagName
    输出：HtmlPageMatched

1   初始化标签栈 Stack
2   初始化标签 TempTag
3
4   While (TempTag = ReadNextTag())
5       If (Name(TempTag) == Tag AND IsStart(TempTag) == True)
6           Push(Stack, Tag)
7       Else If (Name(TempTag) == Tag
8                AND IsStart(TempTag) == False)
9           Pop(Stack)
10  If IsEmpty(Stack))
11      Return HtmlPage
12  Else
13      Foreach Tag in Stack
14          position = Index(FindNextTag(Tag))
15          Insert(TagName, isClose = True, Index = position)
16      Return HtmlPage
```

2.4.3 网页文本抽取算法

在 HTML 标准中存在三类实体：

a) 成对但是所包含的文本在网页上不可见的；

b) 成对的文本可见的；

c) 自封闭的标签，以及转义字符

为了抽取 HTML 网页的正文，我们需要抽取出 b 类标签中的可见内容，再对 c 类标签进行一些替换，例如将<p>和
替换为字符'\r\n'，将<和<替换成字符'<'。

为此定义算法 CleanHtml 如下

```
输入：Html
输出：Text

1  定义列表 MuteTagList = {t0, t1,…tn,}，包含所有 a 类标签
2  定义字典 Dictionary，字典中的 Key 为在文档中宽度不为 0 的自封
   闭 HTML 标签，Value 为标签对应的字符。
3  SilentCount = 0
4
5  While (TempTag = ReadNextTag(Html))
6      If (TempTag in MuteTagList)
7          If (IsStart(TempTag))
8              SilentCount++
9          Else
10             SilentCount–
11         Continue
12     If (IsSelfClosed(TempTag))
13         Append(Text, Dictionary[TempTag])
14     Else
15         Append(Text, ExtractText(TempTag))
16 Return Text
```

2.4.4 网页文档分割算法

HTML 文档分割的目地是尽量的按照网页上的布局把全文分割成块，以支持对网页更细粒度的处理。早期 HTML 使用<p>、<li>等标签标识文档段落，后来人们又转向使用<table>格式化页面，于是<tr><td>标签反倒变成最有用的分割标志，随着 CSS 的出现，目前的趋势正在转向用 标签标识文档段落。

图 2-5 是一个典型的 Blog 页面 div 层次结构图，提取自图 2-3 中的网页。其中的 div 标签和网页内容的对应关系如下。目前的 Blog 架站程序只有有限的几种，经过长时间的发展已经相当成熟，在这类架站程序中生成的页面中使用 标签切割网页内容是可行的。

```
<div id="container">
  <div id="left">
    <div id="content">
      <div class="navigation">
        <div class="clear"/>
      <div id="post-471" class="post">
      <h3 id="comments">
      <ul class="commentlist">
      <h3 id="respond">
      </div>
    </div>
  <div id="sidebar">
    <div id="header">
    <div id="search">
    <div id="tag">
      <div id="side-bottom"> </div>
    </div>
  </div>
<div id="footer">
  <div class="copyright">
  </div>
```

Container	主题框架
Left	左侧框架
Content	内容框架
Navigation	导航栏
Post-471	正文
Sidebar	侧边栏
Header	侧边栏标题
Search	搜索栏
Tag	Tag 云
footer	版权页

图 2-5 典型 Blog 页面 div 层次结构

在 2.2.1 节已经分析过 DOM 模型的效率问题，由于基于 DOM 模型的算法的内存占用和计算量过大，有些时候甚至需要在内存中调用浏览器的 COM 对象以支持某些高级操作所以无法胜任海量网页的处理情况。而 SAX 模型又很难对网页进行全局处理。为了克服这个问题，我提出了一种基于 SAX 的页面切割算法。在实现高效处理的同时，按照网页的块层次对网页进行切割。

```
<div id="container">
  </div>
<div id="left">
  </div>
<div id="content">
  </div>
<div class="navigation">
<div id="post-471" class="post">
<div id="content">
  <h3 id="comments">
  <ul class="commentlist">
  <h3 id="respond">
 </div>
```

图 2-6 分割后的页面 div 结构

该算法的基本假设是将网页的 2D 的 div 树向下压平成线性结构。图 2-6 所示便是图 2-5 中的部分 div 结构被压平之后的样子。注意 Content 块出现了两次：它第一次出现的时候包含的是从图 2-5 中 Content 块开始，到 Navigation 块之间的部分，而第二次出现则是包含了从 Post-471 块结束到 Content 块结束的内容。

实现这一转换的 SegmentHtml 算法如下，和上面的标签匹配算法一样利用标签栈来模拟树的递归结构。默认的 SplitterList 包含 和<body>标签，在处理 Web 1.0 风格页面时也可以再加入<table>、<tr>和<td>标签。

```
    输入：Html，SplitterList
    输出：BlockList

1   初始化标签栈 Stack
2   TempTag = ""
3   Position = 0
4   BlockList = []
5
6   While (TempTag = ReadNextTag())
7       If (Name(TempTag) in SplitterList)
8           If (IsStart(TempTag))
9               If (IsEmpty(Stack))
10                  Push(Stack, TempTag)
11              Else
12                  Block = Html[Postition:Index(TempTag)]
13                  Append(BlockList, Block)
14                  Push(Stack, TempTag)
15          Else
16              Block = Html[Position:Index(TempTag)]
17              Pop(Stack)
18          Position = Index(TempTag)
19      Else
20          Continue
21  Return BlockList
```

2.4.5 Spam 检测算法

在 2.3.3 小节中我们已经列出了常见 Spam 行为的一些特征，本节我们将结合这些特征和对 Spam 文档的数据分析提出一系列启发式规则，用来帮助我们进行判断。

Keyword Stuffing 是对正文抽取最有害的一类 Spam 行为，我们主要针对这类 Spam 进行处理。Ntoulas[1]等人对 Microsoft 的 Live Search[61]搜索引擎数据进行了统计，发现一半以上的网页正文少于300词，只有12.7%的网页正文多于1000词，当文档长度达到 2000 词时该文档是 Spam 的概率超过了 55%。但是在 Ntoulas 的统计中他们也发现 Spam 概率和文档长度虽然有很强的相关性，但是其概率有

很大的波动，单纯以词作为分类依据的Spam分类准确率很难超过60%。再考虑到Keyword Sutffing的Spam文档通常都不会在Keyword之间加上标点，结合“长度”和“无标点”这两个性质我们提出第一条启发式规则：

Rule 1. 如果文档中出现超过300词的无标点句子，判断其为Spam

当前的多数搜索引擎都充分利用了网页的半结构化特征，对不同字体，位置出现的内容在计算相关性时赋予不同的权重。注意到这一现象，部分人把网页的<title>字段作为Keyword Stuffing的目标。统计Yahoo Research提供的WebSpam-UK2007数据集[139]我们发现，当<title>字段长度超过10的时候有超过90%的网页是Spam页面。为此定义规则如下：

Rule 2. 如果文档的标题长度超过10个单词，判断其为Spam

另外一类Keyword Stuffing连成一个超长的单词，例如freemp3, vediodownload等。这样做的原因是有不少用户会无意中忽略单词间的空格，尽管现在的主流搜索引擎都提供了关键词修正的功能，但还是会首先按照用户的关键词检索。这类Spam的目标就是这类因为手误造成的错误搜索。统计Reuters-RCV1数据集得到英文文本的平均词长为4.5 bytes，于是有了第三条规则：

Rule 3. 如果文档平均词长超过8或小于3，判断其为Spam

针对Keyword Stuffing大量堆积热门关键词，而这类关键词不可能是语言中的高频词汇的特点，我们可以获得两个特征。一个是高频词在文档中的所占的比例，另一个是文档对高频词表的覆盖率。结合对Reuters-RCV1数据集[85]的统计，定义规则如下：

Rule 4. 如果前500的高频词占文档长度不超过15%，判定其为Spam

Rule 5. 如果文档的词汇覆盖前500的高频词不到60%，判定其为Spam

另一类在Blog中常见的Spam是Post Spam。针对这类Spam往往只在Comment中留下一条链接的特征，我们首先对整个文档进行切割。将每个Comment作为独立的单位进行处理，判断这条Comment中的锚文本(Anchor Text)在Comment真文中的覆盖率。这样分别处理也避免了因为个别Comment中的各种Spam现象误删整篇Blog的情况。定义规则五：

Rule 6. 如果文档中只有一条链接，Html块中锚文本覆盖率超过80%，或全文锚文本覆盖率超过20%，判断其为Spam

2.4.6 Blog网页的正文抽取算法

为了准确的提取出网页中的正文信息，人们已经做了大量的工作。本文提出

了三种不同的网页正文提取方法，并对其性能和效果进行比较。这些方法充分利用了 Web 2.0 网站都是由大量经后台程序生成的相同结构页面组成的性质，通过分析网站的模板来进行正文抽取。

2.4.6.1 基于 DOM 树的方法

对于每个 Blog 页面，都可生成一个一棵如图 2-7 所示的 DOM 树。同一个 Blog 中获得的 DOM 树都应该有相似的结构。那么显然只要我们获得了这个框架结构，就可以按图索骥的按照这个结构来获得目标正文。

实现这一算法的关键是对树的相似性进行计算，并抽取两棵树的最大共同子树。已知的最佳算法[132]复杂度仍然达到$O(n_1n_2 + {l_1}^2 + {l_1}^{2.5}l_2)$，其中 n_i 和 l_i 分别为树的大小和节点数量。再考虑到建立 DOM 树的高昂开销，这样的算法显然只能应用在实验环境

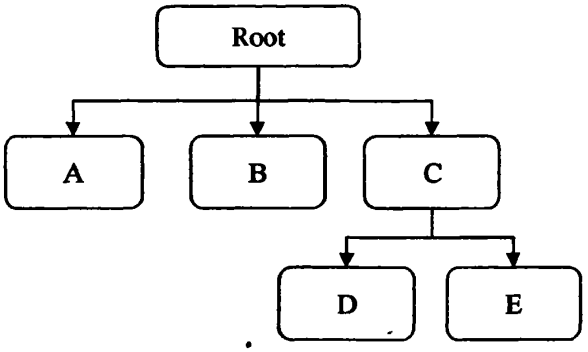

图 2-7 DOM 树图

2.4.6.2 基于序列的方法

树的最大共同子树算法复杂度太高，DOM 难以建立。但是如果能把二维的树形结构转化成一维的数列，那么就可以大大降低算法的复杂度。在不影响正文的情况下，我们可以利用 2.4.3 中提到的 HTML 文档分割算法，把图 2-7 的树，变成如图 2-8 所示的文本块序列。

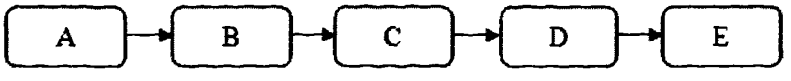

图 2-8 序列化 DOM 树

对两条序列应用最大子序列（Longest Common Subsequence）算法的复杂度为$O(l_1l_2)$，其中 l_i 为序列 i 的长度。可以看到在该方法下，避免了建立 DOM 树的开销，同时算法复杂度也大为降低。

该方法的问题是以序列方式抽取的框架损失了 DOM 树中的继承关系。经过

一点实验会很容易的发现，从同一 Blog 中获得的网页，即使他们看上去非常相似，序列的长度却可能因访客评论等页面附加信息的存在而变得非常不同。所以从序列抽取出来的框架并不能完整的过滤所有页面的噪声信息。

2.4.6.3 基于全站文本块去重的方法

利用网站模板信息进行信息抽取的思路存在计算复杂度高，抽取精度难以控制的问题。注意到这个问题，我采用了另外一种利用同网站网页过滤噪声的方法，即对全站网页进行分割，再对分割出来的内容块进行累加操作以寻找出现次数较多的内容块。经过分析我们发现，这类高频内容块主要有两个来源：网站模板噪声、和访客的 Spam 评论。我们利用这个噪声文本库对 Blog 网页进行处理，收到了极好的效果。

算法 CleanSite 如下，

```
    输入：HtmlList
    输出：TextList

 1  AllBlocks = []
 2  TextList = []
 3
 4  Foreach Html in HtmlList:
 5      Foreach Block in SegmentHtml(Html):
 6          Append(AllBlocks, MD5(CleanHtml(Block)))
 7  BlackList = FindFreqBlock(AllBlocks, Threshold = 2)
 8
 9  Foreach Html in HtmlList:
 10     Foreach Block in SegmentHtml(html):
 11         If (MD5(CleanHtml(Block)) in BlackList)
 12             Continue
 13         Else
 14             Append(TextList , CleanHtml(Block))
 15
 16 Return TextList
```

第三章 基于统计模型的文本情感分析

3.1. 引言

大量的主观性地，带有个人情感色彩的在线文档是Web 2.0时代的主要特点。另一方面，“某人对某事物的态度”又是大众信息需求中很大的一部分。据文献[73][74]统计，80%的互联网用户层搜索过产品评论，87%的用户在消费时会受到在线评论的影响，80%的Blog作者曾经撰写过产品评论[49]。

用户对产品评论信息的需求产生了不少相关的在线服务，同时也大大刺激了自然语言处理领域对文本情感分析的研究。早在七八十年代，Carbonell[70]和Wilks[144]就分别对文档的主观性和观点分析进行了研究；到九十年代末，该领域的演就取得了一些进展[90][91][134]但并没有获得很多关注；直到二十一世纪初，随着基于统计模型的机器学习方法在信息检索和自然语言处理领域的大规模使用和Web 2.0带来的大规模数据以及广泛的应用场景的出现，这一领域才迎来了发展的高峰。

目前，基于统计模型的文本情感分析方法是研究的主流。它和传统的文本主题分类问题有很多相似之处，许多模型和方法也都是在文本分类领域首先出现后再被应用到情感分析领域来的。两者之间的主要区别在于，为了更好的对人类细微的情感表达进行处理，情感分析需要引入更多的语法、语义上的先验信息。因此本文将主要关注文本分析中的特征选择问题，探究各种语言特征以及权重计算方法在情感分析场景下的应用效果。本文结合常用的情感分类模型，对词的Unigram、词性、否定词特征和不同的权值调整方法及位置信息的使用对情感分析造成的影响进行了分析。

后续内容组织如下：

第2节，介绍相关的研究工作

第3节，介绍在文本分类中广泛应用的两个统计模型

第4节，对各种特征选择方法在最大熵模型下的应用进行分析

第5节，给出在词级别和句子级别情感极性分析下各特征及组合的分类效果实验和结果分析

3.2. 基于统计模型的文本分类技术

文本情感分析问题可以看做一系列特殊的文本分类任务，其类别被定义为“主观/客观”、“正面/负面”，分类的对象可以是词、句子或整篇文档。传统的文本分类问题已经有很多年的研究基础，许多模型算法的都达到了很成熟的应用水平。因此在探究文本情感分析问题之前，有必要简单介绍一下文本分类领域的成果。

在传统的文本分类中，分类系统使用标注好的训练样本训练出分类器，按照事先定义的分类体系对文本进行标注。典型的基于概率统计的文本分类处理流程如图 3-1 所示，

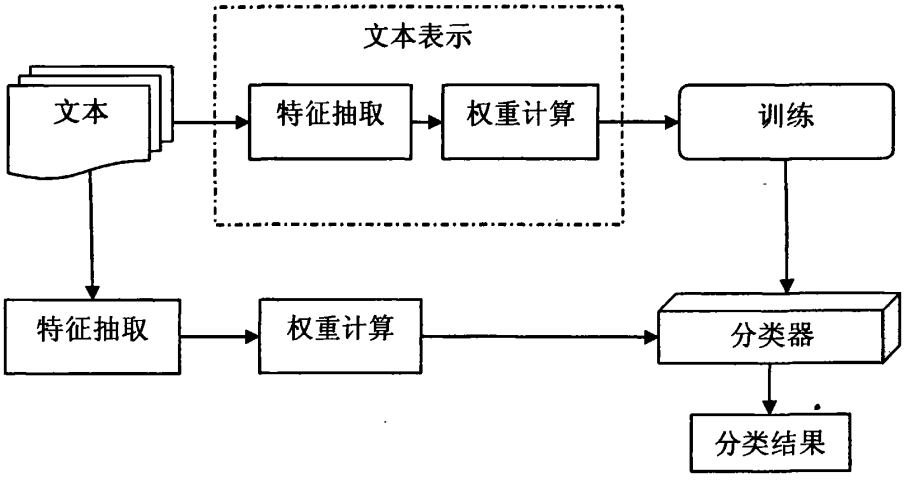

图 3-1 经典文本分类系统

3.2.1 文本的表示和向量空间模型

为了让计算机系统处理文本，需要把自然语言转化成计算机可以识别处理的模型表示。在文本分类中，最常用的文本表示模型就是向量空间模型（Vector Space Model, VSM）。该模型由 Salton[114]等人在 70 年代提出，在当前的机器学习和信息检索领域中有非常广泛的应用。

在向量空间模型中，文本空间 D 被看作是由一组正交词汇向量张成的向量空间，其中的每个文本 d 都被映射到向量空间 D 中，表示为向量 $d=[(t_1:w_1),(t_2:w_2),\dots,(t_N:w_N)]$，其中$t_k(k=1,\dots,N)$为文档空间的一个特征，$w_k$为$t_k$的权重。

在文本的表示过程中，特征抽取从文本中寻找并抽取出t_k，权重计算则赋给t_k一个合适的权重w_k。

3.2.2 特征抽取

特征抽取解决的是特征的选择和获得这一系列的问题。

在特征抽取中包含着很多复杂繁复的任务，例如上一章中网页的正文提取，对中文文档的分词、对英文文档的 Tokenization 和 Stemming、文档的词性标注、短语识别、停用词移除等工作都在特征抽取过程中完成。

3.2.3 特征选择

由于自然语言的复杂性，人们往往希望尽可能全面的收集其特征信息，Unigram、Bigram、Trigram、词性、命名实体标注，往往使用的特征越多获得的性能就越好。但特征的增加也会造成维数灾难，极大的增加分类器训练和分类时的开销。

因此，选择哪些特征来表示文档就成为在文本分类问题上一个非常关键的问题，它将极大的影响分类器的性能和准确率，其重要性不亚于机器学习分类器模型的选择。

除了利用先验知识对分类问题本身进行分析选择特征外，人们也使用一些统计学工具来帮助特征选择。常见的特征选择方法有互信息（Mutual Information），信息增益（Information Gain）和χ^2统计[141]等几种。

3.2.4 权重计算

文本被表示为模型后，整个文本往往转化为一个高维向量。由于不同的特征对文本的代表性不同，就需要给这些特征赋以权值。不同的特征计算方式往往对应不同的特征。

常用的文本特征计算方法有布尔权重、词频权重、TFIDF 权重（Term Frequency and Inverse Document Frequency）权重、信息熵权重等。更进一步的，利用主成分分析、线性鉴别分析和奇异值分解等技术对向量空间进行变化，衍生出了以潜语义分析[33]（Latent Semantic Analysis, LSA）为代表的一类特征抽取方法，这些方法在文本分类信息检索等领域被广泛使用。

3. 3. 分类模型

文本表示成向量之后，既可以用基于统计模型的分类模型进行训练和分类。经典的文本分类算法包括朴素贝叶斯（Naïve Bayes, NB）[84] [92][113]、k 近邻（k-Nearest Neighbor, KNN）[43]、boosting[117]、规则学习[25]、人工神经网络（Artificial Neural Network, NNet）[111]、线性最小平方拟合（Linear Least Square Fit, LLSF）[140]、最大熵[40][99] 、支持向量机（Support Vector Machine, SVM）[36] [72]等方法。Sebastiani[118]和 Yang[142]等人分别对多种常用的统计学习方法在文本分类上进行了对比研究。一般认为，朴素贝叶斯方法最为简单，并且在某些应用中已能达到较高性能；而 SVM 方法总体上分类精度较好。以下将简单对最常用的两种分类器，既朴素贝叶斯和最大熵模型进行介绍。

为了下文表述方便，首先对文本分类问题做一下模型化表述。

文本 d 的类别标记为 c，其一个文本 d 的类别标签记为 c，其文本特征以向量形式表示为：

一个文本d的类别标签记为c，其文本特征以向量形式表示为：

$$d = [(t_1: w_1), (t_2: w_2), \dots, (t_N: w_N)] \qquad \text{式（3-1）}$$

其中$t_k (k = 1, \dots, N)$为文档空间的一个特征，w_k为t_k的权重。判断文本d的类别标签即为求解下式最优化问题：

$$c^* = arg\, max_c\, P(c|d) \qquad \text{式（3-2）}$$

因此，建立分类模型的直接目的在于估计后验概率 *P(c|d)*。

3.3.1 朴素贝叶斯模型

朴素贝叶斯模型（Naïve Bayes Model）是一种最基本的生成模型（Generative Model）。与下文将介绍的判别模型（Discrimitive Model）不同，生成模型不是直接估计后验概率 *P(c|d)*，而是分别对条件似然概率 *P(d|c)*和先验概率 *P(c)*进行估计，再利用贝叶斯进行转换，得到后验概率

$$P(c|d) = P(d|c) \times P(c)/P(d) \qquad \text{式（3-3）}$$

P(d)对于目标文档 d 是一个常量，不会影响对 c^*的求解，因而可被略去。于是式（3-3）转化成求解最优化问题式（3-4），

$$c^* = arg\, max_c\, P(c) \times \prod_{k=1}^{n} P(t_k|d)^{w_k} \qquad \text{式（3-4）}$$

上式中的类别的先验概率 *P(c)*和条件似然概率 *P(d|c)*可以采用最大似然估计方式获得。

朴素贝叶斯模型的最大优点在于其实现非常简单。尽管它的条件独立假设在现实文本中通常并不成立，但是这类分类器仍然可以达到非常理想的效果。它的主要问题在于同时需要对 *P(d|c)*和 *P(c)*两个参数进行最优化，再加上从实验样本

统计出来的$P(c)$往往与真实情况有一定差距，这使得它的准确率较基于相同独立假设的最大熵模型低。

3.3.2 最大熵模型

3.4.2.1 最大熵原理与应用背景

最大熵（Maximum Entropy）原理源自统计物理学，其基本原理为：当需要对一个随机事件的概率分布进行预测时，所做的预测应当满足全部已知的条件，而对未知的情况不要做任何主观假设。所谓“不要把鸡蛋放在同一个篮子里”，这样做出假设时，概率分布最均匀，预测的风险最小，而此时概率分布的信息熵最大，因而这种模型得名“最大熵”。简单地说，就是要保留全部的不确定性，将风险降到最小。

最大熵原理的实质就是，在已知部分知识的前提下，关于未知分布最合理的推断就是在所有符合已知知识的推断中，最不确定最随机的那个。任何其他选择都意味着增加了其他的先验知识，这些知识与我们已经观测到的现象无关[34]。

最大熵模型在形式上是最漂亮的统计模型，而在实现上是最复杂的模型之一，如何构造最大熵模型，即最大熵模型的训练方法，一直以来是最大熵原理实用性的主要瓶颈。九十年代， Della Pietra 兄弟等研究员，提出了 IIS 算法（Improved Iterative Scaling，IIS）[37]，使得最大熵模型的训练时间降低了两个数量级，并首次将最大熵模型应用于自然语言处理领域[38]。同时期， Ratnaparkhi 将最大熵模型成功的应用于自然语言处理中的词性标注和句法分析问题[39][40]，这是首次在实际信息处理应用中验证了最大熵模型的优势。此后，在自然语言处理、文本信息处理领域中的语言建模[41][42]、文本切分[43]、指代消歧[44]、文本分类[18]、中文分词[45]等多个方面，最大熵模型都取得了非常显著的应用成果。

3.4.2.2 最大熵建模描述

与朴素贝叶斯模型不同，最大熵模型直接对后验概率$P(c|d)$进行估计，是一种典型的判别式模型(Discrimitive Model)。

给定训练文本集合 D 及文本所属的类别标签集合 C，每一个文本$d \in c$看作是一个样本，已知其特征向量$d = [(t_1: w_1), (t_2: w_2), \ldots, (t_N: w_N)]$及类别标签 c，则构成最大熵模型的训练样本集合$\{(d_1,c_1),(d_2,c_2),\ldots,(d_i,c_i),\ldots(d_{|D|},w_{|D|})\}$。对于每一个样本$(d,c)$，通过在训练样本集合上的最大似然统计，可以得出每个样本的实验概率分布，记为：

$$\tilde{P}(d,c) = \frac{\#(d,c)}{|D|} \qquad \text{式（3-5）}$$

其中，*#(d,c)*为样本*(d,c)*在训练集中的统计频数。而对于每个特征项t_k，与所有类别标签相结合，分别构成一个特征指示函数$f_k(d,c)$。在传统的最大熵中$f_k(d,c)$通常定义为一个布尔函数，例如，在文本主观性分类中，“，然而”是一个显著的主观性语义的转折句式结构，对应的特征函数可表达如下：

$$f_k(d,c)=\begin{cases}1 & d\text{中出现“，然而”结构，且类别}c=\text{"主观"}\\ 0 & \text{其它}\end{cases} \qquad \text{式（3-6）}$$

每个特征函数$f_k(d,c)$都与样本*(d,c)*相对应，根据式（3-5）中定义的样本的实验分布，可以计算每个特征$f_k(d,c)$的实验分布的期望值为：

$$\tilde{E}(f_k)=\sum_{d\in D,c\in C}\tilde{P}(d,c)f_k(d,c) \qquad \text{式（3-8）}$$

从训练样本数据中得出的各个特征项的实验分布期望值，即为建模中的已知经验知识。最大熵思想中的核心思想之一是对已知知识的充分满足，因此，要求模型估计出的每个特征项的期望值，与训练样本集合中相应的实验分布期望值保持一致，通过如下约束等式表示：

$$E(f_k)=\tilde{E}(f_k) \quad \forall f_k \qquad \text{式（3-10）}$$

而最大熵思想中的另一个核心思想是对未知信息不作任何主观假设，即要模型保持最多的可能性，选择最均匀的分布。则条件概率模型$P(c|d)$的均匀性可以表示为：

$$H(P)=-\sum_{d\in D,c\in C}\tilde{P}(d)P(c|d)\log P(c|d) \qquad \text{式（3-11）}$$

最大熵建模中最优化问题的求解方法，不是本文讨论的重点。在此略过。

3.4. 特征选择和权值计算

3.4.1 N-Gram

在情感分类的生成模型中，假设文本是由一个作者作为信息源生成的，这可以看成一个马尔可夫随机过程。那么将文本表是为一个词序列的话，这个此序列就形成一个马尔可夫链。该马尔可夫链的每个观测状态都是一个词语，显然是一种离散马尔可夫随机链。

根据马尔可夫链性质，每个词的出现概率应该只与他前面出现的 N 个历史词项有关，这就是所谓的 N 阶马尔可夫性。根据所考虑的历史词项数量不同，既 N 的取值不同，存在集中常用的 N-Gram 特征：

Unigram（N=0），假设词语序列中各个词项相互独立，与其前面出现的所

有词项无关；

Bigram（N=1），假设词语序列中每个词项的出现只与其前面一个历史词项相关。

Trigram（N=2），假设词语序列中每个词项的出现与其前面两个历史词项相关。

随着N-Gram特征的马尔可夫性阶数上升，生成的向量空间文本表示模型的阶数会快速暴涨。Trigram以上的特征将导致非常严重的数据稀疏，对训练样本的数量有非常高的要求。因此在文本分类问题上人们通常只使用Unigram或Bigram特征。

在文本分类领域，对于使用高阶N-gram特征一直存在争议。Pang等人[15]发现仅使用unigram特征的分类器在进行情感极性分类时准确率高于bigram。但Dave等人[82]稍后在不同实验数据集上发现适当的使用bigram和trigram特征可以获得比unigram更好的表现。

3.4.2 Unigram及其权重计算

可以说，使用文本中的词汇是情感分析中最简单、最基本也最重要的分类特征。同样，针对词汇的Unigram特征进行权值计算的研究进行的也最多，最充分。以下介绍几种最常见的Unigram特征权值计算方法。

为了表述方便，对文本的Unigram特征形式化的描述如下：

文本集合D中，d_i表示集合中的第i篇文本，t_k代表集合中的一个特征词项$tf(tk, d_i)$表示词项频率，定义为词项t_k出现在文本d_i中的次数；$df(t_k)$表示文本频率，定义为文本集D中，出现过词项t_k的文本数；表示文本集合中所有文本的总数；以$w(t_k, d_i)$表示词项t_k在文本d_i中的权重值，则各种文本特征权重定义如下：

1. 布尔权重

布尔权重是最简单的加权方式，如果文档d_i中出现了词汇t_k，则$w(t_k,d_i)$为1，若没有出现则$w(t_k,d_i)$为0，既：

$$w(t_k, d_i) = \begin{cases} 1 & tf(t_k, d_i) \\ 0 & tf(t_k, d_i) \end{cases} \qquad \text{式（3-12）}$$

2. 词频权重

布尔权重只能表现词汇出现与否，但是不能表示词汇出现的频率。在此模型下所有词汇的重要性相同。词汇权重假设一个词汇在文档中出现的次数越高则越能代表文档，既：

$$w(t_k, d_i) = tf(t_k, d_i) \qquad \text{式（3-13）}$$

3. 归一化词频权重

使用词汇权重虽然强调了高频词汇对文档的重要性，却忽略了文章长度造成的影响。假设文档 d_0 中出现了词汇 t_0 三次，同样文档 d_1 中也出现了词汇 t_0 三次，但是 d_0 的长度是 d_1 的十倍。在此情况下显然 t_0 对文档 d_1 的代表性应该强于文档 d_0。更进一步做假设，如果存在文档 d_0 和由 d_0 原样复制三遍形成的文档 d_1，那么两个文档的语义信息其实完全相同，其特征向量也应相同，但是直接使用词汇权重不足以表达这种区别，因此引入了归一化词频权重以消除文章长度造成的影响，既：

$$w(t_k,d_i)=\frac{tf(t_k,d_i)}{\sum_{t_k} tf(t_k,d_i)} \qquad \text{式（3-14）}$$

4. TFIDF 权重

以上集中算法都把文档作为彼此独立的单位进行考虑，并且引入整个文档集合作为整体的特征。在对文档进行比较的时候，我们关注的是那些造成文档彼此不同的、特别的词汇，而不是在所有文档中广泛出现的所谓停用词。关于词频统计的假设是词汇 t_0 在文档 d_0 中出现了越多，它对 d_0 就越有代表性。但是同时如果词汇 t_0 在整个文档集 D 的每篇文档出现的都很多，它就不如一个在文档集 D 中出现的很少，却唯独在文档 d_0 中多次出现的词汇 t_1 来得那么有代表性。于是人们引入一个新的全局文本词频[26][115]（Documents Frequency）参数 $df(t_k)$，将其和词频参数 $tf(t_k,d_i)$结合以作为词汇的权重。

标准的 TFIDF 计算形式如下：

$$\cdot \quad tfidf(t_k,d_i)=tf(t_k,d_i)\times log\left(\frac{|D|}{df(t_k)}\right) \qquad \text{式（3-15）}$$

TFIDF 是信息检索中最常用的算法，它在计算文本相似度上表现很好。但是在用来作为文本分类的参数时有时不如简单的布尔权重或加权词频。本文中使用如下的平滑后的 TFIDF 计算方法：

$$\mathrm{tfidf}(t_k,d_i)=\sqrt{\mathrm{tf}(t_k,d_i)}\times\log\left(\frac{|D|}{\mathrm{df}(t_k)}\right) \qquad \text{式（3-16）}$$

上式中，对原始词频 $tf(t_k,d_i)$开方处理已对高频此进行平滑，以获得一个较为平衡的特征词项权值分布。同时，如同在之前许多研究中所采用的方法一样[119]，为了使词项权重与文本长度无关， 本文中在每个文本 d_i 内，对各个特征词项的 TFIDF 值进行余弦归一化处理，使最终得到的 TDIDF 权重值在[0，1]区间内。如此得到的平滑公式如下所示，TFIDF 公式的其它变体见文献[96][120][121]

$$w(t_k,d_i)=\frac{\mathrm{tfidf}(t_k,d_i)}{\sqrt{\sum_{t_k}(tfidf(t_k,d_i))^2}}=\frac{\sqrt{tf(t_k,d_i)}\times\log(\frac{|D|}{df(t_k)})}{\sqrt{\sum_{t_k}(\sqrt{tf(t_k,d_i)}\times\log(\frac{|D|}{df(t_k)}))^2}} \qquad \text{式（3-17）}$$

5. 信息熵权重

IDF(Inverse Document Frequency)并非唯一考虑到文档集全局信息的权值计算方法，在这一问题上信息熵权重提供了更漂亮的数学模型。在信息论中对信息熵有严整的定义，只在一篇文档中出现的词汇其信息熵为1，在所有文档中都出现的词汇信息熵为0,其它词汇都介乎0和1之间。与单纯累加文档集频率的IDF不同，计算词汇在文档集中的信息熵还考虑到了词汇的分布情况。信息熵的计算公式如下

$$Entropy(t_i) = 1 + \sum_{i=1}^{N} \frac{p_{ij} * log\, p_{ij}}{log\, N}$$

$$p_{ij} = \frac{tf(t_i,d_j)}{\sum_{j=1}^{N} tf(t_i,d_j)} \qquad \text{式（3-18）}$$

在信息检索领域，基于词频特别是TFIDF的权重计算方法获得了巨大的成功。但是人们通过许多实验[15][71][76]证明了在情感分析问题上，布尔权重的效果往往优于词频方法。其原因也许是词频权重的假设，“词汇出现的越多越能代表文档”主要适用于对文档主题的代表性，而在情感分析问题上，往往是一些低频词汇或在字典里根本不曾存在过的如“goooooooooooood”，“bugfested”之类的新词或复合词更能代表文档的客观性和情感倾向。

3.4.3 词性

文档中词汇的词性（Part of Speech, POS）是情感分析和观点挖掘中一个常用的特征。能很好的帮助对多义词汇消歧[145]是它获得广泛应用的主要原因，当然随着自然语言处理技术的发展、条件随机场[68]等新工具的引入，自动词性标注（Part-Of-Speech Tagging）算法的精度大大提高，也使得词性作为一种特征更为可靠了。

基本的词性包括名词、动词、形容词、副词、代词、介词和连词8种，但常见的词性标注算法通常都会输出50到100种不同的词性标签，例如NN代表单数名词，NNS代表复数名词，NP代表单数专有名词等等。这与情感分析中所需要的粒度不符，需要进行一定的映射操作以合并过细的标注项。

Mullen[127]和Whitelaw[19]最早使用形容词作为情感分析特征，Hatzivassiloglou[44]等人证明了形容词能很好的帮助客观性分类和观点发现任务。

3.4.4 否定词

对否定词的处理是观点和情感极性分类中一个很重要的问题，因为否定词的出现往往会对句子的极性产生决定性的影响。例如“我喜欢这本书”和“我不喜欢

这本书”，从传统的基于统计的文本分类的角度看这两句话非常相似，但是其情感极性完全相反。

否定词作为特征有很多种形式，一种方法是在文档表示中不考虑否定词问题，而是把否定词的出现作为一个二阶特征，在后处理中加入；也有人[116]为否定词设定一个影响域，其影响范围内的其他词汇都附加一个特殊符号被转化为另一个不同的词汇。例如句子“I don't like this book.”，在影响域为 1 的情况下转化为“I-Not don't like-NOT this book”。

对否定词进行特征抽取的困难有两点，都来自于自然语言的复杂性：

相同的否定词在不同的语境下其作用不同。比如当强否定词 *No* 和词汇 *wonder* 结合起来变成词组“*No wonder*”时就不再具有否定含义，另外还有 “*not best*”这样的否定语法表达肯定语义的表达，可见否定词在自然语言中的作用远比二元逻辑中的 NOT 操作符复杂得多。Kennedy 和 Inkpen[75]发现使用否定词作为特征可以提高情感分析的精度约 3%，但进一步的否定词分析则需要引入更复杂的语法分析。

对否定词进行建模的另外一个困难来自于人们有时候会用嘲讽、夸张等方式来表示否定含义。即使在处理系统中加入语法模板也很难捕捉到这类否定语义。电影评论数据集[75]的平均情感分类准确率低于产品评论[82]，部分原因就在于人们在撰写影评的时候更倾向于使用辛辣的讽刺。

3.4.5 同义词扩展

在情感分析中用 WordNet 之类的字典进行同义词扩展的主要有两种方法，

在无监督的情感极性分类里，往往存在一个事先给定的由具有强烈情感倾向的词组成的“种子列表”。Andreevskaia 和 Bergler[3]以该种子列表为基础，利用 WordNet 字典中同义、反义等关系生成扩展列表，再通过计算文本中被包含在两个列表中词的个数做情感分析。其他人[5][7][122]则进一步将这种基于字典的方法和文本挖掘中的一些词汇共现关系统计结合起来,或在获得的列表基础上引入更多语法分析方法[45][69]。

在句子级别的有监督情感分析中，使用同义词扩展可以帮助解决由于文本过短造成的数据稀疏问题。

3.5. 实验

3.5.1 数据集与工具包

本文的情感分析实验主要针对两个分类问题，词级别和句子级别的情感分类，分别采用两套网络文本数据集。

本文使用的词级别情感分类数据来自于我们手工标注的 Blog 日志。我们对日志中的每个词，按照正负情感极性进行标注，在正负情感倾向的词汇各有 3 个强度等级，加上中立词共 7 个类别。各个类别的标注词汇数量如下表。无情感倾向的词并未明确标注，我们在实验中从完全没有情感词标注的中性句子里随机抽取出词汇作为中性词训练数据。

表 3-1 词级别情感标注数据集的各类别词汇数量

3	2	1	0	-1	-2	-3
70	982	357	-	303	511	29

本文研究中使用的句子集正面/负面语料集[54]首先在文献[104]中公布使用。该数据集由 5331 个标注为正面倾向、5331 个标注为负面倾向的文本片段构成，全部文本取自 ROTTEN TOMATOES 网站[62]。

我们使用 Stanford POS Tagger 对词汇进行词性标注[80][81]，否定词特征则通过我们手工标注的辞典进行判断。

本文中的分类实验使用标准的最大熵分类器，100 次迭代，执行 5 折的交叉检验（Cross Validation）。实验中直接以分类的准确率作为评价标准。

3.5.2 语言特征选择方法的对比

在最大熵模型中， 从理论上看，在建模过程中应该已含有特征选择和特征加权的性质：即一个特征 t_k 是否必要，并且它的重要性有多强，都由最大熵模型中分配给它的参数 λ_k 决定。如果一个特征是冗余的，那么它在最大熵模型的将得到一个零权重；而一个显著特征将获得一个相对高的权重。

然而，由于最大熵模型实现时，采用最大似然估计以及参数训练的迭代近似算法，理论上的特性并不能完全保证。因此，我们仍然认为良好选择并加权的特征将使得最大熵模型更有效。

按照第 3.4 节所讨论的特征选择和权重计算办法，我们使用了词汇的 Unigram 特征（Term）、词性特征（POS）、否定词特征（Neg）和同义词扩展特征（WordNet）的 4 种特征及其权值计算加权组合。

在句子级别的情感分类实验中，我们对词汇的 Unigram 特征使用布尔权重。由于在词的情感极性分析实验中，单纯的使用 Unigram 特征完全没有意义。为了获得词汇的上下文信息，我们引入了以目标分类词汇为中心的五元组。在实验中我们对五元组中的每个词汇进行相同的权重计算，特征的位置信息则通过在特征

前加注标签体现。

表 3-2 词和句子级别情感极性分类结果

features	Word			Sentence	
	sentiment-2	polarity-3	polarity-7	sentiment-2	polarity-3
Term	85.46%	82.79%	68.50%	**98.04%**	**83.90%**
Term + POS	89.58%	87.00%	73.00%	89.38%	75.61%
Term + POS + Neg	**91.08%**	**88.60%**	**74.85%**	91.27%	78.13%
Term + POS + WordNet	88.70%	86.04%	73.13%	89.40%	75.94%

我们使用这些特征实验了在两种数据上实验了有情感/无情感(Sentiment-2)，正面/中性/负面(Polarity-3)和情感分级（+3/+2/+1/0/-1/-2/-3, Polarity-7）三种分类任务。结果如表 3-所示，从特征选择的角度分析，我们可以看到：

对于词级别的情感分类任务：

- 任务的难度随分类类别数的增加提高，但是各种特征组合的优劣比较在三种不同分类任务中都是一致的。
- 只使用词汇特征的特征选择方式表现最差，显然单纯使用词汇特征没法充分的表达词汇间的语法关系。
- 引入词性和否定词特征可以明显的提高词级别感情分类的精度，这也与前文中提到的 Mullen[127]和 Kennedy [4]的发现相符。
- 同义词扩展特征在分类中只能起到反作用。我们分别实验了三种不同的 WordNet 扩展规则，没有观察到他们在对分类精度的影响上有任何区别。

对于句子级别的分类任务：

- 只使用词汇特征的精度要比所有其他方法都有好，这与词级别分类任务的情况正好相反。
- 词性特征在分类中只能起到反作用
- 否定词特征能够优化加入词性特征的结果，但跟仅使用 unigram 特征的准确率仍然相差很远。
- 和在词分类任务中的表现一样，同义词扩展无法帮助对词义进行消歧反而引入了更多噪声，之前提到的数据稀疏假设也并不成立。与在非监督学习中的巨大作用不同，同义词扩展技术在基于统计的情感分析中的应用还需要更多研究。

通过对实验数据的分析，我们可以清楚的认识到不同的情感分析任务需要不同的特征组合。使用基本的 unigram 词汇特征能获得尚可的分类效果，但是要想再提高精度则需要有选择的引入更多特征。

单纯的将词性、否定词等附加特征加入文本表示模型中和词汇特征一起训练很难收到理想的效果。我们看到，当把附加特征和位置信息结合并引入词窗口后，

附加特征可以明显的改善分类效果。但是在句子级别实验中，没有使用这些方法，直接添加特征到表示模型的结果很不理想。

第四章 Blog观点检索系统

4.1 引言

作为本文前述技术的应用尝试，本文作者在2008年参加了TREC Blog Track评测。本章将以该评测为主线，对参评系统进行介绍。

4.2 TREC Blog评测介绍

4.2.1 评测历史及发展现状

随着主题观点检索的研究兴起，在2006年，美国国家标准技术局（National Institute of Standards and Technology，NIST）主办的文本检索会议TREC[51]的系列评测项目中，首次引入了Blog检索项目（Blog Track）[47]，其目的在于对针对Blog的信息检索进行探索性的研究。至今，该项目已成功举办了三届。在这三年的Blog评测中，主任务均为观点检索（Opinion Retrieval Task），该评测项目与其他信息检索评测的主要不同就在于，其他检索侧重于信息内容的事实性方面，而Blog观点检索关注于信息内容的观点性方面[101]。

在2006年的Opinion Retrieval任务中，共有14个单位参加；到了2007和2008年，参赛单位则达到了20家，成为TREC各项评测专项中最炙手可热的参赛项目。在这几年的评测中，多家参评单位被邀请提交了各自的参赛系统报告。这些报告中描述了参评单位所使用的方法、工具以及对结果的分析。根据三年来的评测总结[67][89][101]和各参评单位的报告，这里我对Blog观点检测的评测现状以及其中一些代表性的工作总结如下：

在过往几届的Opinion Retrieval任务中，绝大多数的参评单位都使用二阶段的处理方法。第一阶段，采用传统的Ad-hoc信息检索技术，在Blog文本集中寻找与给定主题相关的文本，按照相关性进行排序后获得相关文档列表；第二阶段，通过一些主观性判断方法，对第一阶段获得的列表进行过滤和重排序。

在第一阶段的主题相关检索中，各参评单位都使用了一些现成的开源检索平台工具，如Indri[60]、Terrier[46]等等，通过这些工具完成文档的索引和相关性索引。这类工具所用到的相关性检索模型有TFIDF向量空间模型、语言模型、贝

叶斯推理网等等。同时由于页面预处理效果对后端的相关性和主观性判断任务的准确率有至关重要的影响，一些网页文档预处理技术如 Spam 检测、网页切分、查询扩展、相关反馈等技术也都有应用。

在第二阶段的主观性判断任务中，用到的方法大致有以下三种：

- 基于词典的方法：利用一些已有或自动生成的或手工标注的情感词及其情感分值的列表，根据相关文档中的情感词出现的频率和位置等信息，对文档进行观点性打分排序。这类方法对基于主题相关性检索结果的改变作用上，好坏不一，在有的系统中收到了一些效果，但是在有些系统上应用这类技术后的排序准确率反而不如应用之前。
- 语义分析方法：对文档中的文本内容进行词性标注、句法分析等语义解析，使用其中的代词、形容词、副词等词性的词语、短语作为观点性内容的标识。这类方法的有效性在评测中表现较为有限。
- 情感分类方法：利用一些已有的主/客观语料资源，采用机器学习的方法（参评系统中大多使用支持向量机），训练出主客观分类器，对文档中的观点性内容进行判别和打分。但由于目前还缺乏较大规模的情感分类的训练语料资源，并且已有的语料与 Blog 数据集中的内容特性上差距较大，这类方法的作用收到了很大限制。使用这类方法的参评单位之间表现出来的水平区别很大程度上与使用的训练语料的质量有关。

4.2.2 评测数据、任务与相关技术指标

在 TREC Blog Track 评测中，采用的是 Glasgow 大学提供的 Blog06 数据集[18]。该数据集收集了 2005 年 12 月至 2006 年 2 月间，从 100,649 个 Blog 上采集来的文档，其中 permalink 文档为评测中的检索对象。Blog06 数据集的简单情况如下表 4-1 所示，

表 4-1 Blog06 数据集资源统计

Feeds		Permalink Documents		Homepages		总计
数量	大小	数量	大小	数量	大小	大小
753,681	38.6G	3,215,171	88.8GB	324,880	20.8GB	148.2GB

在 TREC Blog Track 的 Opinion Retrieval 任务中，每年评测主办方都发布 50 个查询主题（topic）。对每个 topic，参评系统需要找出对该主题表达了某方面主观观点的 Blog 文章。查询的目标包括是传统的命名实体（Named Entity，如人名、地名、机构名），也可以是概念（如技术类型）、产品名称或者一个事件。该任务可以描述为："人们如何看待 X"，其中 X 为查询目标[89]。Blog Track 中的查询主

题格式与其他TREC评测任务中相似，图4-1是2007年的一个评测topic。每个topic包含以下几部分：

1）topic编号，对应<num>字段；

2）topic标题，对应<title>字段，是主要的查询目标，在baseline系统中仅允许利用该字段信息；

3）topic描述，对应<desc>字段，对主题的简要解释说明；

4）topic陈述，对应<narr>字段，对主题查询相关的补充说明。

```
<top>
    <num> Number: 1048 </num>
    <title> Sopranos </title>
    <desc>
        Description:
        Find opinions about "The Sopranos", a very popular, long running
        television program.
    </desc>
    <narr>
        Narrative:
            The Sopranos is the story of a mob leader and how he balances his
        life with his personal family life. Opinions as to whether mob violence
        should have been glorified on television are relevant. Documents
        expressing the attraction people had for watching the show are relevant.
        Positive comments regarding how well the male lead performs his dual
        life are very relevant.
    </narr>
</top>
```

图4-1 TREC评测Topic示例

在Opinion Retrieval任务中，各个参评单位最多可提交5个检索结果列表，其中必须有一个仅使用查询的标题字段、不加入观点性判断处理的、自动运行的检索结果，作为基准（baseline）系统，以便对后续作复杂查询处理、加入观点性评估技术的系统性能做对比。返回检索结果时，每个topic最多返回前1000个按照相关性和观点性综合排序的文档编号列表及相应的分值；而主办方在对返回的检索结果进行评估时，以-1～4级对文档进行标记评估，分别代表：-1，未能判断；0，不相关；1，相关但没有观点性；2，相关且具有负面观点；3，相关

且同时具有正面和负面观点；4，相关且具有正面观点。其中，2～4标记的文档均为具有观点性的文档，在观点检索中评判时，仅认为这三类文档是有效结果；而标记为1的文档，在通常的信息检索中为有效的相关结果，但在此项评测中，由于不具有观点性内容而被视为无效结果。

在Blog Track的评测中，参照其他TREC的ad-hoc检索任务，主要采用以下几项指标，对各参评系统所提交结果的性能做衡量：

- 平均准确率（Mean Average Precision，MAP）

MAP为每个相关文档被检索到时的精确率的平均值，即

$$MAP(Q) = \frac{\sum_{i=1}^{r_Q} \frac{i}{\#Doc_Q(i)}}{R_Q} \qquad \text{式（4-1）}$$

其中R_Q为查询Q在文档集合中相关文档的总数，r_Q为检索出的相关文档数，$\#Doc_Q(i)$为在检索结果中，第i篇相关文档被检索出时，之前已被检索出的文档数。

- R-准确率（R-Precision，R-Prec）

R-准确率指全部R_Q个相关文档找到时的准确率，即

$$R-Prec(Q) = \frac{R_Q}{\#Doc_Q(R_Q)} \qquad \text{式（4-2）}$$

- 前10项准确率（Precision at 10 documents，P@10）。

P@10指检索结果中前10篇文档中相关文档所占的比率。

其中，MAP作为主要评估排名指标。在评测中，使用TREC官方发布的trec_eval脚本[52]，自动生成上述评测指标值。

4.3 Blog观点检索系统设计与评测

4.3.1 Blog主题检索系统

本文的Blog观点检索系统采用的是两阶段（two-stage）的检索框架模式，如前4.2.2节所述，该模式也是Blog Track中绝大多数参评系统所采用的观点检索策略。本节将首先介绍系统中的第一阶段——Blog主题检索平台。

根据TREC Blog Track的评测要求，首先需要提交一个仅使用查询主题的<title>字段词条，并且不加入观点识别处理的自动运行的baseline系统，作为后续评测基础。因此，本文首先使用Indri自带的默认trec-web结构HTML解析器，对Blog06数据集进行解析、索引，并仅使用07年的50个topic中的<title>字段词条作为查询词，得到baseline系统Pris07Basee的检索结果。根据官方评测结

果数据，该系统的主要性能指标如下表4-2所示，表中同时列出了07年Blog Track测试的其他参评系统的性能指标值。

表 4-2. TREC 2007 Blog 评测——title-only 自动运行结果[89]

Group	Run	MAP	R-prec	b-Bref	P@10
UIC	uic1c	**0.4341**	**0.4529**	**0.4724**	**0.69**
UAmsterdam	uams07topic	0.3453	0.3872	0.3953	0.562
IndianaU	oqlr2fopt	0.335	0.3925	0.378	0.576
UGlasgow	uogBOPFProxW	0.3264	0.3657	0.3497	0.552
DalianU	DUTRun2	0.319	0.3671	0.3686	0.6
FudanU	FDUTisdOpSVM	0.3179	0.3467	0.3501	0.454
FIU	FIUDDPH	0.3053	0.3498	0.3475	0.492
UNeuchatel	UniNEblog3	0.3049	0.3438	0.3266	0.516
CAS	Relevant	0.3041	0.36	0.3779	0.446
Uarkansas Littlerock	UALR07BlogIU	0.2911	0.3263	0.3134	0.58
UWaterloo	UWopinion3	0.2631	0.3344	0.298	0.496
CAS	NLPRPTD2	0.2587	0.3088	0.2956	0.456
Zhejiangu	EAGLE1	0.2561	0.3159	0.2867	0.428
BUPT	**Pris07Base**	**0.2466**	**0.3018**	**0.2835**	**0.456**
KobeU	KobePrMIR01	0.246	0.3011	0.2744	0.44
NTU	NTUManualOp	0.2393	0.2659	0.2749	0.486
KobeU	Ku	0.1689	0.2417	0.219	0.254
RGU	rgu0	0.1686	0.2266	0.2163	0.288
UBuffalo	UB1	0.1501	0.2001	0.1887	0.266
Wuhan	NOOPWHU1	0.0011	0.0071	0.0072	0.008

从上表结果可见，我们07年测试提交的系统的相关性检索阶段的性能并不理想。因此我们在Blog网页解析、Spam过滤、查询构造与扩展、域查询等方面进行了一些尝试。

4.3.1.1 Indri 相关检索平台

本文系统的Blog主题相关性检索平台使用了Massachusetts大学与Carnegie Mellon大学共同开发的Indri信息检索系统[60]。Indri系统是基于语言模型（Language Modeling）[105]与推理网（Inference Network）[130]相结合的检索框架[123]。这两种技术已经在信息检索任务中被广泛的研究和应用，是当前最为有效的检索模型。Metzler和Croft首次提出将语言模型和推理网这两种模型结合用于信息检索中[94]。

Indri检索系统具有强大的结构化查询语言，能够支持构造复杂的查询，包

括：

- 查询词项的加权（#weight 操作符）
- 词项的有序/无序、与或非的逻辑组合关系（#owN、#combine、#filrej 等操作符）
- 词项的同义性、近似性、句法关系（#symc 等操作符）
- 已抽取的实体、日期范围、数值范围及比较（如 Bush.author、#date:before、#date:after、#date:between、#less、#greater、#equal 等操作符）
- 句子、段落、域（field）、文档结构及多文档的不同粒度级别的检索（term.field、#any、#combine[field](q1 ... qn)等操作符）

此外，Indri 直接具有对 TREC 结构的文本、XML、HTML 及普通的纯文本的解析、索引功能，并且新版本中还加入了对 doc、pdf 等高级文本格式的支持。同时，Indri 系统还具有非常稳健、高效的，对超大规模数据集的快速索引、并行查询的处理能力和兼容性，已成功的应用于 TREC 千兆比特专项（Terabyte Track）评测中[95]。

4.3.1.2 网页信息抽取子系统

由于 Blog06 数据集中，在线抓取的 Blog 网页非常杂乱，对网页数据的解析预处理则显得尤为重要。Indri 自带的 HTML 网页解析模块仅实现了较简单的 HTML 标签删除处理方法,对于复杂的 HTML 结构显然是过于简单，是导致上述基本系统的检索性能较差的一个主要原因。

本文首先使用上文中的非英语文档过滤算法清除与 Blog06 的主题任务无关的非英语文档，之后进行标签对齐和文档正文抽取。我们使用在正文抽取过程中使用 Spam 检测算法过滤掉 Spam 网页文本块。重新解析后的网页中仅包含页面的文本内容，所得到的数据集合大小缩减为约 15G。

表 4-3. Indri 的页面净化算法和本文算法的比较

系统	MAP	MAP 提高	R-pre	P@10
最好	0.4341		0.4529	0.69
Pris07Base	0.2466	-	0.3018	0.456
Pris08Page	0.3761	52.51%	0.4122	0.6840
最差	0.0011	-	0.0071	0.008

在重新解析后的 Blog06 permalink 数据集合上，除了 HTML 解析模块之外，仍采用 Indri 系统默认的索引和检索参数对净化后的网页全文进行索引，获得索引 Pris08Page。我们这个索引上只使用 07 年评测数据的 Title 字段进行检索，结合官方发布的标准数据和脚本[52]对结果进行评测，结果如表 4-所示。可以看到，

使用本文的页面正文提取算法获得的数据集，较 Baseline 系统的性能有明显的提高。

4.3.1.3 段落级别相关性检索系统

本文在 2.4.3 节提出了基于 SAX 接口的页面内容块分割算法，使用该算法可以对文本进行自动分段。如果以段落为单位进行索引，则可以避免一部分由于文档过长或文档多主题共存造成的伪相关现象，在损失一部分召回率的情况下改善相关检索的准确率。因此我们进行了如下实验：

我们使用与上一节相同的 Indri 检索系统和索引参数，对在抽取 Pris08Page 数据集的过程中切分出来的的网页段落直接进行索引，获得索引 Pris08Paragraph。使用 07 年的评测数据的 Title 字段检索，针对每个 query 获得相应的段落相关性排序。对每篇文档，取相关性最高的段落作为文档与 query 的相关性分值。从表 4-4 中可以看到，使用相同的页面净化算法，更细的检索粒度确实可以提高检索效果。

表 4-4 段落级别相关性检索结果（Title-Only）

系统	MAP	MAP 提高	R-pre	P@10
最好	0.4341		0.4529	0.69
Pris07Base	0.2466	-	0.3018	0.456
Pris08Page	0.3761	52.51%	0.4122	0.6840
Pris08Paragraph	0.4543	84.23%	0.4935	0.7
Pris08Combine·	0.5023	103.69%	0.535	0.75
最差	0.0011	-	0.0071	0.008

最后我们将 Pris08Paragraph 和 Pris08Page 的结果进行合并，将两者的相关性分值加权相加获得了 Pris08Combine。结合了页面和段落级别的相关性计算优势，Pris08combine 使我们获得的最好的相关性结果。

4.3.1.3 查询扩展子系统

根据 Blog Track 评测规定，在 Blog 相关性检索方面，除了提交自动运行的 baseline 系统，其他系统中可以采用任何其他方法和资源，包括人工方式，对检索系统进行改进。通过 Pris07Basee 和 Pris08Base 系统与其他系统的比较，可以发现尽管应用的网页信息抽技术可以显著的提高相关检索的准确率，但是由于这种准确率的提高是以大量去噪获得的，所以会降低系统的召回率。因此，充分利用 Indri 系统的结构化查询语言，对查询主题 Topic 进行扩展构造，也是后续改进工作的一个重要方面，我们使用的方法包括：

- 使用 Indri 结构化查询语言，对各个 topic 中的 title 字段的已有查询词做

进行权重调整，增强区分度高的主题相关词在相关性计算中的权重

- 使用与或逻辑操作符帮助构造查询
- 使用文本窗对查询次进行扩展
- 挑选 topic 的<desc>、<narr>字段中的部分关键词，对查询进行人工查询扩展；
- 利用 google 搜索引擎作为伪相关反馈数据源，将查询词条进行网络检索，自动抓取首页的高相关性页面进行统计分析。
- 使用维基百科（Wikipedia）进行查询扩展

例如，图 4-2 中示例的查询主题“Sopranos”，通过 Indri 结构化查询语言，作查询词扩展和结构化查询构造如图 4-2 所示：

```
<query>
    <num> Number: 1048 </num>
    <text>
        Sopranos.(title)
        #syn
        (
            Sopranos
            Soprano
        )
        #weight
        (
          1.0   Sopranos
          1.0   Soprano
          0.2   lyric
          0.2   television
        )
    </text>
</query>
```

图 4-2. TREC Blog Track 查询扩展与结构化查询主题示例

根据上述方法，对 2007 年 Blog Track 的全部 50 个 topic 进行查询扩展和结构化构造后，其相关性检索性能见于表 4-5。同时，也对 2007 年 Blog Track 测试中，在非限定的查询结果的最好/最差性能做对比。

表 4-5. 查询扩展与基于文档 title 域的查询结果对比

系统	MAP		MAP 提高
	Title Only	Query Expanded	
最好	0.4341	-	-
Pris07Base	0.2466	0.3432	39.17%
Pris08Page	0.3761	0.4625	22.97%
Pris08Paragraph	0.4543	0.4996	9.97%
Pris08Combine	0.5023	0.5478	9.06%
最差	0.0011	-	-

4.3.2 基于情感分类的 Blog 观点检索系统

本文的 Blog 观点检索系统采用了两阶段模型，既第一阶段进行相关性检索获得与主题“相关”的文档列表，再对相关文档进行情感极性分析，将文档极性和相关性结果结合后重新排序，获得观点性文档列表。

在参加 2007 年的评测时，我们采用了 3.5.1 节里使用过的句子级别文本片段进行主客观分类样本训练了最大熵模型，以之判断 Blog 文档全文的观点极性。由于训练语料与真实数据在规模和文本特性上存在很大区别，因此引入该情感分析模块后反而大幅劣化了我们的检索性能。

在今年的测试中，官方发布了人工标注过的文档集。我们以该文档集作为训练样本训练了句子级别的情感极性分类器，再以文档的句子级别情感分值序列作为样本训练篇章级别主客观分类器[106][147]。

该模型在在 2007 年主题集合上进行对比评测，使用 2006 年 Blog Track 中的主题对应的标记文档集合。下表中 PrisOpnS 对应于使用简单句子得分的结果，prisOpnD 则使用层叠式文档模型。这两种观点检索模型，结合上文中介绍的基于段落的检索系统作为基本的 ad-hoc 主题检索系统，进行对比评测如表 4-6 所示。

表 4-6. Blog Opinion Retrieval 的对比评测[147]

系统	MAP	MAP 提高	R-Prec	bBref	P@10
最好	0.4341	-	0.4529	0.4724	0.69
prisTopic	0.3357	-	0.386	0.3725	0.578
prisOpnS	0.3615	7.69%	0.4	0.396	0.542
prisOpnD	0.3633	8.22%	0.399	0.3687	0.566

值得注意的是，表中所示的由 Illinois 大学提交的最好成绩在 2007 年的评测中非常突出。而当年第二名的 MAP 指标仅为 0.3453，可以认为，本文 Blog 观点检索系统的整体性能已经达到当前领先水平。

第五章 总结与展望

6.1. 工作总结

本文对 Blog 检索中涉及的网页处理、文本情感分析技术进行了有针对性的研究工作，在此基础上实现了一套 Blog 观点检索系统。在研究的过程中取得了一些成果，也有很多不足。

对于网页的解析处理，考虑到目前网络信息处理系统的应用现状和面临的主要挑战，本文将研究的重点放在了网页、特别是 Blog 网页的正文提取上，同时也兼顾讨论了页面级别、页面内容块级别的 Spam 检测技术，以及网络文本的语言识别问题。在吸收现有技术的基本思想，并充分分析其性能局限的基础上，在之前讨论的每个方面分别提出了一套高效、健壮的互联网网络文本处理算法。以此为基础实现的系统在参加 TREC 测试时有很好的表现，具有相当的实用性。

在文本情感分类方面，分别对词汇的 N-gram 特征及其各种权重计算方法、词性特征、否定词特征和同义词扩展特征在当前情感分析领域的应用和效果进行了分析。通过词级别和句子级别的情感极性分类实验，探究了几种特征及其各种组合的应用效果。证实了在文本表示中不宜直接引入高级特征，而需在特征中加入语法、位置信息才能帮助提高分类准确率。

围绕 TREC Blog 评测的需求，我们综合运用了上文中提到的 HTML 解析、噪声标签过滤、文本内容提取等网页预处理方法；以 Indri 检索平台为基础，利用结构化查询语言以及篇章段落级文档结合的检索策略，有效的提高了 Blog 文档的主题相关性检索性能。在 Blog Track 数据集上的评测指标表明，本文构造的 Blog 观点检索系统达到了较高的性能水平。

6.2. 需要进一步解决的工作

在本文的工作中，也存在很多不足，需要进一步的解决：

第一，在网页正文提取中，没有更加有效的利用 HTML 的结构化信息，对于正文的频繁 div 路径模式这个非常有价值的特征没有利用；在语言识别上没有提出更加通用的方法；提取出的正文也缺乏 Blog 正文和读者评论之间的分类标注。

第二，文本情感分析的特征选择问题。虽然得出了简单在文本表示模型中添加高阶特征无助于提高分类精度的结论，但是在如何解决这一问题的探索上仍然浅尝辄止，对于句子级别的特征选择并没有给出一个合理的方案。这些需要在将来的工作中继续探索。

第三，篇章段落结合的相关检索策略固然能提高精度，但是对于空间和时间的复杂度要求都过高，需要继续研究在全文检索中支持段落相关性的方法。

最后，由于作者的学识水平有限，一些问题的研究有待于进一步深入、完善，文中不当之处难免，恳请各位前辈、专家、同行批评指正。

参考文献

[1] A Ntoulas, M Najork, M Manasse, D Fetterly. Detecting spam web pages through content analysis. WWW06, pp.83-92.

[2] A. Bencz′ ur, K. Csalog′ any, T. Sarl′ os and M. Uher. SpamRank – Fully Automatic link Spam Detection. In 1st International Workshop on Adversarial Information Retrieval on the Web, May 2005.

[3] Alina Andreevskaia and Sabine Bergler. Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In Proceedings of the European Chapter of the Association for Computational Linguistics (EACL), 2006.

[4] Alistair Kennedy and Diana Inkpen. Sentiment classification of movie reviews using contextual valence shifters. Computational Intelligence, 22(2, Special Issue on Sentiment Analysis)):110–125, 2006.

[5] Andrea Esuli and Fabrizio Sebastiani. Determining the semantic orientation of terms through gloss analysis. In Proceedings of the ACM SIGIR Conference on Information and Knowledge Management (CIKM), 2005.

[6] Andreevskaia A, Bergler S. Mining wordnet for a fuzzy sentiment: sentiment tag extraction from wordnet glosses. In Proc. of the 11th Conf. of the European Chapter of the Association for Computational Linguistics, 2006, 209-216.

[7] Anthony Aue and Michael Gamon. Automatic identification of sentiment vocabulary: Exploiting low association with known sentiment terms. In Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing, 2005

[8] Ashish, N. and Knoblock, C. A., Semi-Automatic Wrapper Generation for Internet Information Sources, In Proceedings of the Conference on Cooperative Information Systems, 1997, pp. 160-169.

[9] B. Davison. Recognizing nepotistic links on the web. In AAAI-2000 Workshop on Artificial Intelligence for Web Search, pages 23–28, 2000.

[10] B. Wu and B. D. Davison. Identifying link farm spam pages. WWW'06, 2006.

[11] B. Wu, V. Goel & B. D. Davison. Topical TrustRank: using topicality to combat Web spam. WWW'2006.

[12] Beeferman D, Berger A, Lafferty J. Statistical models for text segmentation. Machine Learning, 34(1-3), 1999, 177-210.

[13] Beesley, Kenneth R. 1998. Language identifier: A computer program for

automatic natural-language identification of on-line text. In Languages at Crossroads: Proc. Annual Conference of the American Translators Association, pp. 47–54.

[14] Berger A, Della Pietra S, Della Pietra V. A maximum entropy approach to natural language processing. Computational Languages, 22(1), 1996, 39-71.

[15] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 79–86, 2002.

[16] Bruce R, Wiebe J. Recognizing subjectivity: a case study in manual tagging. Natural Language Engineering, 5(2), 1999, 1-16.

[17] Buckley C, Voorhees EM. Retrieval Evaluation with Incomplete Information. In Proc. of the 27th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 2004, 25-32.

[18] C. Macdonald and I. Ounis. The TREC Blog06 Collection : Creating and Analysing a Blog Test Collection DCS Technical Report TR-2006-224. Department of Computing Science, University of Glasgow. 2006.

[19] Casey Whitelaw, Navendu Garg, and Shlomo Argamon. Using appraisal groups for sentiment analysis. In Proceedings of the ACM SIGIR Conference on Information and Knowledge Management (CIKM), pages 625–631. ACM, 2005.

[20] Cavnar, William B., and John M. Trenkle. 1994. N-gram-based text categorization. In Proc. SDAIR, pp. 161–175.

[21] Chakrabarti, S., Integrating the Document Object Model with hyperlinks for enhanced topic distillation and information extraction, In the 10th International World Wide Web Conference, 2001.

[22] Chakrabarti, S., Joshi, M., and Tawde, V., Enhanced topic distillation using text, markup tags, and hyperlinks, In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval , ACM Press, 2001, pp. 208-216.

[23] Chakrabarti, S., Punera, K., and Subramanyam, M., Accelerated focused crawling through online relevance feedback, In Proceedings of the eleventh international conference on World Wide Web (WWW2002), 2002, pp. 148-159.

[24] Chen SF, Rosenfeld R. A Gaussian prior for smoothing maximum entropy models. Tech. Rep. CMUCS-99-108, Carnegie Mellon University, 1999.

[25] Cohen WW, Singer Y. Context-sensitive learning methods for text categorization. In Proc. of the 19th Annual Int. ACM SIGIR Conf. on Research and Development in

Information Retrieval, 1996, 307-315.

[26] Croft, W. Bruce, and David J. Harper. 1979. Using probabilistic models of document retrieval without relevance information. Journal of Documentation 35(4):285–295.

[27] Csiszár I. I-Divergence geometry of probability distributions and minimization problems. The Annals of Probability, 3(1), 1975, 146-158.

[28] D. Fetterly, M. Manasse & M. Najork. Detecting phrase level duplication on the World Wide Web. SIGIR'2005.

[29] D. Fetterly, M. Manasse and M. Najork. Spam, Damn Spam, and Statistics: Using Statistical analysis to locate spam web pages. In 7th International Workshop on the Web and Databases, June 2004.

[30] D. Gibson, K. Punera, and A. Tomkins. The volume and evolution of web page templates. In Proc. 14th WWW (Special interest tracks and posters), pages 830–839, 2005.

[31] Darroch JN, Ratcliff D. Generalized iterative scaling for log-linear models. Annals of Mathematical Statistics, 43, 1972, 1470-1480.

[32] Das SR, Chen M. Yahoo! for Amazon: sentiment extraction from small talk on the web. In Proc. of the 8th Asia Pacific Finance Association Annual Conf., 2001.

[33] Deerwester S, Dumais ST, Furnas GW, et al. Indexing by latent semantic analysis. Journal of the American Society for Information Science,·41(6), 1990, 391-407.

[34] Della Pietra S, Della Pietra V, Lafferty J. Inducing features of random fields. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(4), 1997, 380-393.

[35] Deng Cai, Xiaofei He, Ji-Rong Wen, and Wei-Ying Ma. Block-Level Link Analysis. In Proceedings of SIGIR, 2004.

[36] Dumais S, Platt J, Heckerman D, et al. Inductive learning algorithms and representations for text categorization. In Proc. of the 7th Int. Conf. on Information and Knowledge Management, 1998, 148-155.

[37] Dunning, Ted. 1994. Statistical identification of language. Technical Report 94-273,Computing Research Laboratory, New Mexico State University.

[38] Esuli A, Sebastiani F. Determining the semantic orientation of terms through gloss classification. In Proc. of the 14th ACM Int. Conf. on Information and Knowledge Management, 2005, 617-624.

[39] Finn A, Kushmerick N, Smyth B. Genre classification and domain transfer for

information filtering. In Proc. of the 24th BCS-IRSG European Colloquium on IR Research: Advances in Information Retrieval, 2002, 353-362.

[40] Genkin A, Lewis DD, Madigan D. Large-scale bayesian logistic regression for text categorization. 2004. Available at http://www.stat.rutgers.edu/~madigan/papers/.

[41] Goldberg AB, Zhu X. Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization. In Proc. of HLT-NAACL 2006 Workshop on Textgraphs: Graph-based Algorithms for Natural Language Processing, 2006, 45-52.

[42] H.-Y. Kao, J.-M. Ho, and M.-S. Chen. WISDOM: Web intrapage informative structure mining based on document object model. TKDE, 17(5):614–627, 2005.

[43] Han EH, Karypis G, Kumar V. Text categorization using weight adjusted k-nearest neighbor classification. In Proc. of the 5th Pacific-Asia Conf. on Knowledge Discovery and Data Mining, 2001, 53-65.

[44] Hatzivassiloglou V, McKeown KR. Predicting the semantic orientation of adjectives. In Proc. of the 8th Conf. on European Chapter of the Association for Computational Linguistics, 1997, 174-181.

[45] Hiroshi Kanayama and Tetsuya Nasukawa. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 355–363, Sydney, Australia, July 2006. Association for Computational Linguistics.

[46] http://ir.dcs.gla.ac.uk/terrier/

[47] http://ir.dcs.gla.ac.uk/wiki/TREC-BLOG/

[48] http://openmind.media.mit.edu/

[49] http://technorati.com/blogging/state-of-the-blogosphere/

[50] http://tidy.sourceforge.net

[51] http://trec.nist.gov/

[52] http://trec.nist.gov/trec_eval/trec_eval_latest.tar.gz

[53] http://www.altavista.com/

[54] http://www.cs.cornell.edu/people/pabo/movie-review-data

[55] http://www.cs.stir.ac.uk/~kjt/software/web/htmlfix.html

[56] http://www.douban.com/

[57] http://www.epinions.com/

[58] http://www.google.com/

[59] http://www.imdb.com/

[60] http://www.lemurproject.org/indri/

[61] http://www.live.com/

[62] http://www.rottentomatoes.com

[63] http://www.wjh.harvard.edu/~inquirer/

[64] http://www.yahoo.com/

[65] Hu M, Liu B. Mining and summarizing customer reviews. In Proc. of the 10th ACM SIGKDD Iint. Conf. on Knowledge Discovery and Data Mining, 2004, 168-177.

[66] Hu M, Liu B. Mining opinion features in customer reviews. In Proc. of the 19th National Conf. on Artificial Intelligence (AAAI-2004), 2004, 755-760.

[67] Iadh Ounis, Craig Macdonald, Ian Soboroff. Overview of the TREC 2008 Blog Track. In Proc. the 17th Text REtrieval Conf., 2008

[68] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Intl. Conf. on Machine Learning, 2001.

[69] Jaap Kamps, Maarten Marx, Robert J. Mokken, and Maarten de Rijke. Using WordNet to measure semantic orientation of adjectives. In LREC, 2004.

[70] Jaime Carbonell. Subjective Understanding: Computer Models of Belief Systems. PhD thesis, Yale, 1979.

[71] Janyce M. Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. Learning subjective language. Computational Linguistics, 30(3):277–308, September 2004.

[72] Joachims T. Text categorization with support vector machines: learning with many relevant features. In Proc. of the 10th European Conf. on Machine Learning, 1998, 137-142.

[73] John A. Horrigan. Online shopping. Pew Internet & American Life Project Report, 2008.

[74] Kelsey group. Online consumer-generated reviews have significant impact on offline purchase behavior. Press Release, November 2007.

[75] Kennedy A, Inkpen D. Sentiment classification of movie reviews using contextual valence shifters. Computational Intelligence, 22 (2) , 2006, 110–125.

[76] Kiduk Yang, Ning Yu, Alejandro Valerio, and Hui Zhang. WIDIT in TREC-2006 Blog track. In Proceedings of TREC, 2006.

[77] Kleinberg, Jon M. 1999. Authoritative sources in a hyperlinked environment.

JACM 46(5):604–632.

[78] Koch, Peter-Paul (May 14, 2001). The Document Object Model: an Introduction. In Digital Web Magazine, 2006

[79] Konheim, Alan G. 1981. Cryptography: A Primer. JohnWiley & Sons.

[80] Kristina Toutanova and Christopher D. Manning. 2000. Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000), pp. 63-70.

[81] Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In Proceedings of HLT-NAACL 2003, pp. 252-259.

[82] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of WWW, pages 519–528, 2003.

[83] Lan Yi, Bing Liu, Xiaoli Li. Eliminating Noisy Information in Web Pages for Data Mining. In Proc. of the SIGKDD'03 Conf., pages 296-305, 2003

[84] Lewis DD. Naive (Bayes) at forty: The independence assumption in information retrieval. In Proc. of the 10th European Conf. on Machine Learning (ECML), 1998, 4-15.

[85] Lewis, D. D.; Yang, Y.; Rose, T.; and Li, F. RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research, 5:361-397, 2004.

[86] Lin WH, Wilson T, Wiebe J, et al. Which side are you on? Identifying perspectives at the document and sentence levels. In Proc. of the 10th Conf. on Computational Natural Language Learning, 2006, 109-116.

[87] Liu H, Lieberman H, Selker T. A model of textual affect sensing using real-world knowledge. In Proc. of the 11th Int. Conf. on Intelligent User Interface, 2003, 125-132.

[88] M. R. Henzinger: Finding near-duplicate web pages: a large-scale evaluation of algorithms. SIGIR'06, 2006.

[89] Macdonald C, Ounis I, Soboroff I. Overview of the TREC 2007 Blog Track. In Proc. the 16th Text REtrieval Conf., 2007

[90] Mark Kantrowitz. Method and apparatus for analyzing affect and emotion in text. U.S. Patent 6622140, 2003. Patent filed in November 2000.

[91] Marti Hearst. Direction-based text interpretation as an information access refinement. In Paul Jacobs, editor, Text-Based Intelligent Systems, pages 257–274. Lawrence rlbaum Associates, 1992.

[92] McCallum A, Nigam K. A comparison of event models for Naive Bayes text categorization. In Proc. of the AAAI-98 Workshop on Learning for Text Categorization, 1998, 41-48.

[93] Mei Q, Ling X, Wondra M, et al. Topic sentiment mixture: modeling facets and opinions in Weblogs. In Proc. of the 16th Int. Conf. on World Wide Web, 2007, 171-180.

[94] Metzler D, Croft WB. Combining the language model and inference network approaches to retrieval. Information Processing and Management Special Issue on Bayesian Networks and Information Retrieval, 40(5), 2004, 735-750.

[95] Metzler D, Strohman T, Turtle H, et al. Indri at TREC 2004: Terabyte Track. In Proc. of the 13th Text REtrieval Conf. (TREC 2004), 2004.

[96] Moffat, Alistair, and Justin Zobel. 1998. Exploring the similarity space. SIGIR Forum 32(1).

[97] N Jindal, B Liu. Opinion Spam and Analysis. In Proceeding of the international conference on Web search and web data mining, pp.219-230.

[98] Ni X, Xue G, Ling X, et al. Exploring in the Weblog space by detecting informative and affective articles. In Proc. of the 16th Int. Conf. on World Wide Web, 2007, 281-290.

[99] Nigam K, Lafferty J, McCallum A. Using maximum entropy for text classification. In Proc. of the Int. Joint Conf. on Artificial Intelligence IJCAI-99 Workshop on Machine Learning for Information Filtering, 1999, 61-67.

[100] Nigam K, Lafferty J, McCallum A. Using maximum entropy for text classification. In Proc. of the Int. Joint Conf. on Artificial Intelligence IJCAI-99 Workshop on Machine Learning for Information Filtering, 1999, 61-67.

[101] Ounis I, Rijke M, Macdonald C, et al. Overview of the TREC-2006 Blog Track. In Proc. the 15th Text REtrieval Conf., 2006.

[102] Page, Lawrence, Sergey Brin, Rajeev Motwani, and TerryWinograd. 1998. The Page-Rank citation ranking: Bringing order to the web. Technical report, Stanford DigitalLibrary Technologies Project.

[103] Pang B, Lee L, Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques. In Proc. Conf. on Empirical Methods in Natural

Language Processing, 2002, 79-86.

[104] Pang B, Lee L. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In Proc. of the 43rd Annual Meeting on Association for Computational Linguistics, 2005, 115-124.

[105] Ponte JM, Croft WB. A language modeling approach to information retrieval. In Proc. of the 21st Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 1998, 275-281.

[106] H.He, B.Chen, L.Du. PRIS in TREC 2008 Blog Track. In Proc. of the 17th Text Retrieval Confrance, 2008, 1-5.

[107] Ratnaparkhi A, Reynar J, Roukos S. A maximum entropy model for prepositional phrase attachment. In Proc. of the ARPA Human Language Technology Workshop, 1994, 250-255.

[108] Ratnaparkhi A. A maximum entropy model for part-of-speech tagging. In Proc. of the Empirical Methods in Natural Language Conf., 1996.

[109] Ratnaparkhi A. Maximum Entropy Models for Natural Language Ambiguity Resolution. [PhD thesis]. University of Pennsylvania, 1998.

[110] Rosenfeld R. Adaptive Statistical Language Modeling: A Maximum Entropy Approach. [PhD thesis]. Carnegie Mellon University, 1994.

[111] Ruiz ME, Srinivasan P. Hierarchical neural networks for text categorization. In Proc. of the 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 1999, 281-282.

[112] S. Debnath, P. Mitra, N. Pal, and C. L. Giles. Automatic identification of informative sections of web pages. TKDE, 17(9):1233–1246, 2005.

[113] Sahami M. Learning limited dependence Bayesian classifiers. In Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining, 1996, 335-338.

[114] Salton G, Wong A, Yang CS. A vector space model for automatic indexing. Communications of the ACM, 18(5), 1975, 613-620.

[115] Salton, Gerard, and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. IP&M 24(5):513–523.

[116] Sanjiv Das and Mike Chen. Yahoo! for Amazon: Extracting market sentiment from stock message boards. In Proceedings of the Asia Pacific Finance Association Annual Conference (APFA), 2001.

[117] Schapire RE, Singer Y. BoosTexter: A boosting-based system for text categorization. Machine Learning, 39(2-3), 2000, 135-168.

[118] Sebastiani F. Machine learning in automated text categorization: a survey. Tech. Rep. IEI-B4-31-1999, Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT, 1999.

[119] Sebastiani F. Machine learning in automated text categorization: a survey. Tech. Rep. IEI-B4-31-1999, Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT, 1999.

[120] Singhal, Amit, Gerard Salton, and Chris Buckley. 1995. Length normalization in degraded text collections. Technical report, Cornell University, Ithaca, NY.

[121] Singhal, Amit, Gerard Salton, and Chris Buckley. 1996b. Length normalization in degraded text collections. In Proc. SDAIR, pp. 149–162.

[122] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In Proceedings of the International Conference on Computational Linguistics (COLING), 2004.

[123] Strohman T, Metzler D, Turtle H, et al. Indri: A language model-based search engine for complex queries (extended version). CIIR Technical Report, 2005.

[124] Subasic P, Huettner A. Affect analysis of text using fuzzy semantic typing. IEEE Trans. on Fuzzy Systems, 9(4), 2001, 483-496.

[125] Tim Berners-Lee, Information Management: A Proposal. CERN (March 1989, May 1990).

[126] Tim Berners-Lee, Mark Fischetti. Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor, ISBN 978-0-06-251586-5, HarperSanFrancisco, 1999

[127] Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 412–418, July 2004. Poster paper.

[128] Turney PD, Littman ML. Measuring praise and criticism: inference of semantic orientation from association. ACM Transactions on Information Systems, 21(4), 2003, 315-346.

[129] Turney PD, Littman ML. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. Technical Report EGB-1094, National Research Council Canada, 2002.

[130] Turtle H, Croft WB. Evaluation of an inference network-based retrieval model. ACM Trans. on Information System, 9(3), 1991, 187-222.

[131] Vasileios Hatzivassiloglou and Janyce Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In Proceedings of the International Conference on Computational Linguistics (COLING), 2000.

[132] W. Chen. New algorithm for ordered tree-to-tree correction problem. Journal of Algorithms, 40:135?158, 2001.

[133] W. Scott Means, Michael A. Bodie: The Book of SAX, No Starch Press, ISBN 1-886411-77-8

[134] Warren Sack. On the computation of point of view. In Proceedings of AAAI, page 1488, 1994. Student abstract.

[135] Whitelaw C, Garg N, Argamon S. Using appraisal groups for sentiment analysis. In Proc. of the 14th ACM Int. Conf. on Information and Knowledge Management, 2005, 625-631.

[136] Wiebe J, Bruce R, Bell M, et al. A corpus study of evaluative and speculative language. In Proc. of the 2nd SIGdial Workshop on Discourse and Dialogue, Vol.16, 2001, 1-10.

[137] Wiebe J, Riloff E. Creating subjective and objective sentence classifiers from unannotated texts. In Proc. of the 6th Int. Conf. on Computational Linguistics and Intelligent Text Processing, 2005, 486-497.

[138] Xue N. Chinese Word Segmentation as Character Tagging. Computational Linguistics and Chinese Language Processing, 8(1), 2003, 29-48.

[139] Yahoo! Research: "Web Spam Collections". Crawled by the Laboratory of Web Algorithmics, University of Milan.

[140] Yang Y, Chute CG. A linear least squares fit mapping method for information retrieval from natural language texts. In Proc. of the 14th Conf. on Computational Linguistics, 1992, 447-453.

[141] Yang Y, Pedersen JO. A comparative study on feature selection in text categorization. In Proc. of the 14th Int. Conf. on Machine Learning, 1997, 412-420.

[142] Yang Y. An evaluation of statistical approaches to text categorization. Journal of Information Retrieval, 1, 1999, 69-90.

[143] Yi J, Nasukawa T, Bunescu R, et al. Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques. In Proc. of the 3rd IEEE Int. Conf. on Data Mining, 2003, 427-434.

[144] Yorick Wilks and Janusz Bien. Beliefs, points of view and multiple environments. In Proceedings of the international NATO symposium on artificial and

human ntelligence, pages 147–171, New York, NY, USA, 1984. Elsevier North-Holland, Inc.

[145] Yorick Wilks and Mark Stevenson. The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation. Journal of Natural Language Engineering, 4(2):135–144, 1998.

[146] Ziv Bar-Yossef, Sridhar Rajagopalan. Template Detection via Data Mining and its Applications. In Proc. of he WWW'02 Conf., pages 580-591, 2002

[147] 陈博. Web 文本情感分类中关键问题的研究. [学位论文]. 北京邮电大学, 2008.

致谢

谨以此文献给我的三姑。我成长的每一个脚步，我现在和将来取得的每一点成绩都离不开她在我少年时代对我的照顾和教诲。在我最失落的时候，三姑也从来没对我丧失过信心。可就在我刚要取得一些成绩的时候，三姑却离我们而去了。就以这篇论文，聊表我的一点感激和思念吧。

感谢我的导师郭军教授，是他严谨扎实的治学态度、勤勉的工作作风、对学术问题孜孜不倦的追求精神，对我为学、为人都有着深刻的影响，而在今后这些影响将继续在我们身上发酵、升华，贯穿一生。我庆幸能在学生生涯的最后遇到这样的良师，感谢导师两年多来对我的悉心栽培，在此向他表示我最衷心的感谢和最诚挚的敬意。

感谢徐蔚然老师和陈光老师，是他们将我引入到如此迷人的信息检索领域。感谢两位老师对我的关心与帮助，也感谢他们为我们信息检索组的发展所付出的心血与贡献。

感谢陈博师兄和何慧师姐，以及茹昭、王倩、李倩、彭韬、许威等所有 PRIS 检索组的诸位师兄师姐，感谢你们帮助我如此之快的进入到检索领域，感谢你们不厌其烦的回答我的无数问题。

特别感谢 SIGBT 小组的周卉、李思、高晖吉同学，和你们一起努力的日子是我最宝贵的回忆，和你们一起参加的比赛是我最骄傲的成就。

感谢王肇刚、何川、翁云鹤、蒋鑫、方旭、王冠雄等同班同学，和你们在一起的这些狼狈不堪的日子，是我们最后的青春。这青春，没有遗憾。

感谢我的父亲母亲，是你们在生活上和学业上的关心、照顾和支持帮助我完成这将近 20 年的学生生涯。你们的期待是我人生中最大的动力。

新的道路正在眼前，我将满怀感恩走向前方

攻读学位期间发表的学术论文

[1] 杜磊，基于网页分析的Blog文本提取，中国科技论文在线，2009