分类号_____

U D C_____ 编号_____

中国科学院研究生院
硕士学位论文

基于自然语言处理和机器学习的文本分类及其应用研究

王 懿

指导教师 王晓京 研究员

中国科学院成都计算机应用研究所

申请学位级别 硕 士 学科专业名称 计算机软件与理论

论文提交日期 2006年4月12日 论文答辩日期 2006年6月14日

培养单位 中国科学院成都计算机应用研究所

学位授予单位 中国科学院研究生院

答辩委员会主席_____

分类号_____

U D C_____ 编号_____

中国科学院研究生院
硕士学位论文

基于自然语言处理和机器学习的文本分类及其应用研究

王 懿

指导教师 王晓京 研究员

中国科学院成都计算机应用研究所

申请学位级别 硕 士 学科专业名称 计算机软件与理论

论文提交日期 2006年4月12日 论文答辩日期 2006年6月14日

培养单位 中国科学院成都计算机应用研究所

学位授予单位 中国科学院研究生院

答辩委员会主席_____

NLP and ML Based Text Classification and Its Application

Dissertation Submitted to

Chengdu Institute of Computer Applications, Chinese Academy of Sciences

For the degree of

Master of Engineering

By

Yi Wang

Computer Software and Theory

Dissertation Supervisor:

Xiaojing Wang

基于自然语言处理和机器学习的文本分类及其应用研究

专　业：计算机软件与理论
研究生：王懿
导　师：王晓京

摘　要

本文讨论了基于自然语言处理和机器学习的文本分类任务，提出了文本分类中新的特征降维方法，并结合两种不同的机器学习算法，观察了不同的降维方法和机器学习算法相组合完成文本分类的性能随特征空间维数变化的现象，并尝试探讨了造成这种现象的原因。具体描述了完成分类任务所需要的自然语言处理、降维和机器学习的算法及其理论基础。设计了紧凑的数据结构和算法过程来实现本文提出的降维方法。分析了文本分类对搜索在效果和效率上的帮助，阐述了文本分类在信息过滤中的应用，并结合招聘信息服务系统的设想分析了其在主动信息服务方面的应用。本文还分析了文本分类各个阶段可能的改进发展方向。

文本分类分为两个阶段完成，分别采用了自然语言处理和机器学习的技术。因此，文本分类在理论研究上的价值体现在对这两种技术的推动。然而文本分类的意义远不如此。分本分类对于提高网上信息检索的效果和效率很有帮助，是推进个性化服务，改进信息获取模式的重要方面，也是内容安全的基础。因此好的分类性能是关注的焦点。研究文本分类任务的理论和工程问题，将具有重要意义。

关键字：特征降维；文本分类；自然语言处理；机器学习

NLP and ML Based Text Classification and its Applications

M.S. Candidate: Yi Wang (computer software and theory)

Directed by: Xiaojing Wang

Abstract

This dissertation mainly dicussed the Natural Language Processing (NLP) and Machine Learning (ML) based text classification, in which a new mean-variance based feature reduction was raised and anlyized. The relation between the classification effectivness and the dimension of feature space was examined, together with using two different learning methods in the second stage. Also, the NLP and ML were described including both the theoretical, algorithm aspect and the engineering aspect, in which an efficient data structure was given. The application of text classification in search engine and information filtering was discussed. Furthermore, the exact effectiveness and efficiency the TC can help the search engine gain and the application in personalized information push in an exact system of recruiting information push were discussed. The possible improvement was also given in this dissertation.

The text classification can be achieved in two stages, using NLP and ML correspondingly. Thus, evidently, TC has its theoretical value in pushing the research of NLP and ML. Its applications in search engine by improving its effectiveness and efficiency, in the personalized information push service, in the pattern of information gaining and content safety are also very meaningful. So, text classification has become an important task both in theory and engineering.

Key words: Feature Reduction, Text Classification, Natural Language Processing, Machine Learning

目 录

第一章 引言

1.1 文本分类的应用背景

今天，我们生活在两个世界中，一个是物理世界，另一个是信息世界。世界研究的对象不再仅仅包括物理世界的物质和能量，还包括另外重要的一维——信息。信息是什么，如何多快好省的存储信息，处理信息，传输信息和利用信息；围绕信息有一系列需要研究的科学和工程问题。而且，研究的成果直接与人们的工作生活息息相关。本文工作大的背景就是在如何利用信息方面。

如何利用信息，常常是容易被人们忽视的问题。因为这是信息科学技术和其他科学技术的交接部分。信息技术的研发领域产业人员通常认为，信息被电子化，处理（计算），传输，存储，是比较重要的工作，而信息的利用，自然留给使用信息的领域。信息的处理，有高速的计算机，先进的体系结构，精湛的制作工艺，高速的处理器，延迟不断减小，带宽容量不断增大，成本不断降低的内存；信息的存储有海量高保真的各种存储设备；信息的传输的速度，延迟，安全性，是通信和信息安全研究的内容；而信息的利用方面的研究和工程实践却因为科学技术的限制和人们的重视程度，在和信息其他方面的研究相比下处于落后状态。而近期，信息利用方面的研究逐渐成为热点，尤其是随着在它在商业领域中的应用，搜索引擎激烈竞争，商业智能兴起，日益受到重视。

信息的利用包括众多方面，比如图象处理技术帮助医务人员从医学影像中获得更多有价值的知识帮助诊断，数据挖掘在金融领域帮助从业人员从已有的股票数据中获得某种模式，分析影响走势的因素，模式识别在航空航天以及信息安全中的应用，都是利用计算技术帮助信息在各个应用领域能够被充分的分析，充足的应用，从而充分的发掘信息的价值。计算处理过的信息本身必须经过筛选，挖掘和分析，才能被充分利用，产生效益。本文工作的应用背景之一便是在网络世界中，从动态变化，海量大小的网页中获取信息，按用户的需求尽量准确的呈现给用户。这和当下搜索引擎的目标是一致的。搜索引擎拓展了传统的信息检索（IR Information Retrieval），因为传统的检索是在相对静止的文档库中进行，文档库的组织形式规范（很少存在文档间的链接），并且容量不太大（相比互联网上数

以亿计的网页），而且信息的形式主要是文本，不涉及图像、声音等多媒体信息。互联网上的搜索引擎还要考虑在大量用户的时变检索需求下，能够以用户可以接受的时间，（研究表明通常不超过 1 秒）[1]， 将符合用户检索需求的网页返回给检索用户，同时希望这个返回的网页列表尽量满足用户的信息需求。总之，就是在海量的，不断变化的网页信息库中，在尽量短的时间里返回尽量准确的网页列表给用户。

需要说明的是，在本文工作的主要时间，也就是 04 年夏到 06 年春的时间内，恰巧是搜索引擎研究从比较冷门发展到由产业界带动，商业引擎激烈竞争的过程。然而在这繁荣的背后，应该看到还有哪些领域的支撑技术是需要研究而研究的不足的。本文工作的主要任务----文本分类的研究和实现就是这样一个基于自然语言处理和机器学习的任务，它是提高网页搜索效果的新的切入点。因此文本分类在搜索引擎中的应用反映在引进新的方法提高检索的效果上。

除此以外，文本分类还在新的信息服务模式方面有重要应用。本文工作还给出了文本分类在特定的信息服务环境——邮件过滤和招聘信息提供方面——的应用。

1.2 文本分类的定义

文本分类是文本自动分类（ATC Automated Text Classification）的简称，是指用计算机程序自动确定指定文档和预先指定类别的隶属关系[2]。比如指定一篇文本属于体育类别，或者属于音乐类别，政治类别等预先设定好的一个或者多个类别。

完成文本分类主要有三大思路。一种最直观的方式是简单的匹配，比如某文本中类别词出现多就认为属于这个类别，这种方法比较粗糙，因为往往类别词是“元数据”性质的，因此并不一定会多次出现在属于这个类别的文本中。比如政治类别的文章可能很少出现“政治”这个词。 因此，第二种方法发展了第一种方法，统计类别元素下专家认为可能出现的词，认为如果这些词出现的次数多，则相应文本属于这个类别。可以认为这是一种基于规则的知识工程的方法，规则由专家制定，制定过程繁复并且容易出错，成本也很高。因此现在广泛采用的方法是统计学习的方法。本文工作也是采用这种机器学习的方法，同时，结合自然

语言处理的技术来完成文本分类，这正应和了本文的题目----基于自然语言处理和机器学习的文本分类。顺便指出，在本文的环境中对文本（text）、文档（document）不加区别的使用，在不关心网页中除了文字以外的其他对象，标记信息以及链接信息时，也不加区分的使用网页指代文本。

文本分类所针对的文本在使用的语言上有单一语种或者交叉语种的情况，本文工作在单一语种情况。而且不同语种的文本分类在预处理阶段的工作略有不同，会涉及到采用不同的自然语言处理技术。本文工作是针对汉语文本进行，因此需要先对文本进行分词，这在后面第二章会详细介绍。

定义中指出，文本由程序归属到不同的预先制定的类别。需要指出，分类的类别可以是层次树的结构关系，大类下又有小类。本文工作使用的是单一层次的类别模型。另外，“预先制定”的类别在目前还缺乏统一的标准，尤是在中文领域，而且“预先制定”不能完全适应网络应用背景下的变化，因此，对自动扩展类别的研究作为可能的改进方向将在本文第五章涉及。

综上所述，本文工作的任务是文本分类，采用的技术是自然语言处理、机器学习。针对的应用包括 1）在搜索引擎的应用背景下，探索信息检索效果的提高和新的可能信息服务模式；2）其他领域的拓展应用：以邮件过滤为例的信息过滤、信息内容安全方面的应用；以招聘信息服务为例的主动信息服务方面的应用。

作为引言，本章其余内容将介绍本文工作的总体框架，以及后续各个章节的安排。

1.3 本文工作

文本分类是本文工作的核心任务，完成它主要包括两个大的阶段，分别主要采用了自然语言处理的方法和机器学习的方法。第一个阶段的任务是把文本表示成向量形式，第二个阶段的任务是把训练文本集合的文本向量输入学习机器进行学习，训练学习机器。这样训练好的学习机器称为分类器，分类器接收新到文本的向量，输出这个向量对应文本的类别。最后，作为对文本分类效果的检验，统计这个分类器对测试文本的分类效果指标，作为对这个文本分类完成效果的评价和进一步改进的指导。其中，训练语料库是指人工搜集的正确指定了类别的文本

集合，文本和其所属的类别都输入学习机器进行训练；测试语料库是类别待学习机器进行指定的文本的集合，但给出了文本属于的类别，以供对比判断分类是否正确。

本文具体讨论了完成文本分类所需要采用的各种技术方法，其中包括作者独立提出的新的特征降维方法；展示了完成文本分类的逻辑过程；论述了实现的工程问题；描述了对分类效果的检验指标，分析了各种方法在各种维度下的性能差异；探讨了文本分类可能的改进方向；讨论了文本分类在提高搜索引擎搜索效果方面的应用以及在新的信息浏览模式和针对特定领域信息需求方面的应用。

1.4 本文的安排

下面各个章节将就文本分类工作的各个方面依次展开讨论。第二章将讨论文本分类第一阶段基于的技术，采用的方法，其中各个小的阶段的具体方法，包括本文工作独立提出的特征降维的方法；描述完成文本分类第二阶段的主要方法，重点是本文工作所采用的两种机器学习方法----Knn 和 SVM；陈述用于评价分类性能的方法和指标。第三章将在第二章宏观方法流程介绍的基础上，描述本文工作的文本分类器设计实现的主要工程问题；并且按第二章所述性能评价方法，参照其他的方法进行分类的效果，对本文工作的文本分类器的性能做出分析、对比和评价。第四章探讨文本分类在提高搜索效果和信息过滤、主动信息服务方面的应用。第五章展望了文本分类工作可能的改进，并对全文做了小结。最后是参考文献、发表论文列表和致谢。

第二章 文本分类的主要阶段、性能评价

2.1 概述

自动文本分类任务是对未知类别的文字文档进行自动处理，判别它们所属预定义类别集中的一个或多个类别。随着各种电子形式的文本文档以指数级的速度增长，有效的信息检索、内容管理、信息过滤、信息推送等应用变得越来越重要和迫切。文本自动分类是这些应用中一个有效的解决途径，已成为一项具有实用价值的关键技术。近年来，多种统计理论和机器学习方法被用来进行文本的自动分类，掀起了文本自动分类研究和应用的热潮。

历史上先后主要出现了两种研究体系来解决文本分类问题。一种是 70-80 年代的基于规则的方法，文本分类任务作为一种特殊的专家系统而出现。具体而言，就是先由专家根据自己的知识，制定很多用于分类文本的规则。用这些规则去计算文本应该属于的类别。这样方法的缺点是显然的，首先在于规则的制定是非常困难和难以检验的，从某种意义上说，这甚至比让专家自己去分类文本代价还大。另一种现在普遍使用的方法是学习的方法。在准备输入学习机器的向量时会结合到自然语言处理的方法，把文本表示成向量。这是机器学习和自然语言处理组合的一个很好的应用。

在这个方法体系下，把文本分类的任务分为两个阶段来完成。第一步是文本的处理。把文本表示成下一步进行分类计算所需要的向量形式。第二步是对这些代表文本的向量进行分类。这是一个典型的模式识别问题。可以采用多种机器学习的方法处理这个问题。这也是本文题目中使用“基于机器学习”的原因所在。比如，统计机器学习的代表支持向量机，比如最近邻居 KNN 算法，比如决策树，ＩＤ３，Ｃ４．５等等。具体参见 2.3 节。而第一步把文本计算成向量使用到了自然语言处理的技术。因为希望尽量使得向量能够“代表”文本，因而不是简单的使用文本中语言的最小编码单位，比如汉字和字母，而是希望尽量分析出文本语言所具有的词法结构，从而在一定意义上反映文本的语义。这是本文题目中基于“自然语言处理”的原因所在。这拓展了自然语言处理作为一个封闭体系在人机交互中的应用。特别的，作为中文文本分类，这种文本向量的计算的重要性更

加彰显。这也是本文工作——完成文本分类任务——的一个重要工作。具体参见2.2节。

完成文本分类器的训练后，就可以对输入的新到文本用同样的方法计算其向量形式输入训练好的学习器进行分类，最后输出相应文本的类别。而分类效果的好坏评价将采用一定方法和指标，这在2.4节中介绍。

2.2 文本分类第一阶段——文本表示成向量

2.2.1 概述

这个阶段的主要工作包括三个步骤，即，文本预处理，特征降维和实际计算文本在这些特征上的值，从而将文本表示成为向量。文本预处理是指把文本转化为原始特征空间中元素的序列。对于不同语言书写的文本，预处理过程和复杂程度不同。比如对于英语，预处理主要是去掉停用词，还原词形为词干，得到“干净”的文本。而对于中文，由于中文词语是连续书写，采用词语作为特征项需要先从连续的文本中分离出一个个的词语来，所以预处理阶段的主要工作是分词。预处理完成后，文本成为了词语的序列。下面一步就是对这些序列的词语空间进行降维，即减少要用来表示文本的特征词的数量，以降低计算的代价，同时去掉对于表征文本特征不重要甚至起反作用的词，提高整个分类的效果。 这样，就确定了用于表示文本的特征项。接下来的第三步工作就是按照确定的这些特征项的一种度量，计算文本在这些特征上这些度量下的值，最后形成文本的向量。

下面四个小节分别详细描述三个步骤涉及的技术和使用的方法，并详细描述本文工作具体所采用的方法和步骤。

2.2.2 中文文本预处理

中文预处理的主要工作是对文本进行分词。分词所采用的主要技术是自然语言处理。自然语言处理又常常不加区别的被称作自然语言计算，或者计算语言学。计算语言学的目标是利用计算机科学中的算法和数据结构建立一个语言的计算理论，并且可以利用这个计算模型分析和生成（自然）语言，从而可以实现更为有效和自然的人机交互。另一方面，在科学上的动机是为了对语言如何工作这个问题获得更好的理解。在本文涉及的工作中，我们只关注第一个方面的任务。而

且在这里其应用已被拓展，不再只是自成体系的人机交互，而是作为文本处理的过程被应用的。

自然语言处理的应用分为两大类，基于文本的应用和基于对话的应用。基于对话的应用是在预先的知识库中搜索用户问题的答案，并在前端使用语音识别的技术。这不在本文讨论之列。基于文本的应用主要在如下的应用领域进行：从文本库中找到与特定话题相关的合适文本（信息检索）；在文章中抽取与特定主题相关的信息（信息抽取），将文档从一种语言翻译成为另外一种语言（机器翻译），为特定目的对文本进行总结（自动文摘）。值得注意的是，可以使用其它简单的匹配技术来实现查找与特定话题相关的文章的任务，而不是用自然语言理解的技术。比如，可以查找含有与某个话题相关的关键词的文章，只要我们预先定义了该话题的关键词集合。这也是目前几乎所有搜索引擎所采用的方法，因为它实现简单，采用倒排文档的数据结构就可以解决。但是，很显然，这个系统没有“理解”文本，仅仅是使用了简单的匹配技术。我们希望为文章中的信息建立一种表示形式，然后用这种表示形式进行检索。也就是理解系统的关键特征，使得这种表示形式可以用于后续的处理。

自然语言处理分析的层次有语法，语义和语用。句子的句法结构指明了句子中的单词相互关联的方式。比如常常采用树形结构表示。句子的语义计算就是形成对句子语义的上下文无关的逻辑表示。逻辑形式对可能的词义进行编码，并给出词语和短语之间的语义关系。而语用分析就是在得到的句子语法结构（通常表示为树）和其语义关系的逻辑表示后，把这种表示映射为通用的知识表示（一阶谓词演算），以便系统在该应用领域完成适当的任务。本文工作中所用到的自然语言处理主要是指在文本的预处理阶段，因此主要涉及语言的语法分析。

词是最小的能够独立活动的有意义的语言成分[3]。在汉语中，词与词之间不存在分隔符，词本身也缺乏明显的形态标记，因此，中文信息处理的特有问题就是如何将汉语的字串分割为合理的词语序列，即汉语分词。英语文本是小字符集上的已充分分隔开的词串，而汉语文本是大字符集上的连续字串，因而分词是汉语自然语言处理的第一步。这是不同于其他语言的自然语言处理系统的重要特点，也是影响自然语言处理应用的重要因素。分词系统是中文信息处理中的一个主要组成部分，是中文自然语言理解、文献检索、搜索引擎以及文本挖掘系统中

最基本的一部分。汉字的简体／繁体转换、信息检索和信息抽取、搜索引擎、Web 文本挖掘、文本分类、文本校对等中文信息处理系统都首先需要分词作为其最基本的模块。目前，汉语自然语言处理应用系统处理的对象语料规模日增，因此分词的速度和性能变得更为重要。

要做好分词，首先总结处理的语言——汉语的特征，充分认识处理对象特征以更好的认识问题的难点和关键，以针对这些特征进行处理。

汉语是一种词根语，具有如下特点：(1)汉语缺乏形态变化，名词没有性、数、格的变化，动词没有形态的变化以表现不同的句子成分的句法关系，词本身不能显示与其他词的语法关系，其形式也不受其他词的约束。谓语动词也没有时态、语态、语气的变化以表达部分的背景语义；即汉语的语法关系不由词的曲折变化形态来反映，而主要由虚词来体现。这就使得中文语言的自动处理相比英语为代表的曲折语的自动处理更加困难。因为以词的形态变化来反映词性、语法关系甚至语义（比如谓语动词的不同时态反映了句子所描述动作发生的时间。谓语动词的语气变化反映了说话者的主观态度等等），是相对规范形式化的，这正适合计算机进行计算。而通过虚词来体现的句法关系规则不明显，没有特殊的标志，所以困难重重。(2)汉语语言单位没有明确的间隔，词和词之间没有间隔，甚至在引入西方语言的标点前，句子内部各个成分间，句子和句子间都没有间隔。而以英语为代表的曲折语的词和词之间有空格作为明确的间隔，因而在用计算机程序区分一个词时相对容易，主要的工作在于区分语言文本中的非语言词典中标准的词，比如未定义的新词，标点，数字，网址，缩略等。中文文本除了上述在以英文为代表的屈折语中的处理以外，其中更为重要和特别的工作在于在连续的中文文本中区分出一个一个的词。只有在中文连续字序列的基础上，得到了连续的词序列，才能进行语言的进一步处理，因为语法语义的分析都建立在词的基础上，很多需要自然语言处理的应用系统，比如本文的文本分类就需要词作为文本的特征，而不是字。因为词是最小的表义单位，建立在字基础上而不建立在词的基础上的中文处理系统就像把处理建立在字母而不是单词上的英文处理系统一样意义甚少。（3）中文书写的文本中保留有古汉语——文言的特征。比如单字成词，各种成语等等。由于文言特征的存在使得在分词中歧义的可能性大大增加。这不同于英文在语法分析中出现的歧义现象。（4）中文语言使用的形态的多样化相比

英语更突出。中文口语有很多种方言，也有很多种使用习惯，它们不能统一在标准的体系框架下。虽然狭义的自然语言处理并不涉及语音识别领域（因为按使用的专业知识已经很不相同了。语音识别主要是在信号处理领域。），因此不存在各种方言的语音不同带来的问题。但是由于各地使用语言的习惯和演化不同，也会在语言处理上面得到反映。尤其是在处理口语转化得到的文本和文学作品的文本时更是如此。比如，“遛弯儿”是北京地方色彩很浓的词，其意义就是“散步”，它很可能并不收入到词典中（实际上收录到词典中会使得词典更加庞大），但也要求语言处理系统能够识别出来。（5）中文的语法体系本身是从英文的语法理论体系中借用过来的。在套用同样的语法体系对语言文本进行分析本身就会存在这样那样的问题。因为这样的语法体系并不完全适合中文的特点。比如，对于词性的区分，由于英文同意义词在充当不同词性时的形态是不一样的，词性的概念就很必要。不同的词性承担不同的句子成分，在词性和句子成分之间存在比较明确的对应关系。而中文文本中词性和句子成分的概念就比较模糊，比如“检索”一词可以认为是动词，在“请检索一下“检索”“这句话中，第一个检索是动词做谓语，第二个检索是名词作宾语。也就是在固定词性和句子成分的对应关系时，一个词以同样的形态出现，词性却是变化的，这就比英文中固定的词性和词性-句子成分对应关系的特点下的词性标注要困难。

汉语的这些特征决定了针对其他语言处理的方法并不能完全适用于汉语信息处理。汉语信息处理又称中文信息处理，是指“用计算机对汉语的音、形、义等信息进行处理，包括对字、词、句、篇章的输入、输出、识别、分析、理解、生成等的操作与加工”[4]。汉语自动分词已成为众多中文信息处理任务的一项基础共性的研究课题。也是使用自然语言处理的各种应用系统几乎都会使用到的基本模块[5]。

下面陈述中文文本的词法分析——分词及词性标注的工作内容和任务。

中文文本的分词任务首先要完成的是在连续的单字序列中区分出分词词典定义的词，包括实词、虚词和成语、单字词。其次，要区分出词典中没有收录的未定义词、新词，甚至更进一步的要求做命名实体的识别。然后，还要针对汉语中的重叠词离合词以及附着在实词后的词缀做处理。同时也存在和英文词法分析一样的任务：在中文文本中区分出非中文符号，数字，网址，各种缩略等等。在

分析过程中，还要对各种歧义进行处理。最后，在分词的过程中还可以顺带完成词语的词性标注，为下一步的语法语义分析打下基础，也为基于自然语言处理的应用系统做好进一步深入分析的准备。

词典里有“高兴”一词，但是可能没有它的重叠形式，比如“高高兴兴”，或者其重叠形式有赖于自动分析获取添加入词典，这时分词系统要能够识别分析出来。重叠的形式除了上述的“AABB”形式，还有“ABAB”形式，比如“潇洒”一词的“ABAB”重叠形式为“潇洒潇洒”。另外还有例如“糊里糊涂”对“糊涂”这样的重叠形式也在分词系统完成的任务之列[6]。又比如，还有“黑黑”“黑压压”“黑不溜秋”等“新词”（词典中没有定义的词）是“黑”这个词典中定义了的词的不同的重叠形式。除了以重叠方式形成“新词”以外，还有离合现象。比如，游泳一词就可以这样用“小刚游了一下午泳”。又比如“吃饭”可以出现在“我吃过饭了”这样的句子中。这是分词中比较困难的而且是中文分词所独有的现象。很容易被切分成单字。如前所述，由于中文是通过虚词来表达语法关系的，这里用“过”插到“吃饭”一词中，表达“完成”的时态，这在英语为代表的曲折型语言中是通过词形的变化来反映的。比如“have eaten”,而不是在词之间插入其他的虚词这样“离合”的方式。还有一些词的构成是以词缀的方式。比如有的单字保留有很强的语义信息，语法上的限制又很少，因此构词灵活，有很强的构词能力。比如“非”,“性”等等。“党员”是词典中存在的词,“非党员”也应该要求我们的分词模块能够识别和区分出来。比如“先进”是词典中存在的词,“先进性”很可能在词典中没有，但也是希望分词模块能够完成的任务。这种词缀的构词方式表面上很类似于英文中的“派生”构词方法，但是二者对机器程序作词法分析造成的难度是不一样的。英文中的“派生构词法”多表现在相对固定和规则的构成上，词缀也相对固定，而且形成新词后仍然在形式上是一个整体，和其他词有名确的界定。只是进一步分析词的词根词缀而已，是更深入的词法分析，不会对分词造成（tokenization） 造成影响。而中文中能够构成新词的单字数量相对较大，因为决定一个单字是否能够成词的因素主要体现在这个构词成分（通称是一个单字）的语义上，因此比较灵活而难以用规则概括，不像英语为代表的曲折性语言构词的后缀前缀的语法意义很重，因此形式上相对容易描述和分析。在这些以前后缀形式加入原有词形成新词的方式中，能够成为前后缀的

词因为其潜在的语义(中文历史传统中文言的影响)而增加了形成新词的可能性。而且由于这些潜在的词缀本身作为一个单字有其意义,所以还容易造成切分的歧异。处理这种构词现象也是中文分词的任务。

除了以上的任务外,处理未定义词也是分词的任务之一。未定词可以定义为表达一个新概念而产生的词。这在词典中是没有的。而且随着社会的发展,时间的推移总会出现很多新的概念,随之产生新词。新词的处理是分词的任务之一。新词在英语为代表的曲折型语言中比较容易分析,因为通常用产生新的词根来产生一个新词的概率不是很大,通常的方式是在已有的词或词干前后加词缀或用已有的词干或词的组合来形成新词。比如“bloger”是一个新词,意为使用“blog”的人,是按照“名词+er=名词”的规则表示“使用...”的意义。而“超女”是一个新词,对应的词在英文中就是一个书写在一起的缩略词,很容易辨认分析。而因为中文中“超女”一词中的两个字都有潜在的单独的意义,所以存在和前后的字形成词而切分错误。这也是分词需要解决的问题和完成的任务。比如在“赶超女性”的片段中,就存在划分成“赶超”+“女性”的歧义情况。

以上这些任务是中文文本的词法分析所独有的任务,也是中文分词需要注意的难点。除此之外,造成分词任务困难之处还在于切分歧异的大量存在。比如在句子“我对他有意见”中,意见作为一个词应该划分在一起。而在“总统有意见他。”一句中,意见就不能作为一个词划分在一起,此时“意”已与前一个字“有”结合在一起形成一个词,而“见”单字成词[6]。歧义的现象有两种,一种称之为“真歧义”,另一种称为“伪歧义”。真伪的区分在于一个被区分出来的词是不是在真实语料中出现过。出现过为真,没有出现过为伪。按照歧义可能出现的方式,歧义分为交集型歧义,组合型歧义(覆盖型歧义)和混合型歧义。交集型歧义是指这样的情况:字串abc既可切分为ab/c,又可切分为a/bc。其中a,ab,c和bc均是词。如“有意见”,在“我对他有意见。”中切分为“有/意见”;在“总统有意见他。”中切分为“有意/见”,这也属于真歧义,也属于交集型歧义。组合型歧义是指,字序列ab为词,而a和b在句子中又可分别单字成词。例如:“马上”在句子“我马上就来。”中是一个词,在“他从马上下来。”中是不同的词。混合型歧义是指由交集型歧义和组合型歧义自身嵌套或两者交叉组合而产生的歧义。例如:“人才能”在句子“这样的人才能经受住考验。”中可以切分

为“这样的人才/能接受住考验”，也可以切分为“这样的人/才/能经受住考验。”歧义的处理也是分词所要处理的任务之一。

在完成分词的上述任务中，存在很多特有的难点，致使分词的效果不如英文为代表的曲折型语言的词法分析的效果。这些困难主要表现在：(1)词的定义不统一。一方面由于中文的历史传统，单字亦具有丰富的潜在的意义和语法能力，造成人们对词的定义并不完全认同。比如“吃饭”是作为一个词还是一个词组？另一方面，因为现有的语法体系基本上是五四运动后引进的英文的语法体系，并不完全适合中文，比如如前所述，词性与句法成分并不存在如英文那样明确的对应关系，造成对词的认同标准不一。调查表明，对于母语为汉语的被试者，对中文文本中词语的认同率只有70%[7]。虽然国家标准《信息处理用现代汉语分词规范》给出了词和分词单位的非形式定义，但是语言学界对词还没有给出一个为大家广泛接受的、严格且统一的非形式定义。词的形式定义或者抽象定义问题也没有完全解决。(2)汉语的分词还没有形成一个公认的分词标准。这是由第一个问题引起的问题。同一文本可能被不同的人划分为几种不同的分词结果。因而分词系统的性能就很难得统一客观的评价。(3)词的具体判定问题还没有完全解决。尽管《信息处理用现代汉语分词规范》提出了分词单位和一套比较系统的分词规则，但是由于真实文本的复杂性和多样性，实践与理论之间的重大差异，仍然没有能够在词层解决问题。问题的实质在于分词规范和分词词表的构造应该和汉语真实语料库结合起来考虑。同时，除了定性信息外，还必须引入定量信息。另外在计算上也存在较大困难：(1)缺乏合理的自然语言形式模型；(2)如何有效地利用和表示分词所需的语法知识和语义知识；(3)如何对语义进行理解和形式化。而且以上问题的解决都要考虑到现实的可计算性和计算复杂性问题。

做分词主要有以下三种主要的方法：基于词典的方法、基于统计的方法和混合方法[6]。基于词典的分词方法的三个要素为分词词典、文本扫描顺序和匹配原则。文本的扫描顺序有正向扫描、逆向扫描和双向扫描。正向扫描是指从待切分语句的开头开始扫描，而逆向扫描是指从待切分语句的末尾开始扫描。双向扫描是正向扫描和逆向扫描的组合。匹配原则主要有最大匹配、最小匹配、逐词匹配和最佳匹配。最大匹配法的基本思想是：(1)取待切分语句的 m 个汉字作为匹配字段，其中 m 为机器可读词典中最长词条的汉字个数；(2)查找机器可读词典并

进行匹配。若能匹配，则将这个匹配字段作为一个词切分出来；若不能匹配，则将这个匹配字段的最后一个字去掉，剩下的字符串作为新的匹配字段，进行再次匹配。重复以上过程，直到切分出所有词为止。最小匹配法的基本思想是使待切分语句分词后得到的词最少。逐词匹配法是指把词典中的词按由长到短的顺序在待切分语句中进行搜索和匹配，直到把所有的词都切分出来为止。最佳匹配法的基本思想是词典中的词条按照词频的大小顺序排列，以求缩短分词词典的检索时间，从而降低分词的时间代价。基于词典的分词方法的优点是易于实现。其缺点是：(1)匹配速度慢；(2)存在交集型和组合型歧义切分问题；(3)词本身没有一个标准的定义，没有统一标准的词集；(4)不同词典产生的歧义也不同。对于基于词典的分词方法，影响其精度的因素有 ：(1)机器词典中词目的选择和词条的数量；(2)机器可读词典与待切分文本中词汇的匹配关系；(3)切分歧义；(4)未登录词；(5)分词方法。词典对分词精度造成的影响远远大于分词方法本身产生的歧义切分错误和未登录词问题。影响其速度的因素有：机器可读词典的组织结构、匹配的原则和扫描的顺序。

基于统计的分词方法所应用的主要的统计量或统计模型有：互信息、N 元文法模型、神经网络模型、隐 Markov 模型（HMM）和最大熵模型等。这些统计模型主要是利用词与词的联合出现概率作为分词的依据。基于统计的分词方法的优点是：(1)不受待处理文本的领域限制；(2)不需要一个机器可读词典。缺点是：(1)需要大量的训练文本，用以建立模型的参数；(2)该方法的计算量都非常大；(3)分词精度与训练文本的选择有关。下面分别例举基于统计的几种方法。

(a)互信息

互信息是一种度量不同字符串之间相关性的统计量。对于字符串X和Y，其互信息的计算公式如下：

$$\mathrm{MI(x,y)} = \log_2 \frac{p(x,y)}{p(x)p(y)} \qquad (1)$$

其中 $p(x,y)$ 为字符串 X 和 Y 共现的概率，$p(x)p(y)$ 分别为字符串 X 和 Y 出现的概率。

互信息 $\mathrm{MI(x,y)}$ 反映了字符串对之间结合关系的紧密程度 ：(1)互信息 $\mathrm{MI(x,y)} \gg 0$，则 X，Y 之间具有可信的结合关系，并且 $\mathrm{MI(x,y)}$ 越大，结合程度

越强；(2)互信息 $MI(x,y)\approx 0$ 则 X，Y 之间的结合关系不明确；(3)互信息 $MI(x,y)\ll 0$ 则 X，Y 之间基本没有结合关系，并且 $MI(x,y)$ 越小，结合程度越弱。

(b)N元文法模型

N元文法模型可用这样的公式表达。

$$W^{*}=\arg\max_{w}P(W)=\arg\max_{w}p(w_1...w_{N-1})\prod_{i=N}^{l}p(w_i\mid w_{i-N+1}...w_{i-1}) \qquad (2)$$

典型地，取 N=2，为二元语法。高军提出了一种无监督的动态分词方法，采用了一种可变长的 N 元文法模型[8]。它以词信息理论中极限熵的概念为基础，运用汉字字符串间最大似然度为匹配原则。

(C)神经网络模型

尹锋提出了一种神经网络的分词方法[9]。韩客松提出了一种无词典的分词模型系统。其思想是：(1)采用人工神经网络的方法解决汉语歧义字段切分存在的问题；(2)分析网络结构和两类学习算法(BP 和概率网 PNN)对歧义词切分的影响[10]。该方法需要进一步研究下面的问题：(1)分析大样本中隐含层节点数、网络层次、样本数量、和学习次数等因素对分词精度的影响；(2)当语料庞大时，对分词精度的测评；(3)分词的速度与语料大小的关系。基于神经网络方法的优点是：(1)神经网络具有自学习、自组织、并行、非线性处理方式等特点，从而使该方法具备知识表达简洁、学习功能强、开放性好、知识库易于维护和新的优势；(2)分词速度快；(3)精确度较高。缺点是：(1)容易陷入局部极小值点；(2)学习算法收敛速度慢；(3)网络层数及隐含节点选取无确定原则；(4)新加入样本对已学完样本有一定影响。

(d)隐Markov模型

李家福提出了一种基于 Markov 模型的分词系统，主要利用了汉语词长的分布规律[11]。但没有给出实验语料的来源和选取准则，以及使用统计方法的分词结果(包括分词的有效性和准确程度)能够完全满足文本分类需要的原因。Masao Utiyamaf 等人提出了一个与领域无关的、不需要训练数据的文本切分统计模型，以找到文本的最大概率的切分。这种方法的优点是：(1)不需要训练数据；(2)能够利用切分的特征信息，例如切分的平均长度。但是这种方法把文本切分为较少的部分，不能将文本切分为很多部分。基于隐 Markov 模型的分词方法的优点是：

降低了未登录词和专有名词的影响，只要有足够的训练文本就易于创建和使用。本文工作完成文本分类任务的第一阶段文本计算的第一步就是要对中文文本进行预处理。预处理的主要工作就是对连续的中文文本进行分词——把字串转化为词串。采用的方法是层叠隐马尔科夫模型。

时间和状态都离散的马尔科夫过程称为马尔科夫链[12]。一阶马尔科夫链，满足 $p[q_t = j \mid q_{t-1} = i, q_{t-2} = k, ...,] = p[q_t = j \mid q_{t-1} = i]$。且满足下面的假设：两个相邻状态 q_{t-1}，q_t 发生转移的概率 a_{ij} 与时间无关，一旦由样本数据训练好，就不再随时间发生变化，即假设从训练样本中得到的 $a_{ij} = p[q_t = j \mid q_{t-1} = i]\ 1 \le i, j \le N$ 适用于新的测试样本。其中，N 是系统状态总数，并且满足下面的标准随机约束条件：1） $a_{ij} \ge 0$，$\forall i, j$ 2） $\sum_{j=1}^{N} a_{ij} = 1$，$\forall i$。也就是说，任何时候，当前状态 q_t 只与前面相邻的一个状态 q_{t-1} 有关，而与其他状态无关。所以一阶离散马尔科夫链被叫做“无记忆随机过程”。如果当前状态与前面两个状态都相关，称这种模型为二阶马尔科夫过程，以此类推。由于二阶马尔科夫链计算量大，所以一般在应用中采用一阶马尔科夫链。

隐马尔科夫模型 HMM 起源于 20 世纪 60 年代后期的隐马尔科夫链，属于信号理论模型[13]。马尔科夫模型相对隐马尔科夫模型过于简单，应用有限，因为马尔科夫模型中一个测值对应一个状态值。但是在实际中，很多观测现象往往不是一维的，所以每个观测不能直接对应一个状态，而是要经过一个映射，将多维观测向量映射到一个状态中。因此在观测向量和状态之间存在一个映射函数，这时，状态被映射函数隐藏起来，观测不能直接见到状态，这就是“隐”马尔科夫模型的起源。

隐马尔科夫模型用下面的五元组定义。$O = (o_1, o_2, ...o_T)$ 其中，N 表示一个隐马尔科夫模型所包含的状态总数，即有状态集合：$\{S_1, S_2, ...S_N\}$。在隐马尔科夫模型中，每个状态被隐藏起来，比如在分词中，这个状态就是一个一个的词，或者非中文的 token； 在标注词性时就是所有的词法中的词语类别，比如名词，动词等等，都看作是在词性标注时的状态。因为，程序看到的是词语序列，其词性

序列作为词语序列背后的隐藏的状态是不可见的。

M 为每个状态对应的观测事件数。在分词中就是由中文字符组成的中文文本的长度。比如组成“所以”这个词的两个字“所”和“以”是观测到的两个事件，他们都对应“所以”这个词的状态。

$A=\{a_{ij}\}$ 是状态的转移矩阵。其中 $a_{ij}=p[q_{t+1}=j \mid q_t=i]$（$1\le i,j\le N$）表示在时刻t 状态 q_t 为i，在时刻t+1 状态 q_{t+1} 为j，从状态 q_t 转移到 q_{t+1} 的概率为 a_{ij}。因此 A 是一个 $N\times N$ 矩阵。

B=$\{b_j(k)\}$ 为观测时间对应的状态分布。其中 $b_j(k)=P[o_t=V_k \mid q_t=j]$（$1\le k\le M$），表示时刻为 t 状态为 j 时，观测事件是 V_k 的概率。每个时刻观测事件所对应的状态就构成了 B。因此 B 是一个 $N\times M$ 的矩阵。

π 为起始状态概率。它对应一个观测序列 $O=(o_1,o_2,...o_T)$ 起始时刻 o_1 位于某个状态的概率。

隐马尔科夫模型有三个基本问题，不少应用都可以归结为这三个基本问题。

问题 1——估计问题（Evaluation），给定模型 $\lambda=(N,M,A,B,\pi)$ 和观测序列 $O=(o_1,o_2,...o_T)$，求出在模型 λ 下，观测事件序列 O 发生的概率 $p(o\mid\lambda)$。这样可以进一步的对观测序列 O 进行识别和归类。

问题 2 给定模型 $\lambda=(N,M,A,B,\pi)$ 和观测序列 $O=(o_1,o_2,...o_T)$ 求出 $q=(q_1,q_2...,q_T)$ 对应的最佳状态序列 $q=(q_1,q_2...,q_T)$。即，计算出一个对应于这个观测序列的最佳状态序列。最佳状态序列计算采用 Viterbi 算法。这个算法基于算法设计中动态规划的思想。

问题 3 通过训练样本得到模型 $\lambda=(N,M,A,B,\pi)$ 的参数。

(e) EM模型

李家福提出了一种基于 EM 算法的分词方法，EM 算法是在极大似然原则下的一种建模方法，存在模型和数据的过度拟合问题[14]。

(d) 关联词统计语言模型

金凌提出了一种距离加权的关联词统计语言模型，通过距离加权函数来引入

距离信息。其平滑方法采用了基于图灵估计的退化算法[15]。该方法应用到了一个中文整句拼音输入法系统中。实验结果表明该模型比 N—gram 统计语言模型的性能有了一定的提高，但是汉字的识别率有所降低。该方法的优点是：在关联模型中只保存词对之间的关系。随着 M 的增大，只需增加一部分在语料中距离不大于 N-1 且没有出现过的词对之间的关系，模型的规模大小只是略有增加。M 可远大于 N，从而使模型具有更好的性能。由于 N-gram 模型保存的是各个 N 元组的相关信息，随着 N 的增大，N-gram 模型的规模将呈指数级别增长。该方法的缺陷是：(1)关联词模型预测每一个词出现的概率依赖于前面的词；(2)训练语料的大小限制了模型对这些词之间关系的准确表达。

下面再重点介绍本文工作中使用的分词方法的核心——隐马尔科夫模型 HMM。

Markov 链。 Markov 模型最早由 Andrei A. Markov 提出。其最早的应用就是在 1913 年由 Markov 将其应用在为俄文单词字母序列建模。以后就发展成为一种通用的概率模型（工具）。通常我们考虑的随机变量序列中的随机变量并不是独立的。每个随机变量依赖于序列中的前一个随机变量。在这样的系统中，我们可以假设只需要知道当前的随机变量的取值就可以预测下一个随机变量的取值，而不必知道以前的随机变量的信息。也就是说，给出序列中当前随机变量，序列中未来的随机变量与序列中以前的随机变量条件独立。（Conditionally Independent）

假设，$X=(X_1,...,X_T)$ 是一个随机变量序列，其取值范围是一个有限集，称为状态空间 $S=\{s_1,...,s_N\}$，那么马尔科夫性是指：

1）有限视界（Limited Horizon）：

$$P(X_{t+1}=s_k \mid X_1,...X_t)=P(X_{t+1}=s_k \mid X_t) \tag{3}$$

2）时不变性 （Time invariant (stationary)）：

$$P(X_{t+1}=s_k \mid X_1,...X_t)=P(X_2=s_k \mid X_1) \ =P(X_2=s_k \mid X_1) \tag{4}$$

称X为马尔科夫链，若它具有马尔科夫性。

马尔科夫链可以通过下面的转移矩阵 A 进行定义，

$$P=\begin{pmatrix} P_{11} & P_{12} & \cdots & P_{1j} & \cdots & P_{1N} \\ P_{21} & P_{22} & \cdots & P_{2j} & \cdots & P_{2N} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ P_{i1} & P_{i2} & \cdots & P_{ij} & \cdots & P_{iN} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ P_{N1} & P_{N2} & \cdots & P_{Nj} & \cdots & P_{NN} \end{pmatrix} \tag{5}$$

其中 $a_{ij}=P(X_{t+1}=s_j \mid X_t=s_i), a_{ij}\geq 0, \forall i,j$ 并且 $\sum_{j=1}^{N} a_{ij}=1, \forall i$。初始状态取得的概率由 π 给出。$\pi_i=P(X_1=s_i)$ 且 $\sum_{i=1}^{N}\pi_i=1$。

由上所述知，二元模型也就是马尔科夫链。实际上，n 元模型（n-gram）也是一种马尔科夫链。虽然表面上，未来状态依赖于当前及过去的 n-1 个状态，不符合 Limited Horizon 条件，但是可以把前 n-1 个状态包含进一个状态，这样只要 n 是有限值，则任何 n 元模型可以转化为 n-1 阶马尔科夫链。

马尔科夫链可以应用到任何线性序列时间的概率模型。它还可以用下面所示的状态图来表示马尔科夫链。

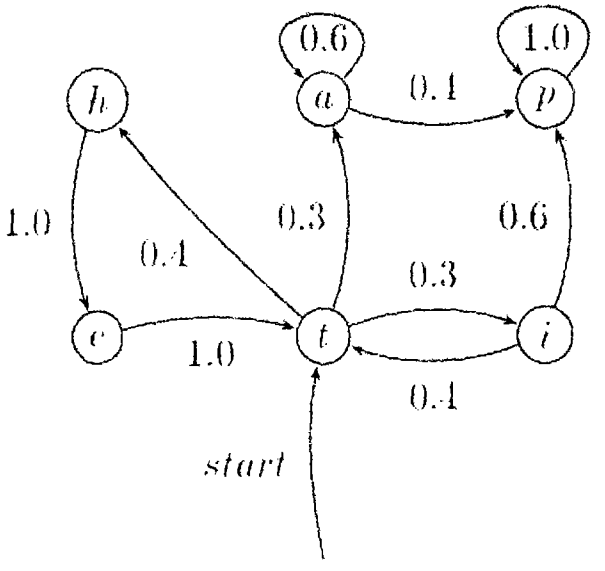


图 1 马尔科夫链的状态图表示

如图1，状态用圈表示，start箭头指向的状态为开始的初始状态。可能的状态间的转换在状态间用箭头表示，在箭头旁注明了相应从箭尾表示状态到箭头表示状态的转移的概率。对每个状态，从它射出的箭头上的概率和为1，对应表示上面所述的马尔科夫性的$\sum_{j=1}^{N} a_{ij} = 1, \forall i$条件。马尔科夫链这样的表示可以看出马尔科夫链可以被认为是不确定的有限状态自动机，扩展了有限状态自动机的状态转移，从绝对确定的转移变为概率的转移。马尔科夫性保证了这样的自动机的状态的有限性。

在马尔科夫模型中，已知机器历经的状态，所以，状态序列或者是它的函数可以认为是马尔科夫模型的输出。一种状态序列的概率，也就是一随机变量序列$X = (X_1, ..., X_T)$的概率可以这样在马尔科夫模型中计算。

$$\begin{aligned} &P(X_1, ..., X_T) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1, X_2)...P(X_T \mid X_1, ..., X_{T-1}) \\ &= P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2)...P(X_T \mid X_{T-1}) \\ &= \pi X_1 \prod_{t=1}^{T-1} a_{X_t X_{t+1}} \end{aligned} \tag{6}$$

下面进一步介绍隐马尔科夫模型。

在隐马尔科夫模型中，不知道模型历经的状态序列，但是已知状态的某种状态函数。在隐马尔科夫模型中，可以认为是背后的状态概率地产生了表面的事件。一个很广泛的应用就是做词性标注，为文本中的每个词标注其词性 POS（Part of Speech）。设想这里背后的状态就是词性的序列，按这个序列生成文本中实际的单词序列，也就是隐马尔科夫模型中观察到的值的序列。隐马尔科夫模型的使用除了体现在适用性广，还表现在便于训练和计算。训练数据同时给出了状态和观察值，因此隐马尔科夫模型的结构就定了，只是依据所给的大量训练数据计算出隐马尔科夫模型中的参数，也就是状态转移概率和状态到观察值的概率。

在本文工作的应用中，训练数据就是大规模的语料。为了保证通过语料中出现的词序列的共现现象的统计能够作为隐马尔科夫模型中的概率参数，我们还假设这个语言现象的频率收敛于概率[16]。

Shai Fine,进一步提出的层次隐马模型(Hierarchical Hidden Markov Model，简称HHMM)的思想。在HHMM中，有多个状态层和一个输出层。每一个上一层状态都对应于若干个下一层的子状态，而每个状态的子状态的分布都是不同的，由

一个隶属于该状态的初始子状态概率矩阵和子状态转移概率矩阵所决定。最底层状态通过一个输出概率矩阵输出到观察值[17]。HHMM的解码问题求解的时间复杂度是$O(NT^3)$，而HMM的解码问题求解的时间复杂度只有O(NT)。

本文工作所采用的开源 ICTCLAS 分词系统采用的类似层次隐马模型的层叠隐马尔可夫模型（Cascaded Hidden Markov Model，简称 CHMM）。不同于 HHMM 的是，CHMM 实际上是若干个层次的简单 HMM 的组合，各层隐马尔可夫模型之间以下面几种方式互相关联，形成一种紧密的耦合关系：各层 HMM 之间共享一个切分词图作为公共数据结构；每一层隐马尔可夫模型都采用 N-Best 策略，将产生的最好的若干个结果送到词图中供更高层次的模型使用；低层的 HMM 在向高层的 HMM 提供数据的同时，也为这些数据的参数估计提供支持。整个系统的时间复杂度与 HMM 相同，仍然是 O(NT)。下面简述用于本文工作进行文本分类及应用的预处理过程——分词模块的工作原理和框架。

如图 2 所示，首先进行原子切分。原子切分是词法分析的预处理过程，主要任务是将原始字符串切分为分词原子序列。分词原子指的是分词的最小处理单元，在分词过程中，可以组合成词，但内部不能做进一步拆分。分词原子包括单个汉字，标点以及由单字节、字符、数字等组成的非汉字串。如“2002.9，ICTCLAS 的自由源码开始发布”对应的分词原子序列为“2002.9/，/ICTCLAS/的/自/由/源/码/开/始/发/布/”。在这层 HMM 中，终结符是书面语中所有的字符，状态集合为分词原子，模型的训练和求解都比较简单，就不再赘述。

我们可以把所有的词按照图 3 分类，其中，核心词典中已有的每个词对应的类就是该词本身。这样假定核心词典中收入的词数为|Dict|，则我们定义的词类总数有：|Dict|+6。

给定一个分词原子序列 S，S 的某个可能的分词结果记为 $W=(w_1,\ldots w_n)$，W 对应的类别序列记为 $C=(c_1,\ldots c_n)$，同时，我们取概率最大的分词结果 W#作为最终的分词结果。则：

$$W\#=\arg\max_{W} P(W) \tag{7}$$

利用贝叶斯公式进行展开，得到：

$$W\#=\arg\max_{W} P(W|C)P(C) \tag{8}$$

将词类看作状态，词语作为观测值，利用一阶 HMM 展开得：

$$W^{\#} = \arg\max_{W} \prod_{i=1}^{n} p(w_i \mid c_i) p(c_i \mid c_{i-1}) \tag{9}$$

为计算方便，常用负对数来运算，则：

$$W^{\#} = \arg\min_{W} \sum_{i=1}^{n} [-\ln p(w_i \mid c_i) - \ln p(c_i \mid c_{i-1})] \tag{10}$$

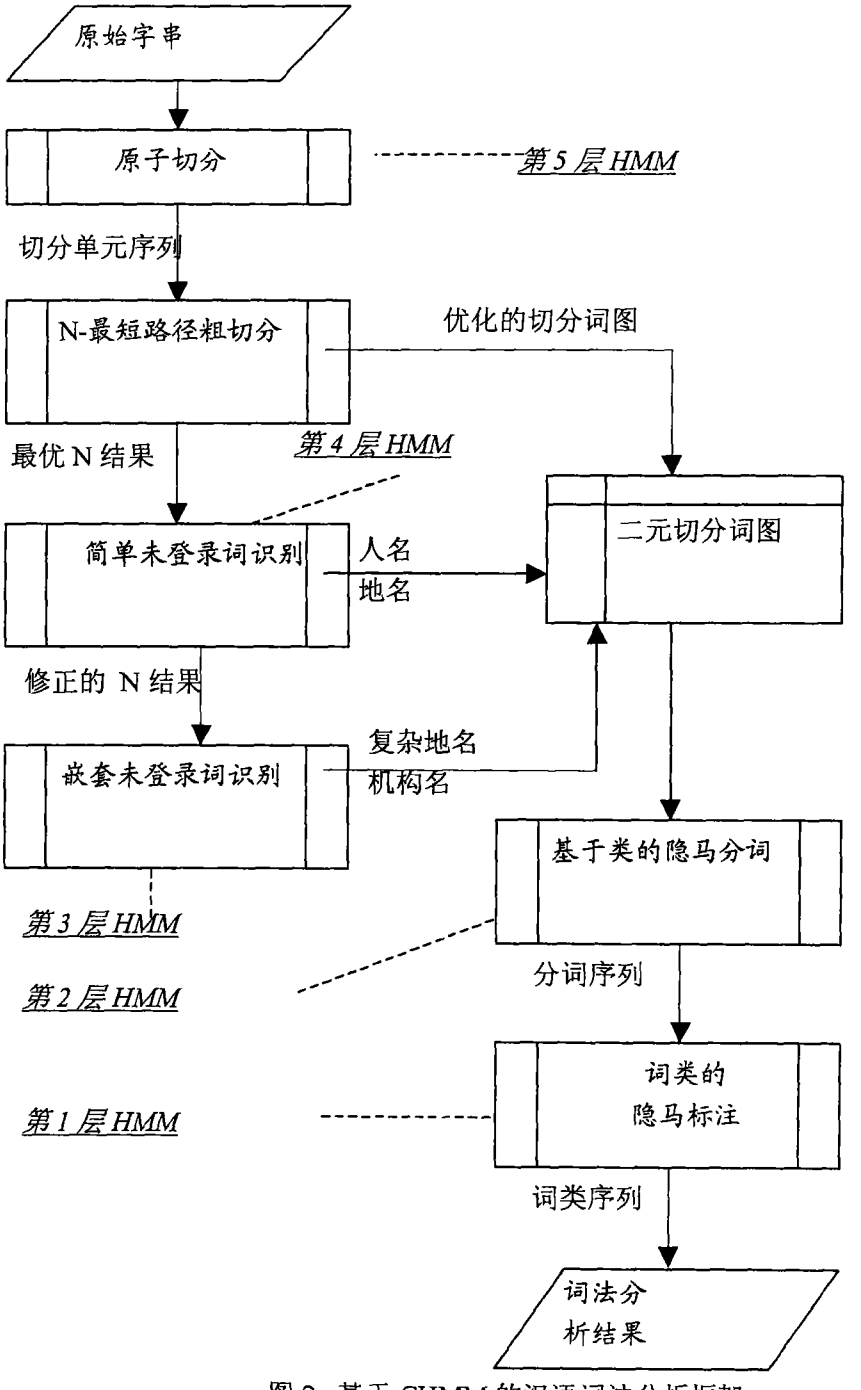


图 2. 基于 CHMM 的汉语词法分析框架

$c_i = \begin{cases} w_i & \text{iff } w_i \text{ is listed in the segmentation lexicon;} \\ \text{PER} & \text{iff } w_i \text{ is unlisted* personal name;} \\ \text{LOC} & \text{iff } w_i \text{ is unlisted location name;} \\ \text{ORG} & \text{iff } w_i \text{ is unlisted organization name;} \\ \text{TIME} & \text{iff } w_i \text{ is unlisted time expression;} \\ \text{NUM} & \text{iff } w_i \text{ is unlisted numeric expression;} \\ \text{STR} & \text{iff } w_i \text{ is unlisted symbol string;} \\ \text{BEG} & \text{iff beginning of a sentence} \\ \text{END} & \text{iff ending of a sentence} \\ \text{OTHER} & \text{otherwise} \end{cases}$

* "unlisted" is referred as being outside the lexicon

图 3 分类词表

根据图 3 中类 c_i 的定义，如果 wi 在核心词典收录，可以得到 $c_i=w_i$，因此 $p(w_i|c_i)=1$。在分词过程中，我们只需要考虑未登录词的 $p(w_i|c_i)$。最终所求的分词结果就是从初始节点 S 到结束节点 E 的最短路径，这是个典型的最短路径问题，可以采取贪心算法快速求解。

在实际应用基于类的分词 HMM 时，切分歧义能否在这一模型内进行融合并排解是一个难题；另外一个关键问题还在于如何确定未登录词 w_i、识别其类别 c_i 并计算出可信的 $p(w_i|c_i)$。

于是，接下来的工作主要有 N-最短路径的切分排歧，未登录词的隐马识别，未登录词识别角色表和嵌套未登录词的识别。

未登录词识别的任务有：1)确定未登录词 w_i 的边界和类别 c_i ；2)计算 $p(w_i|c_i)$。在 N 个候选切分结果的词类序列基础上，引入了高层 HMM 来实现未登录词的识别。和基于类的隐马分词模型类似，我们对初始切分得到的各个词按照其在未登录词识别中的作用，进行分类，并将词所起的不同作用称为角色。与隐马分词中定义的类相比，角色不同的是：类和词是一对多的关系，而角色与词是多对多的关系，即：一个词可以充当多个角色，而一个角色也可以对应多个词。

对于一个给定的初始切分结果 W=(w1,...wn),在一个角色集合的范畴内，假定 R=(r1,...rn)为 C 的某个角色序列。我们取概率最大的角色序列 R#作为最终的角色标注结果。和隐马分词的推导过程类似，我们最终可以得到：

$$R^{\#} = \arg\min_{R} \sum_{i=1}^{n} [-\ln p'(w_i \mid r_i) - \ln p(r_i \mid r_{i-1})] \qquad (11)$$

(其中r_0为句子的开始标记BEG)

R#可以通过Viterbi算法选优得到；在最大概率角色序列的基础上，我们可以简单的通过模板匹配实现特定类型未登录词的识别。识别出来的未登录词为w，类别为c，利用隐马过程可以得到：$p(w|c)=\prod_{j=0}^{k} p(w_{p+j}|r_{p+j})p(r_{p+j}|r_{p+j-1})$。其中$w_i$由第p，p+1...p+k-1个初始切分单元组成。

最后，识别结果及其概率加入到二元 HMM 切分图中，和普通词一样处理，竞争出最佳结果。

复杂地名和机构名往往嵌套了普通无嵌套的人名、地名等未登录词，如“张自忠路”、“周恩来和邓颖超纪念馆”。对于这种嵌套的未登录词，做法是：在低层的 HMM 识别过程中，先识别出普通不嵌套的未登录词，然后在此基础上，通过相同的方法采取高层隐马模型，通过角色标注计算出最优的角色序列，在此基础上，进一步识别出嵌套的未登录词。以切分序列片断“周/恩/来/和/邓/颖/超/纪念馆”为例，先识别出“周恩来”和“邓颖超”为人名 PER，得到新的词类序列“PER/和/PER/纪念馆”，最终就可以识别出该片段为机构名。这样的处理优点在于能够利用已经分析的结果，并降低数据的稀疏程度。

用来训练 HMM 角色参数的语料库是在北大计算语言所切分标注语料库的基础上，甄别出各种类型的未登录词之后，自动转换得到的[18]。

这样文本预处理就完成了，形成的文本的词语集合。

2.2.3 经典的特征降维方法

分类问题的最大特点和困难之一是特征空间的高维性和文档表示向量的稀疏性。在中文文本分类中，通常采用词条作为最小的独立语义载体，原始的特征空间由可能出现在文章中的全部词条构成。而中文的词条总数有十多万条，这样高维的特征空间对于几乎所有的分类算法来说都偏大。所有的文本都要按这个特征空间来计算自己在其上的值，因此特征降维在计算上对降低计算代价有很重要的作用。把文本转换为向量不仅针对训练文本集合中的文本，新到的文本也要首先表示成向量，因此降低向量的维数不仅可以降低训练学习器整个过程的时间，而且，也会缩短在具体判断新到文本属于类别的时间。有效降低计算代价对于实际的工程商业应用十分重要，这也成为推动文本分类在搜索引擎中应用的重要因

素。除了降低计算代价方面的作用以外，好的特征降维会有效去除噪音，使得保留的特征能够更集中的体现文本的特征，从而提高分类的效果。总之，寻求一种有效的特征抽取方法，降低特征空间的维数，提高分类的效率和效果，成为文本自动分类中需要首先面对的重要问题。

目前用于降维的思路主要有两种：一种是把特征转换到新的特征空间,即计算旧的特征的某种函数，形成新的特征，例如LSI (Latent Semantic Indexing) ；另一种是从原始的特征空间中选取一个子集即，特征选择[19]。在特征选择思路下，又有两种主要方法，一种是wapper，一种是过滤（评价函数）。Wapper的方法是通过机器学习的方法学习得到评价函数。评价函数的方法是设计特征上的某种针对分类性能的指标函数，以此指标函数的值作为过滤特征的标准。本文工作采用特征过滤的方法进行特征降维。特征过滤的核心是建立一个指标，或者称为评价函数，过滤函数，用来度量特征，以确定特征是否应该被保留或去除。重要特征的评价函数值大，因而特征得以保留，成为文本向量中的一维属性[20, 21, 22, 23]。

常常使用的评价函数有文本频率 Document Frequency (DF), Chi-square (CHI), 信息增益 Information Gain (IG), 互信息 Mutual Information (MI), Term Strength (TS), GSS Coefficient, Odds Ratio, 等[24]. Yang 等在 Reuters21578 语料库上试验了前面五种方法，认为 DF, CHI, and IG 更为有效[25, 26]。

1. 文档频率方法 DF

词条的文档频率是指在训练语料中出现该词条的文档数。文档频率是最简单的特征抽取技术，由于其具有相对于训练语料规模的线性计算复杂度，它能够容易地被用于大规模语料统计。采用 DF 作为特征抽取基于如下基本假设：DF 值低于某个阈值的词条是低频词，它们不含或含有较少的类别信息。将这样的词条从原始特征空间中移除，不但能够降低特征空间的维数，而且还有可能提高分类的精度。但是在信息检索研究中却通常认为 DF 值低的词条相对于 DF 值高的词条具有较多的信息量，不应该将它们完全移除。在实际运用中常把它作为评判其它评估函数的标准。

2. 信息增益IG

对于词条 t 和文档类别 c，IG 考察 c 中出现和不出现 t 的文档频数来衡量 t 对于 c 的信息增益。我们采用如下的定义式：

$$IG(t)=p(t)\sum_{i=1}^{m}p(c_i|t)\lg\frac{p(c_i|t)}{P(c_i)}+p(\bar{t})\sum_{i=1}^{m}p(c_i|\bar{t})\lg\frac{p(c_i|\bar{t})}{P(c_i)} \quad (12)$$

式中：$p(t)$表示语料中包含词条t的文档的概率；$p(c_i|t)$表示文档包含词条t时属于c_i类的条件概率；$p(\bar{t})$表示语料中不包含词条t的文档的概率；$p(c_i|\bar{t})$表示文档不包含词条t时属于c_i的条件概率；m表示类别数。

3. χ^2统计(CHI)

χ^2统计方法度量词条 t 和文档类别 C 之间的相关程度，并假设 t 和 C 之间符合具有一阶自由度的χ^2分布词条对于某类的统计值χ^2越高，它与该类之间的相关性越大，携带的类别信息也较多[27]。令 N 表示训练语料中的文档总数，C 为某一特定类别，t 表示特定的词条，A 表示属于 c 类且包含 t 的文档频数，B 表示不属于 C 类但是包含 t 的文档频数，C 表示属于 C 类但是不包含 t 的文档频数，D 是既不属于 C 也不包含 t 的文档频数。则 t 对于 C 的 CHI 值由下式计算：

$$\chi^2(t,c)=\frac{N*(AD-CB)^2}{(A+C)(B+D)(A+B)+(c+D)} \quad (13)$$

对于多类问题，分别计算 t 对于每个类别的 CHI 值，再用下式计算词条 t 对于整个语料的 CHI 值，分别进行检验：$\chi^2_{max}(t)=\max_{i=1}^{m}\chi^2(t,c_i)$。式中，m 为类别数。从原始特征空间中移除低于特定阈值的词条，保留高于该阈值的词条作为文档表示的特征。

4. 互信息（MI）

MI 在统计语言模型中被广泛采用。文本中单词 t 的互信息 定义为 t 与所有类的互信息的平均值，

$$f_{Mu}(t)=\sum_{i}p(c_i)\lg\frac{p(t|c_i)}{p(t)} \quad (14)$$

它与信息增益的本质不同在于它没有考虑单词发生的频度，这是互信息一个很大的缺点，其原因是它删掉了很多高频的有用单词。

选择特征抽取方法的一个依据是Y．Ymg的实验。该实验指出IG是最好的测度

之一[25]。 而在[28]的实验中，OR是最好的测度，较差的是MI，最差的是IG。国内有学者认为MI>DF>IG[29]。这些差异可能源于学习算法和对数据域定义的不同。以上几种评估函数有各自的优点和缺陷，可以通过对文本分类的结果进行评价，来选择使用哪一种，或者哪几种结合使用使过滤效率和精度最高。

作者认为，由于中文与英文的文本分类问题具有相当大的差别，体现在原始特征空间的维数更大，文章向量表示更加稀疏，词性变化更加灵活等多个方面。在英文文本分类中表现良好的特征抽取方法未必适合中文文本分类。对中文文本分类中的特征抽取方法进行系统的比较研究十分必要。在下面一节，作者将进一步分析提出一种新的特征过滤函数，以适应中文文本分类特征降维的特点。

最后需要说明的是，有两种策略用于确定降维后特征空间的大小。一种直接给出过滤函数值，不论保留多少维，只要大于给出的过滤函数值的特征均被保留(称为THR)。还有一种策略是制定按过滤函数值由大到小的排列取前若干个。具体又有两种指定方式，一种称为MSV ，即指定降维后空间大小与原始空间大小的比例，另一种称为PFC,直接指定降维后空间的大小.

2.2.4 基于均值方差的特征降维方法

如何将文本表示成可以方便计算而且尽量体现文本语义的形式是文本分类任务遇到的首要问题。下面通过回顾这一领域发展呈现的脉络更好的理解算法提出的思路。

如果什么都不做，采用鸵鸟政策，即直接以文本的编码形成向量，这样的方式几乎不能工作。因为这种方式，虽然保留了文本本身所有的信息，但是却得不到任何反映文本结构、意义的信息甚至是统计特征，而只是停留在最低的信号编码层次。这样还使得计算的代价很大，文章有多长，就有相应长度的向量。并且，对不同的文本，向量长度不同，每一维的意义也不一样，不仅包含文字，也包含在文本中出现的所有的标点符号等，完全没有可比性，因而后续的机器学习是没法有意义的完成的。因此，这种鸵鸟政策是不合理的，不能采用。

因此，人们希望能够在向量的生成中，提取到文本的某些结构信息，使得向量的表示不是编码单位的简单堆叠。因此，这就需要对文本进行预处理。特别的，对于汉语文本，就是要得到文本的词序列，而不是原始的字的编码序列。这项工

作叫做分词。分词是指使用自然语言处理技术对汉语文本进行词法分析。词法分析完成后文本不再是字的连续序列，而是由指定分隔符分开的单词序列。传统语言学根据词的形态结构把语言分为三大类，分析型语言、黏着型语言和曲折型语言：1)分析型语言：词基本上没有专门表示语法意义的附加成分，形态变化很少，语法关系靠词序和虚词来表示。如汉语。2)黏着型语言：词内有专门表示语法意义的附加成分，一个附加成分表达一种语法意义，一种语法意义也基本上由一个附加成分来表达，词根或词干跟附加成分的结合不紧密。如芬兰语、日语、蒙古语等。3)曲折型语言：用词的形态变化表示语法关系，一个形态成分可以表示若干种不同的语法意义，词根或词干跟词的附加成分结合得很紧密，往往不易截然分开。如：英语、德语和法语等。英语是典型的曲折型语言，它的词法分析工作相对简单得多，因为英文单词自然以空格为形式上的分隔符，词法分析只需要处理特殊的符号串即可，提取单词的词干也可以采用预先定义规则的方法实现。具体而言，主要有三项任务：1）Tokenization：把字符串变成词串（auto segmentation）2) Lemmatization：对词的内部结构进行分析 3) POS-Tagging: 词性标注。而中文文本除了要处理特殊的符号串，还需要在连续的字串中识别出词。其中还应考虑重叠词，未定义词。分词工作是汉语处理的一项基本而重要的工作。 在本文工作中，我们暂时把文本预处理简化为对中文文本的分词。分词有多种方法。使用得最多的是统计的方法，采用 n 元语法或者隐马尔科夫模型（HMM）,另外的方法还包括形式语法理论支持下的方法，其核心类似程序设计语言词法分析的算法。

把中文文本都切分好词后，在本文的工作中可以认为是完成了文本预处理的过程。完成预先处理后，接下来要做的工作就是特征降维。正如前文所述，特征降维的作用很重要，它不仅关系计算代价进而影响分类效率，还关系到提高文本分类的效果。有不少经典的方法进行特征约简的工作。详细参见 2.2.3 节。本文工作采用下面所述的全新的特征过滤方法。

特征的原始空间是训练文本中出现的所有的词，或者是这个语言的词典中所有的词。这些词作为计算文本向量每一维的标引值的对应词，在大多数文本中数目是巨大的。简单的把所有词作为文本向量的维，将会引起巨大的计算代价并且引入相当的噪声，使得后续的机器学习难以进行，分类的效果不能忍受。在文本

分类任务中，人们仍然沿袭了文本处理的传统典型方法，比如统计词在文本中出现的频率或者出现在不同文本中文本的总数作为评价词语是否应该取舍的度量标准，这些方法没有跳出原有框架的束缚，没有考虑针对文本分类这个应用特有的性质，因而方法的通用性保留了，却缺少在文本分类这样的特定应用下的有效性。保留怎样单词作为文本向量的维才能够尽量的体现这个词对于标志某篇文本属于某个类别的特征呢？注意这里要求的不再是单词特征对于文本的表义能力，而是对于一篇文章属于某个类而不属于其他类的归属－区别能力。换言之，就是一个词语对于标识一类文本作为一个类区别于另一类文本的共同特征，关注的是属于一个类别的文本的集合的共同的区别于其他类别文本的特征。关注于一个类别的文本的“聚合”属性，和他们作为整体相对于其他类别的“区分属性”。然而，在传统的文本处理中，保留单词的着眼点在于使被保留的特征尽量多的反映文本本身的统计特征。比如，在一篇讲述文学史的文章中，一些很有特色的词，比如“先秦散文”可能作为表示这篇文本的关键词出现，因此在计算这篇文本的向量时，“先秦”“散文”是不能被去掉的重要特征之一，因为它很好的表达和体现了这篇文本的内容特征，是这篇文本区别于其他文本的关键。但是在文本分类的应用背景中这样的分析却不一定成立，这样的关键词也不一定是必须保留的重要特征。因为这个词虽然很好的表达了这篇文本的特点，却可能是这篇文本作为属于某个类别的文本的角色时的例外。在计算单个文本的向量时，可能很需要捕捉这样的例外，它正是使这篇文本内容特征独到的地方。然而在计算一个作为一类文本中的一篇文本的向量表示时，我们需要捕捉的是这一类文本共有的集中的区别于其他类的文本特征，而不是某篇文本区别于其他所有文本的特征。比如，在上面的例子中，如果例举的文本属于文学类别，那么我们在计算其向量表示时，更倾向于保留“散文”这个特征，而不是“先秦”，因为可能“先秦”削弱了文本作为文学类别的特征。当然，如果这篇文本属于“历史”类别，则倾向于保留后者。同理，如果认为“先秦”这个特征的出现在历史类别的文本中出现比较集中，“散文”在文学类别中出现较多时，则这篇文本就可能同时属于这两类。这并非不合理的现象。因为一篇文本可能本身涉及的就是两个或两个以上领域的交叉部分的内容，而且对于比较长的文本，也可能同时涉及两个或两个以上部分的内容。这也就是说我们不希望保留这样的特征，即它虽然很好的体现了单个文本

的特点，而这样的特点却不能代表这一类文本的区别于其他文本的共同的特点，而只是一个“特立独行”，不是“代表本类的集体特征”。但是这样的特征，在传统的处理单个文本的向量计算方法中，被淘汰的可能性不大。那么怎样才能充分使用文本分类这个任务的特点更好的进行文本向量的计算呢？

本文的工作提出了一种全新的算法，称为：基于均值－方差的特征过滤算法。这个算法的基本思想就是把考察一个词对于体现一类文本区别于其他类的共同特征的能力作为判断是否应该保留相应特征的出发点。具体而言，就是寻找词的一种有效度量，考察这种度量形式在区分文本类别时的能力，以此度量的函数作为筛选特征的度量标准。

首先假设训练语料库中的文本有 m 类，每一类有 $n_i(1<=i<=m)$篇文本。为方便直观理解算法，我们把它们排列为以下形式：

$$\begin{bmatrix} d_{11} & d_{12} & \dots & d_{n_1} \\ d_{21} & d_{22} & \dots & d_{n_2} \\ \dots & d_{ij} & \dots. & \dots \\ d_{m1} & d_{m2} & \dots & d_{mn_m} \end{bmatrix} \tag{15}$$

其中 d_{ij} 表示这样的文本，它的类别属于第 i 类，j 为文本在第 i 类中的序号，第 i 类文本共有 a_i 篇文本。为方便起鉴，不妨假设针对词典中的每一个词都有这样的一个阵列，于是，d_{ij} 为一个词典中的词在文本 d_{ij} 中的出现的某种度量。比如，可以是词频，归一化的词频等等。实际上，很多在这个领域得到的结果都可以用在这里作为单词在文本中出现的度量。在本文的工作中，使用归一化的词频率 $\text{occurrence}_w(d_{ij})$ 作为这种度量，并简单的用 d_{ij} 表示。可以认为下面的设想是合理的，即词典中的一个单词能不能成为一个好的特征在于这个词对同类的所有文章表现出了很好的统计上的某种一致性，而对于不同的类别的文章这种统计性质差别比较大。怎样量化这种评价指标体现这种类中的一致性和类间的差别呢？这里，本文的工作是用词语在文章中出现的概率作为词语在文章中的统计特性的指标。词语统计特性在同类文本中的相似性大小用这种统计特性在同类文章中的均值进行度量，词语统计特性在不同类别间的差异用这种统计特性在同类中均值在不同类别中的方差进行度量。理想的状态是对于所有的已出现和可能出现的文本

进行统计，实际情况只能是对于已知的文本进行的统计，也就是针对语料库中的文本进行计算。另一方面，单词在文本中出现的概率用单词在文本中出现的归一化频率近似。这样我们得到这样的计算任务。

1） 针对词典中的每个单词计算它在所有文本中出现的概率。进一步转化为计算单词在文本中出现的归一化频率。

2） 针对每个单词计算出如上所述的概率阵列。

3） 对每个这样的阵列，计算每一行的均值，形成一个列向量。

4） 对这个列向量计算其方差。

5） 每个单词都有对应的这样的方差，形成一个队列，对其从大到小进行排序。

6） 按照一定的规则取这个序列中前若干单词作为应该保留的特征。

其中的规则有两种得到的方式：一种是给这个判定过滤函数的阈值，一种是取给定绝对数目的若干单词，一种是给出应该保留的单词数目所占原始特征空间的比例。作为对于本文工作中提出的特征约简方法的检验和对分类效果与分类方法、降维方法、维数的关系的研究，分别得到了若干维上的不同方法分类的效果，具体描述分析在第三章。

2.2.5 权值计算，向量生成

得到约简后的特征空间，接下来要做的就是按这个特征空间中的项，也就是保留的词语，确定一个在其上的度量，计算每篇文本中在这些词上这个度量的值。这也称作索引（I*ndexing*）。这不是本文工作关注的重点，实际上可以采用不同度量方式，比如，特征在文本中的出现频率 TF。索引（Indexing 权值计算）是确定使用什么样的语言单位(多大的语言单位)及其度量作为向量表示的维，进一步计算出文本的每个语言单位在对应维上的取值。首先的一个想法是采用词在文本中出现的次数作为选定词特征的度量。但是这种方式没有考虑文本的长度，不能使同样特征的权重在不同长短的文本中具有可比性。因此，用文本长度的指标对其进行归一化，也就是采用归一化词频作为词的特征度量。这也是经常采用的一种方式。采用词频作为词特征的度量时，认为词频越高，这个词描述相应文本的能力强的可能性越大。有时，人们还进一步以词频的某些函数来作为词的特征，

这是因为在考虑词的相对重要性时，词 a 的词频是词 b 词频的四倍，不等于其重要性也是四倍强的关系。除此之外还有文本倒频率等计算方法。

经典的方法是采用出现频率和文档倒频率因子的乘积。文档倒频率因子是指一个特征词在文档中出现的总次数的倒数。可以设想，一个词语的文档倒频率因子小，说明它在多篇文档中都出现，因此在这一维上文档的出现频率来表示文档在这一维上的重要性时会受到削弱的影响，因此在词频上加权文档倒频率来表达这种重要性的差别。

2.2.6 小结

综上所述，文本分类第一阶段是把文本表示成向量。其中有三个主要的步骤。第一是利用自然语言处理的方法对中文文本进行分词，第二是用各种方法对特征空间进行降维。对训练语料库的文本进行分词的结果形成的原始的特征空间进行约简，以此作为计算文本向量对应的维。第三个阶段就是对分好词的文本按降维后的特征空间计算向量每一维上的值，从而形成向量。

2.3 文本分类第二阶段——学习机器的训练

2.3.1 机器学习概述

本节将简略的从人工智能展开，逐步讨论机器学习，统计机器学习，以导出本文工作的文本分类所采用的一种学习方法——支持向量机。

2.3.1.1 人工智能背景

（一）人工智能的研究思路可以概括为以下四个方面[30]：

1）图灵测试 是阿兰.图灵于 1950 年提出的，设计的目的是为智能提出一个满足可操作要求的定义。基本思想是使通过人这个智能实体的辨识能力而不是例举智能所需要的能力来对智能体的智能进行评价。即，若人作为询问者在提出一些书面问题后，无法判断答案是否由人写出，那么计算机就通过了测试。要通过这个测试，计算机需要以下若干方面的能力：自然语言处理的能力、知识表示、自动推理、机器学习、计算机视觉和机器人技术。这六个领域也就构成了人工智能的六大部分内容。图灵测试也是在“像人一样行动的系统”这个思路下的方法。

2）认知模型方法 这种方法认为必须深入人类思维的真实过程和本质。这可以通过人的内省和心理测试来达到。只要我们获得了关于思维的足够精确的理论，那么就可能通过计算机程序来表达。而认知科学领域就是把来自人工智能的计算机模型和来自心理学的试验技术相结合，试图创立一种精确而可检验的人类思维工作方式的理论。（生理心理学）认知模型的方法是从类似人进行思考的这个角度进行的研究。

3）思维规则的方法——强调正确的推理

思维规则的方法背后隐含的一个思想是源自古希腊哲学家亚里士多德试图严格定义的“正确思考”，并认为是这些可以被描述为“思维规则”的“正确思考”支配人的意识活动。这也就是后来创立的逻辑学研究领域。19 世纪，逻辑学家发展出一种描述世界上一切事物彼此间关系的精确的命题符号。原则上，可以用程序求解任何逻辑符号描述的问题（半可计算的完成）。但是希望用编制程序来创建智能系统的“逻辑主义”方法有两个难点：1）如何获得非形式化的知识并将之用逻辑符号表达为形式化的形式。特别在当这样的知识并非总为真的情况下。（知识表示）2）在推理过程中如何使得推理能够在可以接受的时间空间里完成。因为原则上半可计算，实际上还要面临计算的复杂度问题。（推理）

4）Agent 的方法

这要求不仅具有表示知识和推理的能力，还要具有自然语言生成的能力、学习的能力和感知的能力。学习是因为对世界更好的理解有助于找到解决问题的更好的有效的策略，感知是为了对行为可能的后果有更好的了解。

（二）人工智能研究的根基涉及到很多学科方面，包括哲学，数学，逻辑学，语言学，经济学，神经科学，心理学，控制论，计算机工程等等[30]。

形式逻辑可以追溯到古希腊的哲学家（如，亚里士多德）。实质性的开端是从乔治.布尔（George Boole）的工作开始的，经 Gottlob Frege 扩展，成为了当今知识表示系统一阶逻辑。19 世纪晚期，把一般数学推理形式化为逻辑演绎的努力已经展开。1931 年哥德尔（Gödel）证明不完备性定理，表明在任何表达能力足以描述自然数的语言中，在不能通过任何算法建立他们的真值的意义上，不存在可判定的真值语句。这进一步引起了图灵的进一步研究，提出了丘奇-图灵论题，说明图灵机有能力计算任何可计算函数。从而区分出了不可计算类。第二

个问题是在可计算的函数类中，什么样的函数的计算所消耗的时间是可以接受的，这称为可操作性。Steven Cook 和 Richard Carp 进行了 NP 完全问题的研究指出大量各种类别的规范的组合搜索和推理都属于 NP 完全问题，任何 NP 完全问题类可规约而成的问题类很可能是不可操作的。

意大利人 Ceroamo Cardano 首先搭建了概率思想的框架。概率很快成为所有需要定量的科学的无价宝，帮助对付那些不确定的测量和不完备的理论。Thomas Bayes 提出了根据新证据更新概率的法则。贝叶斯法则及其衍生的“贝叶斯分析”的领域形成了大多数 AI 系统中不确定推理的现代方法的基础。

经济学的作用来自于经济学实际真正研究的是人们如何选择以达到最理想的效果。这方面很有用的结果是 John Von Neumann 和 Oskar Morgenstern 的著作《博弈论与经济学行为》。决策理论把概率论和有效理论结合起来，为在不确定条件下进行决策提供了形式化和完整的框架。而博弈论指出参与者的行动会显著影响其他参与者的效用。

然而试图用一阶逻辑处理类似医疗诊断这样的领域会表现得无力。原因主要在于以下三个方面：1）惰性：为了确保得到一个没有任何意外的规则，列出了所有需要的前提和结论的完整集合的工作量大得几乎不能完成。并且即使列出也太繁杂得无法使用。2）理论的无知：对于规则描述的理论，这样的规则是否正确还缺少多少结论成立的前提条件还不能完全确定。3）实践的无知：对于适应规则的前提是否成立的事实的调查检验常常是不可能的或者是非常困难的。这可以用如下的图式表示：

A,B,C,D,E,……=>a,b,c,d,e,f,g…… （16）

难以确定一条规则要绝对成立，需要的所有前提条件，尤其是在对规则涉及的领域没有了解彻底的情况下（而这又是通常的情况），而且也很难一一检查确定现在的事实是不是满足规则的前提条件。

这对应于文本分类任务就是采用传统的人工制定规则的基于知识库的方法。显然这对于有效的分类任务是不足的。因此，本文工作采用的是学习的方法。下面就简要介绍机器学习和统计机器学习，作为下一节支持向量机的基础。

2.3.1.2 机器学习的一般概念

如果一个计算机针对某类任务 T 的用 P 衡量的性能根据经验 E 来自我完善，那么我们称这个计算机程序在从经验 E 中学习，针对某类任务 T，它的性能用 P 来衡量[30]。

实际的机器学习问题往往比较复杂，需要定义一类问题、探索解决这类问题的方法、理解学习问题的基本结构和过程。学习系统需要选择要学习的知识的确切类型、对于这个目标知识的表示以及一种学习机制。可以认为学习智能体包含决定采取什么动作的执行元件和修改执行元件使其能制定更好的决策的学习元件。一个学习元件的设计受到下列三个因素的影响：将要学习的是执行元件的哪个组成部分；对学习这些组成部分而言，可以得到什么反馈；最终部分是如何表示的。

（一）下面简要介绍学习理论发展的两个重要阶段[32]。

1．60s Rosenblatt 的感知机

感知机的思想并不是新的，它已经在神经生理学领域被讨论多年。Rosenblatt 把这个模型实现为一个计算机程序，并且通过简单的实验说明这个模型能够被推广。感知器模型被用来解决模式识别问题，在最简单的情况下就是用给定的例子来构造一个把两类数据分开的规则。

感知机利用了最简单的神经元模型的自适应特性来构造分类规则[33]。每个神经元是一个 McCulloch-Pitts 模型，有 n 个输入 $\vec{x}=(x^1,...,x^n)\in X\subset R^n$，和一个输出 $y\in\{-1,1\}$，满足这样的函数关系 $y=\mathrm{sgn}(\vec{w}\bullet\vec{x}-b)$。其中 $\vec{w}\bullet\vec{x}$ 为两个向量的内积，b 是一个阈值，sgn () 是符号函数 $\mathrm{sgn}(u)=1 if u>0 else=0$。即从几何上看，神经元定义了输入空间中取值为 0 和 1 的两个区域，他们被超平面 $(\vec{w}\bullet\vec{x})-b=0$ 分开。通过学习选择适当的参数 $\vec{w}$ 和 b, 从而选择确定了分类函数。

进一步地，若采用神经元的多层结构，即前一层神经元的输出是下一个神经元的输入，最后一层只有一个神经元，这样感知机用分段线性的面把空间 X 分为两部分。学习就是用给定的训练数据寻找所有神经元的适当的系数。确定系数的方法有径向基函数法（BP 技术----神经网络原理，动态规划的算法设计思想）

当时 Rosenblatt 采用的方法是，固定除了最后一个神经元以外的其它所有神经元的系数，在学习过程中寻找最后一个神经元的系数。从几何上看，就是把输入空间变换到一个新的 Z 空间，用训练数据在 Z 空间构造分类超平面。

1962 年，Novikoff 证明了关于感知机的第一个定理，成为学习理论的开始。定理指出，若

1) 训练向量 z 的模以某个常数 R 为界 即 $|z| \le R$；

2) 训练数据能够一间隔 ρ 被分开 即 $\sup \min_i y_i(\vec{z_i} \bullet \vec{w}) > \rho$

3） 可以对感知机进行足够多的训练

则，最多在 $N \le \left\lceil \frac{R^2}{\rho^2} \right\rceil$ 次的修正后构造出将这些数据分开的超平面。

进一步可以证明，如果数据是可分的，那么在有限次修正后，感知机能够将任意无限长的数据序列分开。即，在最后一次修正后，剩下的无穷多数据能够被正确的分开。

当感知机采用下面的训练停止规则时，即，若在第 k 次修正后，接下来的 m_k（ $m_k = \frac{1+2\ln k - \ln \eta}{-\ln(1-\varepsilon)}$ ）个样本都没有使决策规则（分类面的参数）改变，则停止感知机的学习，下列结论成立：感知机会在前 l 步学习中停止学习过程。其中 $l \le \frac{1+4\ln\frac{R}{\rho} - \ln\eta}{-\ln(1-\varepsilon)} \left\lceil \frac{R^2}{\rho^2} \right\rceil$。在学习停止前，感知机已经建立了一个决策规则，它在测试集上的错误率小于 ε 的概率是 $1-\eta$。或者称，它以概率 $1-\eta$ 具有在测试集上小于 ε 的错误率（Aizerman Braverman and Rozonoer 1964）

自此，学习分为两种看法（学派）[31]：

认为使学习机器具有推广性（泛化性能，小的测试错误率）的唯一因素是使它在训练集上的误差最小。即，只要选择使训练错误数最小的神经元系数就足够了。（应用学派）

最小化训练错误数并不是不言而喻的，而是需要证明的。实际上存在一个更为智能的归纳原则，并构造算法实现这一原则，它能够提供更好的推广性能。（理

论分析学派的贡献）

2．学习理论的创立（60s-70s）(理论分析学派的贡献)

对于指示函数集（既模式识别问题），提出了 VC 熵和 VC 维的概念，它们是这一新理论中的核心概念。利用这些概念，发现了泛函空间的大数定律（频率一致收敛于概率的充分必要条件），研究了它与学习过程的联系，并且得到了关于收敛速度的非渐进界的主要结论[34]，并在 1971 年发表了这些工作的全部证明。所得到的这些界使得一个全新的归纳原则——1974 年提出的结构风险最小化归纳原则——成为可能，从而完成了模式识别学习理论[35]。 在 1976 到 1981 年间，最初针对指示函数集得到的这些结论推广到了实函数集，主要内容有： 大数定律（均值一致收敛于其期望的充分必要条件）完全有界函数集和无界函数集一致收敛速率的界，结构风险最小化原则。1984 年，Valiant 提出机器学习应该以模型概率近似正确(1-δ)为指标，而不是以概率为 1 为指标。1989 年发现了经验最小化归纳原则和最大似然方法的一致性的充分必要条件，最后完成了经验风险最小化归纳推理的理论分析。90 年代，开始了对于能够控制推广性能的新的学习机器的合成。这些都是统计机器学习的理论基础。

另外在 60s 提出了统计学和信息论中最伟大的思想之一——算法复杂度思想（solomonoff 1960; Kolmogorov 1965 Chaitin 1966）。 随机性概念的思想可以粗略的描述如下：对于一个长度为 l 的很长的数据串，如果不存在任何复杂度远小于 l 的算法能够产生出这个数据串，则它就构成了一个随即串。算法的复杂度是用实现这个算法的最小程序程度来衡量的（它是确定的，除了反应计算机类型的加性常数外）。而且已经证明，如果对串的描述不能被计算机压缩，则这个串具有一个随即序列的一切性质。这也就说明，如果我们可以在很大程度上压缩对一个给定串的描述，那么所使用的算法就描述了数据的内在性质。 在这些思想的基础上，提出了对于学习问题的最小描述长度归纳推理[36]。这就是机器学习热点之一符号机器学习的理论基础。

实际上符号机器学习和统计机器学习是机器学习研究的热点领域。

（二）有三种典型的机器学习方式[37]

1．增强机器学习

它更加关注随时间变化数据的分析与建模，使用 Markov 与其他非 Markov 方法如 Game Theory。

2．符号机器学习

泛化目标变为数据描述(符号数据分析)。一般规模的数据集合变为具有上百属性超过十万对象的数据集合。

3．统计机器学习

从神经网络的研究转变为统计机器学习与集成机器学习。强调学习算法不应只解决玩具世界问题，已而从非线性算法转变为以线性算法为主。此外，必须考虑学习算法泛化能力，即是对测试数据集的错误率要足够低。统计机器学习的理论基础是前述的机器学习发展的两个重要阶段之第二阶段的理论流派的观点，即对训练数据集合的错误率小不一定导致良好的泛化能力。泛化能力的研究主要有：1) 以样本个数趋近无穷大来描述模型的泛化能力。泛化能力需要使用世界 W 来刻画(而不是有限观测的样本集)。(Duda); 2)从“有限样本”建立模型，以估计其对世界为真的程度（Vapnik)。进而发展成为 Vapnik 的有限样本统计理论。这个统计理论是从“有限样本”建立模型，以估计其对世界为真的程度。其关键在于将泛化误差考虑为一个依赖从问题世界随机选择样本集的随机变量。样本集和模型与误差之间的关系，即，需要确定模型泛化误差的界。研究泛化误差界的目的，除了估计泛化能力之外，主要为了指导算法设计。

一般地说，有限样本归纳的模型与问题世界的统计分布有误差，即，存在损失，或称存在风险。人们希望归纳的模型对问题世界有最小风险。如何描述风险，如何从样本集估计这个风险，这就是泛化的统计问题。这是概率统计理论中的经典问题。可以分为两类：基于先验知识的泛化理论：这需要先验地了解问题世界的部分统计性质。基于数据的泛化估计:(1)以大数定理为基础(无限样本)——与 duda 的观点对应[37, 38] (2)以函数的划分能力为基础(有限样本)——与 Vapnik 的观点对应。不同于以无穷样本为基础的统计理论，这个理论主要考虑在划分样本空间的函数下，风险的下界。Vapnik 在 1971 年奠定了有限样本的统计理论。这经历了三个阶段：PAC 误差，VC 维与最大边缘。最大边缘是 SVM 算法设计的基础。

2.3.1.3 统计机器学习

（一）概述

给定样本集合$(x_1, y_1), \ldots, (x_n, y_n)$，其中，$x_i \in R^d$。学习问题是从函数集$\{f(x, \alpha)|\Lambda \in \alpha\}$中，选择一个最优函数 $f(x, \alpha^*)$，使得它是对 x 与 y 的最好的估计。$f(x, \alpha)$也可以称为一类基函数。对两类问题：$Y=\{0,1\}$，此时称函数集合$\{f(x, \alpha)|\Lambda \in \alpha\}$为指示函数集合。因为本文涉及的工作是有监督两类问题机器学习(分类问题，或模式识别问题)，而不是更一般的回归问题，这样，讨论将限制在指示函数集合的划分问题上。指示函数集合 $f(x, \alpha), \Lambda \in \alpha$ 的 VC 维，是能够被集合中的函数以所有可能的 2^d 种方式分为两类的向量 x1, ...,xd 的最大数目。对一个给定的指示函数集，它的 VC 维说明了其划分样本的能力(这个函数集的复杂程度)。它与样本如何标定其类别无关。对线性指示函数集合，它能够打碎的样本数为样本维数加 1，这是线性指示函数集合的 VC 维。频率(经验风险)到概率(期望风险)一致收敛的条件是 VC 维有限。Vapnik 首先发现了对任何概率测度(与分布无关)的 ERM 原理一致的充要条件：如果指示函数集合 $Q(z, \alpha)$的 VC 维是有限的，经验风险将快速渐进收敛于期望风险。其中 ERM 原理一致是指经验风险与期望风险都收敛到最小的风险值。假设指示函数集合满足一致性条件。期望风险与经验风险满足下述不等式：$R(\alpha) \leq Remp(\alpha)+\Phi(k/d)$，其中，$Remp(\alpha)$是经验风险，$\Phi(k/d)$称为置信范围。d 是指示函数集合的 VC 维。为了获得最小的 $R(\alpha)$(期望风险)，必须使得 $Remp(\alpha)$(经验风险)与 $\Phi(n/d)$(置信范围)同时最小。这就是结构风险最小化原理。

1984 年 Valiant 提出一类机器学习理论 PAC。不求学习结果精确，不求学习一定成功，学习结果除了有小的误差或失败可能，可以基本成功。但是，学习必须在多项式时间完成。称为“概率、近似正确地可学习”，PAC(Probably Approximately Correct)。令 F 是一个待学习的概念类，D 是一个确定但未知的样本集合 S 的分布。对所有的分布 DS，所有概念 $f \in F$，以及所有 $0<\varepsilon,\delta \leq 1$，算法 Ω 输出一个假设 h，使得误差率 $D(h(S) \neq f(S)) \leq \varepsilon$ 至少以概率 $1-\delta$ 成立。将泛化误差考虑为一个依赖随机选择样本集 S 的随机变量，因为 S 与假设(模型)空间 H 对应，因此，这是一个依赖假设空间 H 的随机变量。这样，可以通过样本集合上的随机变量研究泛化误差界。根据分布 D，随机选择的任一个样本集 S，”假设”hS 的

泛化误差界为 errD(hS) ≤ ε(k, H, δ)，以概率 1-δ 成立。其中 k 是 S 中的样本个数。

为了给出便于指导算法设计的泛化界，1998 年，Shawe-Taylor 使用类似的方法给出了最大边缘的泛化界。

$$err(h) \le \sqrt{\frac{c}{k}\left(\frac{R^2}{\gamma^2}\log^2 k - \log\delta\right)} \qquad (17)$$

这个不等式依赖于边缘 γ 。它给出了有几何直观的界描述，从而为算法设计奠定基础。

（二）学习问题的形式化描述[32]

我们可以认为学习问题是利用有限数量的观测来寻求依赖关系的问题。根据样本进行学习的一个模型如图 4 所示。

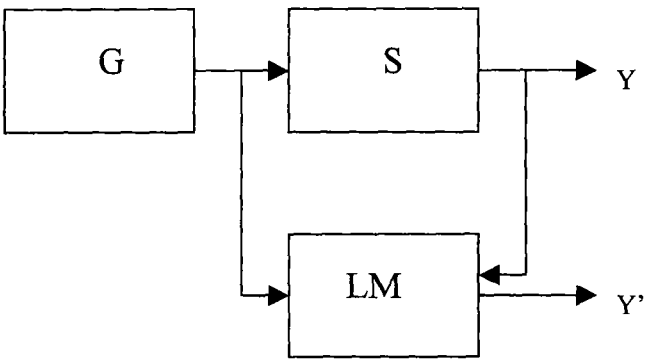


图 4 机器学习的模型图

产生器 G 产生随机向量 $\vec{x} \in R^n$，它们是抽样的来源，独立同分布，服从相同的概率分布函数 F（x）(未知)；

训练器 S 对输入的向量 $\vec{x}$ 根据固定但未知的条件分布函数 F（y| x）产生输出值 y。这个输出值就是有监督学习中教师的指导，比如对于某个样本点，指定其对应的数值。这个数值可以是连续的也可以是离散的，比如可以是 0 或 1 ，以对应实际世界的真假值。

学习器 LM 它实现一定的函数，相应于前述的假设 h，函数属于函数集{f(x, α) |$\alpha \in A$}，相应于前述的假设空间。

学习的问题是从假设空间函数集合中选择最好逼近训练器响应的函数。这种

选择是基于训练集的。训练集由下列形式的独立同分布的样本组成（x,y），它们服从联合分布 F(x,y)=F(x) F(y|x)。当然这些学习器学习的样本满足独立同分布的条件。（i.i.d Independently Identically Disrtibuted）

学习的目标就是使得学习到的函数不仅能很好的逼近描述看到的样本，也能对未知的，学习器尚未看到的样本进行拟合，同时 1）防止对训练数据的过度拟合，2）寻找误差和函数复杂程度的某种折中(这也是基于前述的所估计的函数是不确定的这个观点，因为任何观测不能被绝对精确的描述)。这个目标形式化成下面的要求：即最小化 风险泛函 $R(\alpha)$ 。其中 $\lim_{l\to 0} R(\alpha_l)(p)$ 。即，在联合概率分布函数 $F(x,y)$ 未知，所有可用信息为独立同分布的样本时，寻找函数 $f(x,\alpha_0)\in\{f(x,\alpha)|\ \alpha\in\Delta\}$,能够最小化风险泛函 $R(\alpha)=\int L(y,f(x,\alpha))dF(x,y)$ 。更一般化的描述是最小化风险泛函 $R(\alpha)=\int Q(z,\alpha)dF(z),\alpha\in A$ 。这里的 $Q(z,\alpha)$ 就是损失函数，特定的损失函数具体化了学习问题的类别，可以是模式识别，回归估计，或密度估计等等问题中特定的损失函数形式。本文工作也属于这个框架，特别的，它属于这个框架下的模式识别问题。

令训练器的输出 y 只能取两种值 y={0,1} ,在这种情况下，函数类 $\{f(x,\alpha),\ \alpha\in A\}$ 为指示函数集。则损失函数为

$$L(y,f(x,\alpha))=\begin{cases}0 & \text{若}y=f(x,y)\\ 1 & \text{若}y\neq f(x,y)\end{cases} \qquad (18)$$

在未知样本所满足的分布函数情况下（事实上，知道了样本的分布函数，就失去了学习的意义了），不能通过上式实现最小化风险泛函的目的。这时要转而使用其它的近似途径和原则。其中一个原则就是经验风险最下化原则，相当于前述的使估计的函数尽量满足所有的训练样本点，但是这样做的缺点是明显的，这将在后续分析中看到，因为完美满足样本不一定能完美满足新的测试样本。 这里是作为推导的铺垫，描述经验风险最小化原则 ERM。

（三）经验风险最小化原则

用训练集合 $\vec{z}_1=(\vec{x}_1,y_1),...,\vec{z}_l=(\vec{x}_l,y_l)$ 得到如下经验风险 $R_{emp}(\alpha)=\frac{1}{l}\sum_{i=1}^{l}Q(\vec{z}_i,\alpha)$ 。其中 $Q(\vec{z}_i,\alpha)=L(y_i,f(\vec{x}_i,\alpha))$ 为损失函数， $f(\vec{x}_i,\alpha)$ 为学

习器要学习到的函数。同时，对于训练集合假定其独立同分布，都来自分布函数 $F(\vec{z})$。于是有风险泛函 $R(\alpha)=\int Q(\vec{z},\alpha)dF(z)$。用使经验风险最小的函数 $f(\vec{x},\alpha_{\mathrm{A}})$ 逼近使风险泛函最小的函数作为学习器的输出函数。称这样的原则为经验风险最小化原则。相应就定义了一个学习过程。这个学习过程就是要在未知实例服从的分布函数 $F(\vec{z})$ 下，假设训练样本独立同分布，由这些训练样本，解下面的最优化问题。求函数 $f(\vec{x},\alpha_0)$，$\alpha_0 \in \mathrm{A}$

$$\min \mathrm{R}_{emp}(\alpha)=\frac{1}{l}\sum_{i=1}^{l}Q(\vec{z_i},\alpha) \tag{19}$$

其中 $Q(\vec{z_i},\alpha)=L(\vec{y_i},f(\vec{x_i},\alpha))$ 为损失函数，度量在给定输入 $\vec{x}$ 下，训练器响应 y 和学习器给出的响应 $f(\vec{x},\alpha)$ 间的损失或者说差异。

经验风险泛函 $\mathrm{R}_{emp}(\alpha)=\frac{1}{l}\sum_{i=1}^{l}Q(z_i,\alpha)$，在模式识别学习问题中就是 $\mathrm{R}_{emp}(\alpha)=\frac{1}{l}\sum_{i=1}^{l}L(y,f(x,\alpha))$，在回归问题中实际上就是最小二乘法，在密度估计中就是最大似然估计。在本文工作的环境中，$\vec{x}$ 为输入的表示训练语料库中的文本的向量，y 为其文本所属的类别。特别的，对于某一类，y=1 表示文本属于这一个类别，y=0 表示文本不属于这样的类别。至于这样的指定如何与本文工作的多文本分类问题联系起来，将在 2.3.4 多分类问题一节中陈述。下面进一步分析什么情况下最小化经验风险导致取得最小的实际风险，即获得最好的推广能力。

当使用经验风险最小化原则建立学习过程时，需要确定的一个问题是以经验风险最小化作为目标得到的函数是不是一定能使实际风险能够最小。直观的，这个结论不总是成立。因为用经验风险来代替期望风险，相当于只是用在训练样本集合上的平均误差代替在所有可能样本上的期望误差。但是，可以研究在训练样本足够大，趋于无穷大的过程中的渐进特性——ERM 原则的一致性。在得到无穷大时的情况后可以返回来得到有限样本下的情况，即估计经验风险最小化得到的函数的好坏程度。

称 ERM 原则是一致性的，更准确的称为 ERM 原则对于函数集合当且仅当下

面两个序列依概率收敛于同一个极限 即

1） $\lim_{l\to\infty} R(\alpha_l) = \inf_{\alpha\in A} R(\alpha) \quad (p)$

2） $\lim_{l\to\infty} R_{emp}(\alpha_l) = \inf_{\alpha\in A} R(\alpha) \quad (p)$ （20）

则称经验风险最小化原则对假设空间函数集合和概率分布函数 F（z）是一致的。这保证了可以在经验风险的取值基础上估计可能的最小风险。下面给出满足这种一致性所需的条件。

设对应与假设函数集合 $f(x, \alpha_0) \in \{f(x,\alpha) | \alpha \in \Delta\}$ 的损失函数集合 $\{Q(z,\alpha) | \alpha \in A\}$ 满足 $A \le \int Q(z,\alpha) dF(z) \le B$ 则最小化经验风险一致性的充分必要条件是：经验风险 $R_{emp}(\alpha_l)$ 满足 $\lim_{l\to\infty} P\{\sup_{\alpha\in A}(R(\alpha) - R_{emp}(\alpha)) > \varepsilon\} = 0, \forall \varepsilon > 0$

这也是统计机器学习理论指出的学习过程的一致性问题。即，基于经验风险最小化的归纳原则。在经验风险最小的学习过程中，得到取得最小的实际风险（表现为推广能力）的条件。即经验风险最小化学习过程一致性的充分必要条件。

综上所述，ERM 原则一致的充要条件是：经验风险 $R_{emp}(\alpha)$ 在函数集 $\{Q(\vec{z},\alpha) | \alpha \in A\}$ 上在如下意义上收敛于实际风险（期望风险）：

$$\lim_{l\to\infty} P\{\sup_{\alpha\in A}(R(\alpha) - R_{emp}(\alpha)) > \varepsilon\} = 0, \forall \varepsilon > 0 \quad (21)$$

称这样的一致收敛为一致单边收敛（学习理论的关键定理）。亦即，ERM 原则一致等价于 $\lim_{l\to\infty} P\{\sup_{\alpha\in A}(R(\alpha) - R_{emp}(\alpha)) > \varepsilon\} = 0, \forall \varepsilon > 0$ 一致单边收敛。

定义随机变量序列 $\xi^l = \sup | \int Q(\vec{z},\alpha) dF(\vec{z}) - \frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha) |, l = 1,2,...$ 为一个双边经验过程。这一随机变量序列依赖于概率测度 $F(\vec{z})$，函数集 $Q(\vec{z_l},\alpha), \alpha \in A$。当函数集合 $Q(\vec{z_l},\alpha), \alpha \in A$ 中仅有一个元素时，随机变量序列 $\xi^l = \sup | \int Q(\vec{z},\alpha) dF(\vec{z}) - \frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha) |, l = 1,2,...$ 总是依概率收敛于 0。这也就是统计学的基本定律，大数定律——随着观测数量 l 的增加，随机变量

$\xi^l = \sup|\int Q(\vec{z},\alpha)dF(\vec{z}) - \frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha)|$，收敛于 0。当函数集 $Q(\vec{z_l},\alpha),\alpha \in \mathrm{A}$ 包含有限个元素时，那么随机变量序列 $\xi^l = \sup|\int Q(\vec{z},\alpha)dF(\vec{z}) - \frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha)|$，依概率收敛于 0。这也就是 N 维空间大数定律（函数集中每个函数对应于一维坐标）。N 维空间大数定律可以进一步推广到泛函空间大数定律。即，泛函空间大数定律将给出在什么条件下（函数集合 $Q(\vec{z_l},\alpha),\alpha \in \mathrm{A}$ 和概率分布 $F(\vec{z})$ 满足什么特性下）对包含无穷多元素的函数集合 $Q(\vec{z_l},\alpha),\alpha \in \mathrm{A}$ 中的函数，存在均值到数学期望的一致双边收敛。

对完全有界函数 $Q(\vec{z_l},\alpha),\alpha \in \mathrm{A}$，经验均值 $\frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha)$ 一致双边收敛于其期望 $\int Q(\vec{z},\alpha)dF(\vec{z})$，即

$$\lim_{l\to\infty} P\{\sup_{\alpha\in\mathrm{A}}|\int Q(\vec{z},\alpha)dF(\vec{z}) - \frac{1}{l}\sum_{i=1}^{l} Q(\vec{z_i},\alpha)| > \varepsilon\} = 0 \tag{22}$$

的充分必要条件是等式

$$\lim_{l\to\infty} \frac{H^{\mathrm{A}}(\varepsilon,l)}{l} = 0 \tag{23}$$

成立。特别的，对于在本文工作的模式识别的问题背景里，对指示函数集，其 VC 熵 $H^{\mathrm{A}}(\varepsilon,l)$ 与 ε 无关。即 $H^{\mathrm{A}}(\varepsilon,l) = H^{\mathrm{A}}(l), 0 < \varepsilon < 1$。

因为一致双边收敛是一致单边收敛的充分条件（即，一致双边收敛条件强于一致单边收敛），而一致单边收敛是 ERM 原则（即经验最小化原则）一致的充分必要条件，所以，一致双边收敛是 ERM 原则的充分条件。

上面叙述的是双边收敛在函数集合上逐步拓展形成的函数均值依概率双边收敛于函数期望的条件。而经验风险最小化原则的渐进特性——经验风险最小化原则的一致性——的充分必要条件是单边收敛，这样的单边收敛成立的条件弱于双边收敛。

设 $\{f(z,\alpha)|\alpha \in \mathrm{A}\}$ 是指示函数集合，考虑在样本 $z_1, z_2, \ldots z_l$ 上定义一个量 $N^{\mathrm{A}}(z_1, z_2, \ldots, z_l)$，它代表用指示函数集合可以把最多多少个样本分成任意的类（打

碎）。用这个量来表征函数集合在给定数据集合上的多样性。注意，只要存在 $N^{\mathrm{A}}(z_1,z_2,...,z_l)$ 数量样本的一种分布满足条件就可以了，使得可以总是存在一个指示函数在指示函数集中能够拟合这个分布下数据的任意一种划分。进一步定义随机熵 $H^{\mathrm{A}}(z_1,z_2,...,z_l)=\ln N^{\mathrm{A}}(z_1,z_2,...,z_l)$，它描述了函数集合在给定数据集合上的多样性。定义 VC 熵 $H^{A}(l)=E\ln N^{\mathrm{A}}(z_1,z_2,...,z_l)$ 为 $\{f(z,\alpha)|\alpha\in \mathrm{A}\}$ 在数量为 l 的样本上的熵。它依赖于指示函数集合 $\{f(z,\alpha)|\alpha\in \mathrm{A}\}$，概率测度及观测数目 l,反映了指示函数集合在数目为 l 的样本上的期望的多样性。

学习理论形成了第一个里程碑：若要对一个学习过程使用经验最小化归纳原则，则要求对于要学习的指示函数集合的 VC 熵 $H^{\mathrm{A}}(l)$ 满足

$$\lim_{l\to\infty}\frac{H^{\mathrm{A}}(l)}{l}=0 \tag{24}$$

进一步，还对这个收敛的速度有一个要求，希望这个收敛速度是足够快的，以保证不仅 ERM 最小化原则一致，而且经验风险收敛到最小值的速度是快的。

首先，定义什么是收敛的渐进速度快。收敛的渐进速度快是指对于任何 $l>l_0$，都有下面的指数界成立：$P\{R(\alpha_l)-R(\alpha_0)>\varepsilon\}<\exp(-c\varepsilon^2 l),c>0$ 为常数。所以可以得到在模式识别指示函数集合的情况下，收敛速度快的充分条件是

$$\lim_{l\to\infty}\frac{H^{\mathrm{A}}_{ann}(l)}{l}=0 \tag{25}$$

这是学习理论的第二个里程碑。其中 $H^{\mathrm{A}}_{ann}(l)=\ln EN^{A}(l)$ 为褪火熵。

以上两个里程碑都是在针对特定的概率测度 $F(\vec{z})$ 上得到的。因为 VC 熵 $H^{\mathrm{A}}(l)$ 和退火熵 $H^{\mathrm{A}}_{ann}(l)$ 都是在尽管未知但是确定的概率测度 $F(\vec{z})$ 上构造的。但是，对于实际的情况，尽管我们假设样本都是独立的，并且同分布服从相同的概率测度 $F(\vec{z})$，但是只针对一个概率测度显然不能满足建造一个学习机器能够满足不同样本来自不同的概率测度的问题。因此，建立的第三个学习理论的里程碑是，确定不依赖于概率测度， ERM 原则一致且快的条件。对任何概率测度 ERM 具有一致性的充分必要条件是

$$\lim_{l\to\infty}\frac{G^{\Lambda}(l)}{l}=0 \tag{26}$$

而且在这个条件下，ERM 原则收敛的速度是快的。

为了构造实际有用的算法，除了考虑是否收敛，即 ERM 原则是否一致，还要考虑收敛速度的界。因为收敛速度的界，特别是收敛速度的上界，对于控制学习过程很重要，因为这暗示了在实际中的训练样本集合要多大，而不仅仅是满足一个训练样本充分大的渐进性态。

$N^{\Lambda}(\vec{z_1},\vec{z_2},...,\vec{z_l})$ 为函数集合 $Q(\vec{z},\alpha),\alpha\in\Lambda$ 对于 $(\vec{z_1},\vec{z_2},...,\vec{z_l})$ 的不同划分数。$H^{\Lambda}(l)=\mathrm{E}N^{\Lambda}(\vec{z_1},\vec{z_2},...,\vec{z_l})$ 为 VC 熵，其中的数学期望是针对 $(\vec{z_1},\vec{z_2},...,\vec{z_l})$ 进行的。定义退火的 VC 熵为 $H^{\Lambda}_{ann}(l)=\ln\mathrm{E}N^{\Lambda}(\vec{z_1},\vec{z_2},...,\vec{z_l})$。生长函数为 $G^{\Lambda}(l)=\ln\sup_{\vec{z_1},\vec{z_2},...,\vec{z_l}}N^{\Lambda}(\vec{z_1},\vec{z_2},...,\vec{z_l})$。并且有 $H^{\Lambda}(l)\le H^{\Lambda}_{ann}(l)\le G^{\Lambda}(l)$。下面两个不等式成立。

$$P\{\sup_{\alpha\in\Lambda}|\int Q(\vec{z},\alpha)dF(z)-\frac{1}{l}\sum_{i=1}^{l}Q(\vec{z_i},\alpha)|>\varepsilon\}\le 4\exp\{(\frac{H^{\Lambda}_{ann}(2l)}{l}-\varepsilon^2)l\}$$

$$P\{\sup_{\alpha\in\Lambda}\frac{\int Q(z,\alpha)dF(z)-\frac{1}{l}\sum_{i=1}^{l}Q(z_i,\alpha)}{\sqrt{\int Q(z,\alpha)dF(z)}}>\varepsilon\}\le 4\exp\{(\frac{H^{\Lambda}_{ann}(2l)}{l}-\frac{\varepsilon^2}{4})l\} \tag{27}$$

其中，如果把退火熵 $H^{\Lambda}_{ann}(2l)$ 替换成生长函数，因为 $H^{\Lambda}(l)\le H^{\Lambda}_{ann}(l)\le G^{\Lambda}(l)$，不等式仍然成立。而且因为生长函数与样本具体分布无关，所以有对任意分布都成立的下列两个不等式：

$$P\{\sup_{\alpha\in\Lambda}|\int Q(\vec{z},\alpha)dF(z)-\frac{1}{l}\sum_{i=1}^{l}Q(\vec{z_i},\alpha)|>\varepsilon\}\le 4\exp\{(\frac{G^{\Lambda}(2l)}{l}-\varepsilon^2)l\}$$

$$P\{\sup_{\alpha\in\Lambda}\frac{\int Q(z,\alpha)dF(z)-\frac{1}{l}\sum_{i=1}^{l}Q(z_i,\alpha)}{\sqrt{\int Q(z,\alpha)dF(z)}}>\varepsilon\}\le 4\exp\{(\frac{G^{\Lambda}(2l)}{l}-\frac{\varepsilon^2}{4})l\} \tag{28}$$

用上面的与分布相关或无关的不等式可以进一步得到学习机器推广能力的

界。学习机器的推广能力是指得到最小的经验风险的函数的真实风险是多少。而这个最小的经验风险对于可能的最小风险 $\inf_{\alpha} R(\alpha), \alpha \in \mathrm{A}$ 有多么的接近。在文本工作的模式识别背景下，下面将讨论对于学习函数类型是非负有界的情况进行讨论。

$0 \le \mathrm{Q}(z,\alpha) \le \mathrm{B}, \alpha \in \mathrm{A}$ 是有界非负函数的集合，那么不等式 $R(\alpha) \le \mathrm{R}_{emp}(\alpha) + \frac{B\varepsilon}{2}(1+\sqrt{1+\frac{4\mathrm{R}_{emp}(\alpha)}{B\varepsilon}})$ 以至少 $1-\eta$ 的概率同时对函数集合中的所有函数成立。不等式 $R(\alpha_l) - \inf_{\alpha \in \mathrm{A}} R(\alpha) \le B\sqrt{\frac{-\ln\eta}{2l}} + \frac{B\varepsilon}{2}(1+\sqrt{1+\frac{4}{\varepsilon}})$ 以至少 $1-2\eta$ 的概率对使经验风险最小的函数 $Q(z,\alpha_l)$ 成立。这两个依概率成立的不等式回答了上面的两个问题，即，给出了在取得最小的经验风险时实际风险的上界，给出了取得最小经验风险时，实际风险于可能的最小风险间的差别。

上面关于 ERM 原则一致性及收敛速度的讨论，导出了学习机器推广能力的界。但这个界是概念性而不是构造性的，不能直接用于构造 ERM 原则学习过程的算法。为此，必须找到计算褪火熵和生长函数的方法。这可以利用 VC 维的概念及其与生长函数间的关系来找到一个构造型的界——VC 维界。

一个指示函数集 $f(\vec{x},\alpha), \alpha \in \mathrm{A}$ 的 VC 维是能够被集合中的函数以所有可能的 2^h 种方式分为两类的向量的最大数目 h。也就是能够被这个函数集打散的向量的最大数目。如果对于任意的 n，总存在一个 n 个向量的集合可以被函数集合 $f(\vec{x},\alpha), \alpha \in \mathrm{A}$ 打散，那么函数集合的 VC 维无穷大。

VC 维有限是 ERM 原则一致性的充分条件，也是快的收敛速度的充分条件。

在 VC 维有限的情况下，考虑有限 VC 维 h 的函数集合，在这种情况下，有 $G^{\mathrm{A}}(l) \le h(\ln\frac{l}{h}+1), l > h$。因此，可以有下面的构造性表达式 $\varepsilon = 4\frac{h(\ln\frac{2l}{h}+1) - \ln(\frac{\eta}{4})}{l}$ 成立。如果函数集包含有限个元素且 VC 维有限，则

$\varepsilon = 2\frac{\ln N - \ln \eta}{l}$ 成立，其中 N 为函数集合中元素的个数。

进一步只针对模式识别的情况陈述如下。

L 个样本组成的样本集合，每个样本由一对 x 和 y 组成。其中 x 是 n 维向量，$\vec{x} \in R^n$，y 是和向量相联系的真值。比如在本文的工作中，$\vec{x} \in R^n$ 就是表示文本的向量，y 就是由专家预先认定的该文本是否属于指定类别的判定逻辑真假值。假设这些 $(\vec{x_i}, y_i)$ 都来自一个未知概率分布 $F(\vec{x}, y)$，它们独立同分布。我们的学习机器的任务就是要学习到从 $\vec{x}$ 到 y 的映射。实际上学习机器由一组可能的映射集合 $\vec{x} \mapsto f(\vec{x}, \alpha), \alpha \in A$ 定义，$f(\vec{x}, \alpha)$ 中的 α 是可调参数。称一个学习机器被训练好是指通过向学习机器输入训练数据，在 $\alpha \in A$ 中选定一个 α，得到一个确定的函数 f,使得 f 对新的输入向量能够确定地给出相应的逻辑真假值 y。在本文的工作中就是学习一个函数，使得能够对新到的文本向量输出 y 值，以指示文本是否属于指定的类别。一个已经训练好的学习机器的错误是指学习到的函数对输入计算的输出与实际应该的输出不一致。在本文的工作中就是指对新到文本向量输出的 y 值以指示其是否属于指定的类别与实际文本是否属于指定的类别不一致。简言之，比如对文本 t, 其向量表示为 $\vec{x}$，将其输入训练好的学习机器所学习到的函数 f, 产生的输出值 y=1,表示学习机器认为文本 t 属于指定的类别，而实际上文本不属于这个类别，y=-1,因此出现错误，造成的损失用下式计算：$\frac{1}{2}|y - f(\vec{x}, \alpha)|$ 为 1。若实际上文本属于这个类别，y=1, 则造成的损失按上式计算为 0。 因此，训练好的机器出错的期望值，也就是前述的期望风险，实际风险，$R(\alpha) = \int \frac{1}{2}|y - f(\vec{x}, \alpha)| dF(\vec{x}, y)$，这个式子也就是前面一节中所述的期望风险泛函 $R(\alpha) = \int Q(\vec{z}, \alpha) dF(\vec{z})$ 在模式识别下的特例。这里损失函数 $Q(\vec{z}, \alpha) = \frac{1}{2}|y - f(\vec{x}, \alpha)|$，$\vec{z} = (\vec{x}, y)$。而实际在训练集上出错的统计值为经验风险

$R_{emp}(\alpha)$。定义为 $R_{emp}(\alpha)=\frac{1}{2l}\sum_{i=1}^{l}|y_i-f(\vec{x_i},\alpha)|$，同样的这也是前一小节所述的经验风险在模式识别中的具体化，其中 $Q(\vec{z},\alpha)=\frac{1}{2}|y-f(\vec{x},\alpha)|$，$\vec{z}=(\vec{x},y)$。对于给定的训练集合和训练好的学习机器，$R_{emp}(\alpha)$ 是一个定值。

对指定的学习率 $\eta,0\le\eta\le1$，下面关于学习机器的推广能力的界，即不等式

$$R(\alpha)\le R_{emp}(\alpha)+\sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}} \tag{29}$$

以概率 $1-\eta$ 成立。其中 l 为训练集合的大小，h 为 $\{f(\vec{x},\alpha)|\alpha\in A\}$ 的 VC 维，也是“容量”。称不等式右半部分 $\sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}}$ 为 VC confidence 。这个界的优点在于它独立于数据的分布 $F(\vec{x},y)$，只是要求数据是独立同分布的。而且 VC 维对于学习机器函数集是构造性的，可计算的。

（四）结构风险最小化原则

上面就是关于 ERM 原则的总的理论框架。它是从处理大样本问题出发的。认为样本集是小样本，如果对于数目为 l 的样本，如果比值 l/h,即训练模式数目学习机器函数的 VC 维的比值，是小的。当比值 l/h 较大时，ε 就较小，因此不等式 $R(\alpha)\le R_{emp}(\alpha)+\frac{B\varepsilon}{2}(1+\sqrt{1+\frac{4R_{emp}(\alpha)}{B\varepsilon}})$ 右边第二项就小，于是实际风险于经验风险就更接近。即，较小的经验风险能保证比较小的实际风险。而当样本集大时，l/h 的比值较大，经验风险与实际风险的差别就比较大，就不能保证用经验风险最小化原则定义的学习过程能够得到和它接近的小的实际风险。在这种情况下定义结构风险最小化原则 SRM，同时最小化经验风险和学习机器推广能力实际风险上界满足的不等式的右边部分。

从学习机器推广能力的界不等式（28）以看到，VC confidence $\sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}}$ 对于 l/h 是单调递减的函数。因此，训练样本数量越大，学习机器函数集合 VC 维越小，VC confidence 越小，$R_{emp}(\alpha)$ 夹逼 $R(\alpha)$ 越紧，

在 VC 维小的函数集合中用经验风险最小的学习函数能够使实际风险也足够小。但是当样本容量不是很大时，这种夹逼也不太紧时，仅仅最小化经验风险不足以使学习机器的实际风险足够的小。这是就要采用结构风险最小化的原则 SRM，同时最小化实际风险的上界。具体做法是这样的：找到学习函数的一个子集，这个子集的 VC 维是最小的。在学习的函数集合中引入一种结构。这种结构对整个函数集合形成一个子集嵌套的划分。对每个子集计算其 VC 维 h 或 VC 维的上界。分别用函数子集训练学习机器，找到在相应学习机器函数集合中经验风险最小化的函数。形成一系列的训练好的学习机器。计算每个训练好的学习机器的经验风险和对应函数集合的 VC confidence 的和，即不等式 $R(\alpha) \le \mathrm{R}_{emp}(\alpha) + \sqrt{\frac{h(\log(2l/h)+1) - \log(\eta/4)}{l}}$ 的右边，取取值最小的那个作为最后学习到的函数[40]。

2.3.2 支持向量机

基于结构最小化原则就可以构造支持向量机了。那就是保持经验风险值固定，并最小化置信范围[41]。

（一）数据线性可分

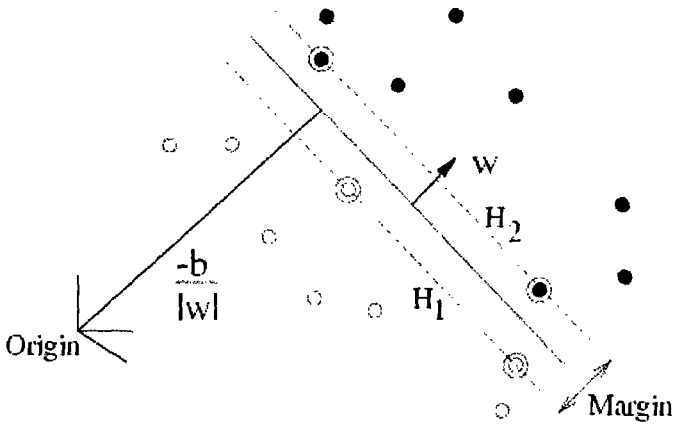


图 5 线性可分示例

定义 假设训练数据 $(\vec{x}_1, y_1), \ldots, (\vec{x}_l, y_l), \vec{x} \in R^n, y \in \{+1, -1\}$ 可以被一个超平面 $\vec{w} \bullet \vec{x} - b = 0$ 分开。如果这个向量集合没有被超平面错误的分开，并且离超平面最近的向量于超平面之间的距离是最大的，则称这个向量集合被这个最优超平面分

开。（Vapnik Chervonenkis 1974 , Vapnik 1979）

定义 若 $y_i=1$， $\vec{w}\bullet\vec{x_i}-b\geq 1$

若 $y_i=-1$， $\vec{w}\bullet\vec{x_i}-b\leq 1$

亦即 $y_i(\vec{w}\bullet\vec{x_i}-b)\geq 1, i=1,2,...,l$ 为分类正确。

等价于 $y\in\{1,0\}$ 时，定义 $y_i(\vec{w}\bullet\vec{x_i}-b)\geq 0$ 为分类正确。

最优超平面就是满足上式 $y_i(\vec{w}\bullet\vec{x_i}-b)\geq 1, i=1,2,...,l$ 并且使得 $\|\vec{w}\|$ 最小化的超平面。（如图 5）

定义 $\Delta-$间隔分类超平面

$\vec{w}^*\bullet\vec{x}-b=0, \|\vec{w}^*\|=1$ 能够以如下形式将向量 $\vec{x}$ 分类。

$y=1$,若 $(\vec{w}^*\bullet\vec{x})-b\geq\Delta$

$y=0$,若 $(\vec{w}^*\bullet\vec{x})-b\leq\Delta$

最优超平面是 $\Delta-$间隔分类超平面，其中 $\Delta=1/\|\vec{w}\|$。

定理 设向量 $\vec{x}\in X$ 属于一个半径为 R 的球。那么 $\Delta-$间隔分类超平面的集合的 VC 维 h 以不等式 $h\leq\min\left(\left[\frac{R^2}{\Delta^2}\right],n\right)+1$ 为界。

推论 测试样本不能被 $\Delta-$间隔超平面正确分类的概率（对应于实际风险）的上界以概率 $1-\eta$ 成立。$P_{error}\leq\frac{m}{l}+\frac{\varepsilon}{2}\left[1+\sqrt{1+\frac{4m}{l\varepsilon}}\right]$，其中，$\varepsilon=4\frac{h(\ln\frac{2l}{h}+1)-\ln\frac{\eta}{4}}{l}$，m 是没有被 $\Delta-$间隔分类超平面正确分类的训练样本数目，h 为定理给出的 VC 维的界。

在以上定理陈述的事实基础上，我们可以看到，按照保持经验风险值固定，并最小化置信范围构造 SRM 原则定义的学习机器是可行的。$\Delta-$间隔分类超平面对于训练数据集合的经验风险为 0(因为所有的训练数据被正确划分)，这是在固定经验风险。最小化置信范围转化成了最小化 $\Delta-$间隔分类超平面的 VC 维 h，

而由于 $h \le \min(\left[\frac{R^2}{\Delta^2}\right], n)+1$，这就转化为使$\Delta$最小。即是对于线性可分数据，寻找最优超平面。

而如何求解最优超平面，就是支持向量机算法问题。由于最大边缘问题已经可以理解为两个闭凸集合之间保持最大距离的问题，因此，有大量的方法可以实现。

（二）线性不可分的情况

转化为线性可分的情况进行研究。这就是要寻找合适空间，设计一个映射，将定义在样本空间上的样本映射到这个空间，使得在这个空间中，样本集可以表示为线性可分问题。这个空间就是 Hilbert 空间。Hilbert 空间是 Von Neumman 为量子力学数学基础提出的一类具有一般意义的线性内积空间。$X=(x_1,\ldots,x_n)\phi\rightarrow|(X)=(\phi_1(X),\ldots,\phi_N(X))$。$\phi$ 是输入空间(样本空间)到特征空间的一个映射。$(\phi_1(X),\ldots,\phi_N(X))$说明可以采用不同映射构成特征空间的基。一般的说，一个线性不可分问题变换到特征空间，使得在特征空间上，这个问题是线性可分的，这个特征空间的维数可能非常高。这给计算带来困难。可以考虑不显式地描述特征空间，而将特征空间上的描述变换为样本空间上的描述。这在泛函分析上是可行的。这就是核函数的方法。因为没有必要显式的表示出原始空间到 Hilbert 空间的映射函数，而只需表示出其点积的形式（1992 Boser Guyon and Vapnik）。

考虑在 Hilbert 空间中的内积一般的表达 $(\vec{z_i}\bullet\vec{z})=K(\vec{x},\vec{x_i})$，其中，$\vec{z}$ 是输入空间中的向量$\vec{x}$在特征空间中的像。根据 Hilbert-Schmidt 理论，$K(\vec{x},\vec{x_i})$可以是满足下面一般条件的任意对称函数（Courant and Hilbert,1953）。

定理（Mercer）要保证 L2 下的对称函数 $K(u,v)$ 能以正的系数 $a_k>0$ 展开成 $K(u,v)=\sum_{k=1}^{\infty}a_k\varphi_k(u)\varphi_k(v)$ 的充分必要条件是 对使得 $\int g^2(u)du<\infty$ 的所有 $g\neq 0$ 条件 $\iint K(u,v)g(u)g(v)dudv>0$ 成立。

支持向量机实现这样的思想，即，它通过某种事先选择的非线性映射将输入向量 x 映射到一个高维的特征空间 Z,在这个特诊空间中构造最优分类超平面。如图 6 所示（要在 n 维空间中构造阶数 $d\ll n$ 的分类多项式，就需要多于 $(n/d)^d$ 个

特征。）

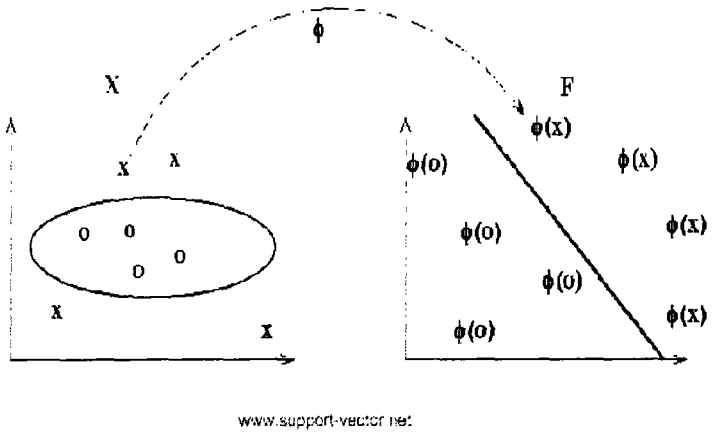


图 6 空间映射示意

内积的回旋可以构造在输入空间中非线性的决策函数 $f(\vec{x})=\mathrm{sgn}(\sum y_i a_i K(x_i,x)-b)$ 等价于在高维特征空间 $\varphi_1(x),...,\varphi_N(x)$ 中的线性决策函数。（$K(x_i,x)$ 是这个线性空间中内积的一种回旋）

要求得系数 a_i 只要寻找泛函 $W(a_i)=\sum_{i=1}^{l}a_i-\frac{1}{2}\sum_{i,j=1}^{l}a_i a_j y_i y_j K(x_i,x_j)$ 在约束条件 $\sum_{i=1}^{l}a_i y_i=0$ 下的最大值。

采用不同的函数作为内积的回旋 $K(x_i,x)$ 可以构造在输入空间中不同类型的非线性决策面的学习机器。在本文的工作中，采用多项式的学习机器。

要构造 d 阶多项式决策规则，可用下面的函数作为内积的回旋。$K(x,x_i)=\left[(x\bullet x_i)+1\right]^d$。由上述的不等式 $h\le\min(\left[\frac{R^2}{\Delta^2}\right],n)+1$，要估计 VC 维必须估计包含训练数据的最小的超球半径 R 以及在特征空间中权值的模。这都依赖于多项式的阶数 d。

这个问题就进一步转换为在某个兴趣点 x0 邻域内的一个局部多项式逼近，根据局部算法的理论，在 x0 周围选择一个半径为 R_β 的球，训练集中 l_β 各元素落入这个球中，然后只用这些数据构造一个决策函数，使得在选定的邻域内错误的

概率最小。这就是下面的问题的解。使泛函 $\Phi(R_\beta, w_0, l_\beta) = \frac{R_\beta^2 \|w_0\|^2}{l_\beta}$ 最小的半径 R_β。

2.3.3 Knn

该算法的基本思路是：在给定新文本后，考虑在训练文本集中与该新文本距离最近（最相似）的 K 篇文本，根据这 K 篇文本所属的类别判定新文本所属的类别，具体的算法步骤如下：

1:根据特征项集合描述训练文本向量

2:在新文本到达后，根据特征词分词新文本，确定新文本的向量表示

3:在训练文本集中选出与新文本最相似的 K 个文本，计算公式为：

$$\mathrm{Sim}(d_i, d_j) = \frac{\sum_{k=1}^{M} W_{ik} \times W_{jk}}{\sqrt{(\sum_{k=1}^{M} W_{ik}^2)(\sum_{k=1}^{M} W_{jk}^2)}} \tag{30}$$

其中，K 值的确定目前没有很好的方法，一般采用先定一个初始值，然后根据实验测试的结果调整 K 值，一般初始值定为几百到几千之间。

4:在新文本的 K 个邻居中，依次计算每类的权重，计算公式如下：

$$p(\vec{x}, C_j) = \sum_{\vec{d}_i \in KNN} Sim(\vec{x}, \vec{d}_i) y(\vec{d}_i, C_j)$$

其中，$\vec{x}$ 为新文本的特征向量，$Sim(\vec{x}, \vec{d}_i)$ 为相似度计算公式，与上一步骤的计算公式相同，而 $y(\vec{d}_i, C_j)$ 为类别属性函数，即，如果 $\vec{d}_i$ 属于类 C_j，那么函数值为 1，否则为 0。

5:比较类的权重，将文本分到权重最大的那个类别中。

直观而言，就是看在新到文本向量的事先指定的 K 个邻居中，哪一个类别占多数，就认为新到文本属于哪个类别。

这种方法的好处在于不用事先花时间训练学习器，但是其对新到文本的在线分类时间会比较长。而且，K 值的确定是靠在实验中不断的调整，还没有好的办法事先确定。不如支持向量机有坚实的理论基础。

为了检验本文工作提出的特征降维算法对分类的效果的影响不会是因为这

种方法对特定的分类学习算法有所偏好，也就是说，特征降维在不同维数上表现出的对最终分类效果造成的提升不是因为这种方法针对了特定的学习算法，本文工作也同时用 knn 算法对降维后的文本向量进行计算分类，以作为向量输入支持向量机的分类性能与维数关系的对比研究。实验的具体分析参见 4.5 节。

2.3.4 多分类问题

上文所述的支持向量机主要针对单分类问题，但是在文本的工作中，文本的分类体系不止一个类别，因此是一个多分类问题。因此在这节专门讨论如何将单分类问题转化为多分类的实现。

首先，对于 Knn 算法，多分类问题的实现是自然的。因此，文本向量可以属于任意个类别，实际文本所属于的类别以与其最近的占多数的文本向量的类别确定。

然而对于支持向量机而言，问题并不是那么直接。首先，回顾一下这个学习算法的原理可能对于解决这个问题是由帮助的。支持向量机的理论基础是统计机器学习、泛化理论。用函数的划分能力 VC 维，最大边缘来估计假设模型对于泛化误差，指导算法的设计。其次，由 Vapnik 给出了经验风险最小化的概念性和构造性的等价条件以及结构风险最小化原则。最后，给出了适用于本文工作的模式识别问题下的可用于具体构造算法的最大边缘泛化误差界。这就是支持向量机器。因此，可以考虑采用待分类向量在每一类训练好的支持向量机中被判断为是否属于该类的信度大小排序确定该向量文本所属的类别。也就是说，采用“投票”的方式确定。当有冲突发生时，比如有两个以上的类别的支持向量机都称一向量属于自己的类别，则离分类器支持向量最远的向量所在的分类器的类别为这个向量所属的类别。同理，对于没有类别的支持向量机称某一向量属于自己的类别时，也采用投票的方式。即，离分类器支持向量最近的向量所在的分类器的类别为这个向量所属的类别。

2.3.5 小结

本节主要介绍了在文本分类第二阶段中采用的学习算法，支持向量机和 Knn。在这之前详细介绍了作为支持向量机理论基础的统计机器学习，泛化理论，以及

一般的机器学习和人工智能的概念。作为实验观察规律和工程实现提高效果，文本工作文本分类采用两种机器学习的算法，以做出类比。

2.4 分类性能的评价

2.4.1 性能评价的方法

完成文本向量的表示并输入学习机器进行训练后，就可以用训练好的机器对新到文本进行分类了。新到文本也按降维后保留的特征和与训练文本一样的索引方法计算其向量表示。这样的分类程序分类的效果如何呢？这不仅是未来用户关心的问题，也是研究者关心的问题。对性能准确的评价有助于指导算法的新的设计和改进。

性能测试的基本方式是使用测试语料库。按照测试语料库是否与训练语料库存在交集的情况把测试分为封闭测试和开放测试。事实上，现在通常使用开放测试，这更能说明学习的实际效果，学习机器的泛化能力。通常希望测试语料库是公有的标准语料库，这样方便各种的不同的分类方法分类效果的对比。目前针对中文文本分类的标准语料库的建设还很缺乏，主要有中国科学院计算技术研究所的863基础测评平台。

本文采用的测试语料库是自然语言处理开放平台上提供的中文文本分类语料库，采用的是开放测试的方法。

2.4.2 性能评价的指标

具体而言，分类性能可以由以下指标进行衡量：宏-准确率，宏-召回率和宏-F1 [10]. 宏-准确率Macro-precision 入下定义[42]:

$$MacroP = \frac{1}{m}\sum_{i=1}^{m} p_i \tag{31}$$

其中 p_i 如下定义

$$p_i = \frac{l_i}{m_i} \tag{32}$$

l_i 表示被分类器正确标记为类别i的文本的数量，m_i表示被分类器标记为i类的文本的总数量。因此p_i 就是对于类别i的分类的正确率。m是文本分类器的分类体

系中类别的数目。

宏-召回率如下定义

$$MacroR = \frac{1}{n}\sum_{i=1}^{m} R_i \tag{33}$$

其中 R_i 如下定义：

$$R_i = \frac{l_i}{n_i} \tag{34}$$

l_i和m的定义如上所述，n_i是实际应该属于类别i的文本的数量。即R_i定义分类器对类别i的召回率。

而宏-f1是上述二者的一种平均，其定义如下：

$$MacroF1 = \frac{MacroP \times MacroR \times 2}{MacroP + MacroR} \tag{35}$$

这些指标反映了文本分类器的分类性能。也是本文工作所采用的分类性能评价指标。

第三章 文本分类的工程实现

大规模内容计算需要仔细考虑算法的时间空间复杂度。前面章节介绍的只是方法的逻辑视图，具体在计算机程序上应该如何设计还要仔细考虑。这就是本章的主要内容。下面将从程序涉及的主要数据结构和主要过程描述按本文所述方法实现文本分类器。

另外，本章还结合试验结果，分析了第二章提出的新的特征降维算法对整体分类效果的影响。

3.1 试验设置

语料库，简而言之就是数据的集合，具体指文本数据的集合。直观的，就是一些计算机文件的集合，每个文件是按某种编码方式比如 ASCII 存储的自然语言书写的文本，称为文本文件。训练语料库是指用于训练学习机器的文本集合，而测试语料库中的文本用于输入训练好的学习机器，以输出其应属于的类别。对测试语料库中文本依次进行这样的测试，统计输出类别的准确率和召回率及其他指标，就可以对分类任务完成的情况作出评价。在自然语言处理领域，测试语料库如与训练语料库相同，称为封闭测试，如果完全不同，则称为开放测试。

在本文的工作中，采用开放测试。语料库来自中文自然语言处理开放平台（www.nlp.org.cn ）上的李荣陆整理的分类语料库。语料库共有政治、艺术、医药、体育、军事、经济、教育、交通、计算机、环境十个类别的文章，每一类有从 200 到 500 不等数目的文本。我们把这个语料库切分成两个互不相交的部分分别作为训练语料库和测试语料库。两个库都包含相同的类别以作检验。

3.2 体系结构

问题定义

系统的功能：对中文自然语言的文本，按照其主题，对其进行分类。

系统的角色和意义 ：

1） 作为实验系统，观察统计分类在测试文本集合上的效果，试验检验完成文本分类的方法的优劣。尤其是本文工作中独立提出的一

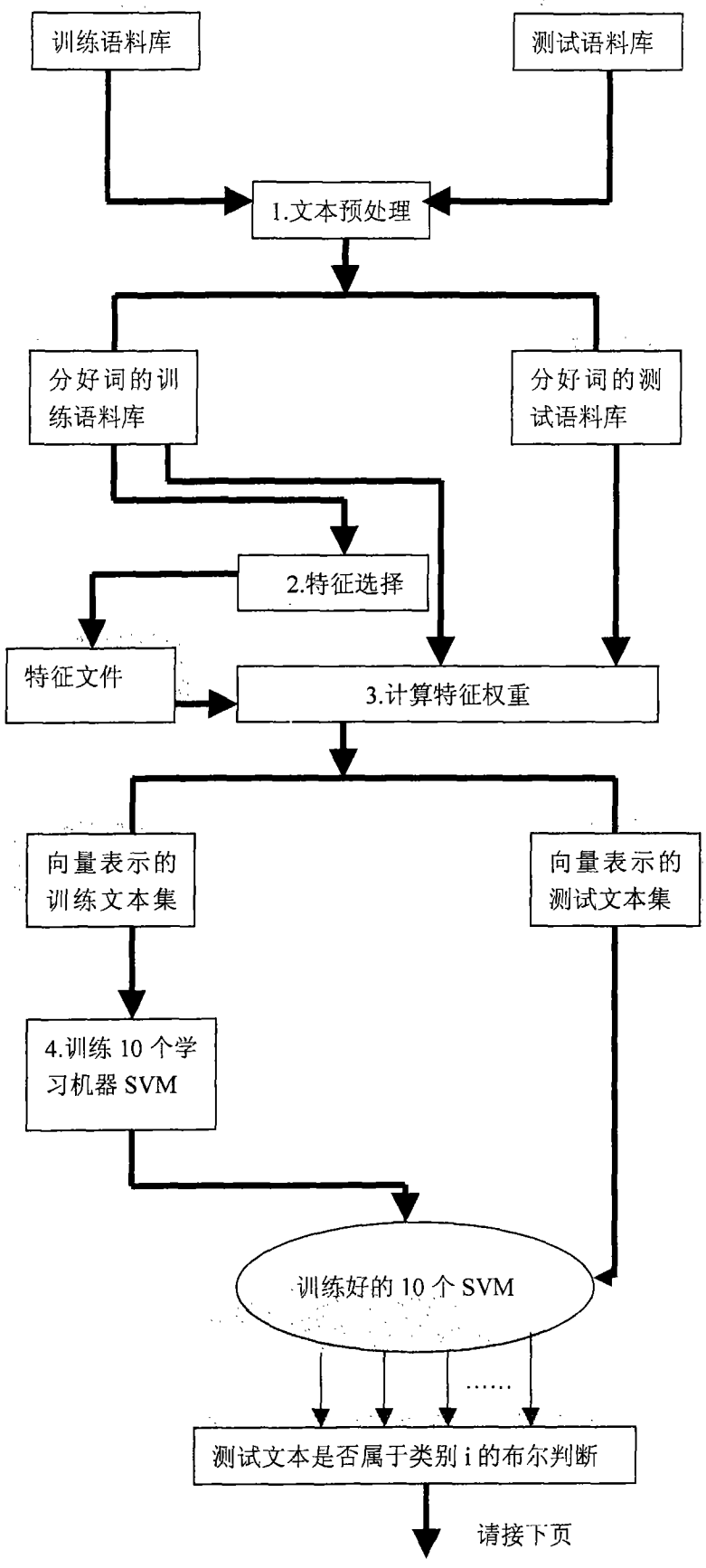
种新的特征选择方法的试验分析。

2） 作为一种应用，能够对新到的文本进行分类。

3） 作为更为实际的应用的核心，比如后文例举的自动的垃圾邮件过滤，以及实时个性化的招聘信息服务的应用。

整体架构（逻辑流程）详见下页图 7。

有阴影的框图表示程序处理的结果或数据文件。没有阴影的框图是处理的过程。云图是注释和解释。


训练语料库
测试语料库
1.文本预处理
分好词的训练语料库
分好词的测试语料库
2.特征选择
特征文件
3.计算特征权重
向量表示的训练文本集
向量表示的测试文本集
4.训练 10 个学习机器 SVM
训练好的 10 个 SVM
.....
测试文本是否属于类别 i 的布尔判断
请接下页

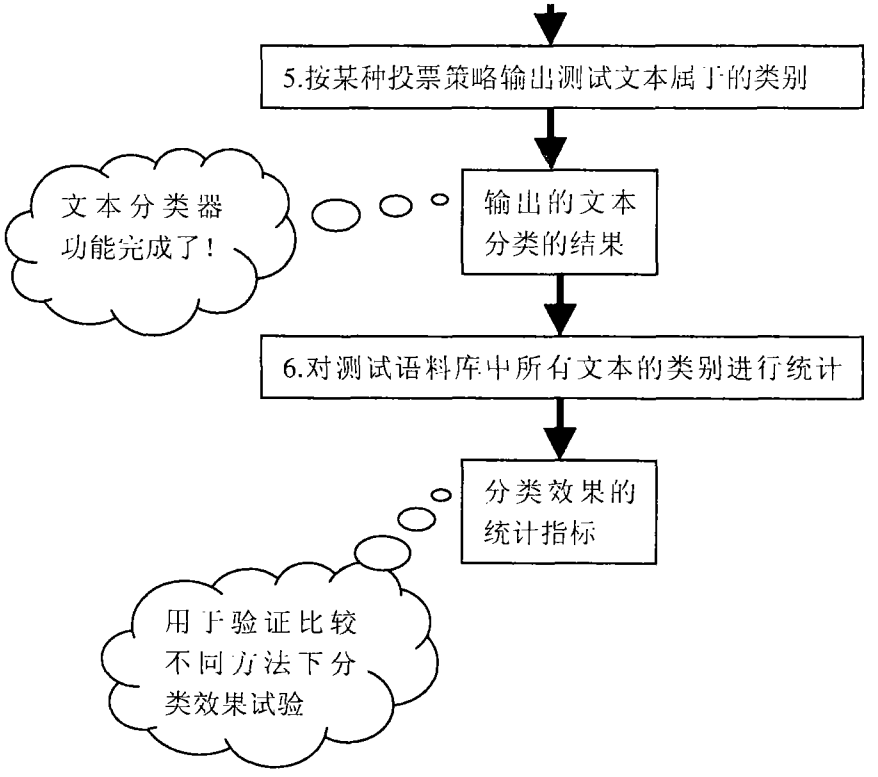


图 7 程序总体流程图

3.3 核心数据结构——词表

降维前原始词的集合是所有训练语料库中出现词的集合，即每篇文本所有出现词语的并集。通常为了达到好的学习效果，希望训练样本集合的质量足够好，容量足够大。这时，在语料库中出现的所有词的数量是很大的，汉语所有词语有 11 万之巨，而本文工作涉及的小型文本库中的词语也有上万。因此，应该考虑用一种高效的数据结构来存储这些词语条目。这里，将之称为词表。

通常最直观的词表是采用数组方式存储，但是在本文工作的环境中是不合适的，因为词是随着读入的训练文本不断增加的，因此插入操作是极其频繁的，数组的效率不高。如用链表，则因为在使用词表时是希望能迅速找到词条，因此也是不合适的。本文工作采用 hash 的方式。这样在扫描文本不断加入新词的过程，

按下面的方式进行即可。按照设计的 hash 函数计算当前词语的 hash 值，在 hash 表中找到对应表项，若该表项为空，则填入这个词的结构；若不为空，则比较是否发生了冲突，若没有冲突则更新该词的统计信息，否则，采用链的方式消解冲突。数据结构如下图所示，定义参见紧接的后文。图 8 中所述数据结构的状态是在读入分好词的训练文本时，建立词语过滤的依据——各个词语统计值——时的状态。待扫描完整个训练语料库后，就可以把统计项链归纳为一个词语的统计项，如图 9 。然后依此统计项的值对 hash 表中的词语项进行稳定的排序，词语位置不变，依次求词语统计项值最高的若干个（指定的降维后的空间维数）词语项，在相应 hash 表项中标记保留，形成图 10 所示的数据结构，写入文件，即形成特征文件，在计算文本的向量在这些特征维上的值时读入内存，作为文本向量 indexing 的依据。

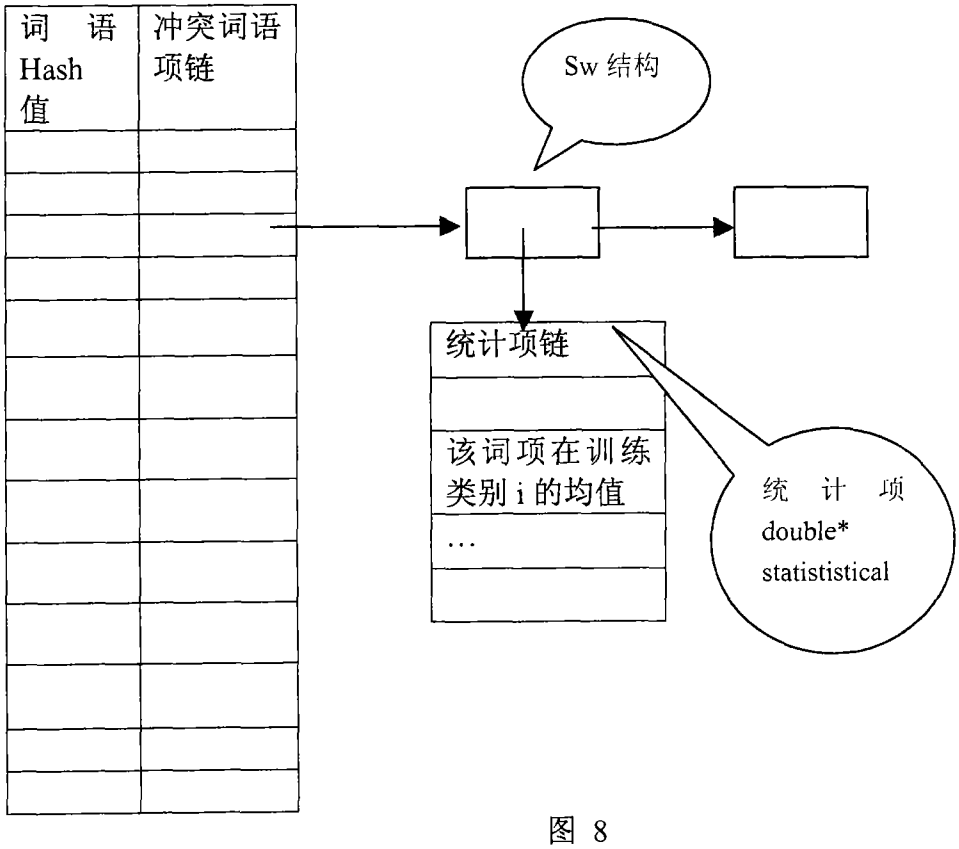


图 8

词 语 Hash 值	冲突词语项链

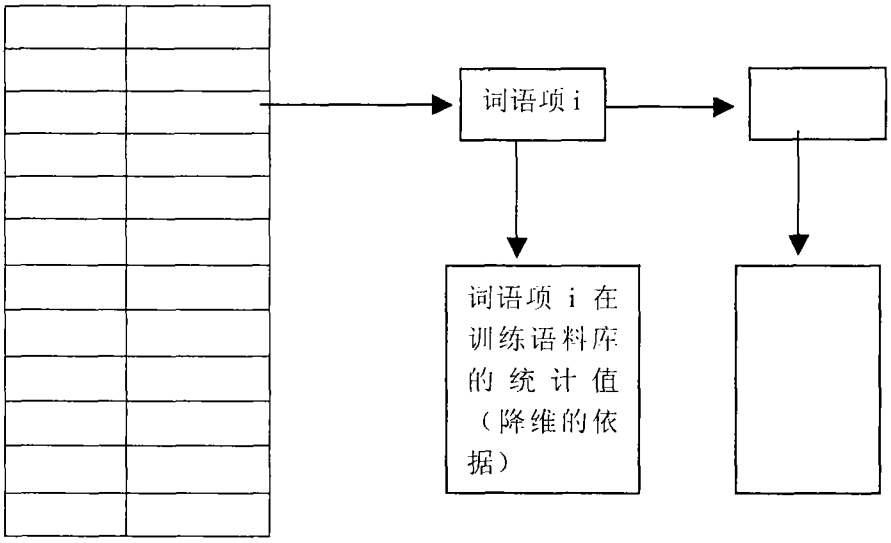


图 9

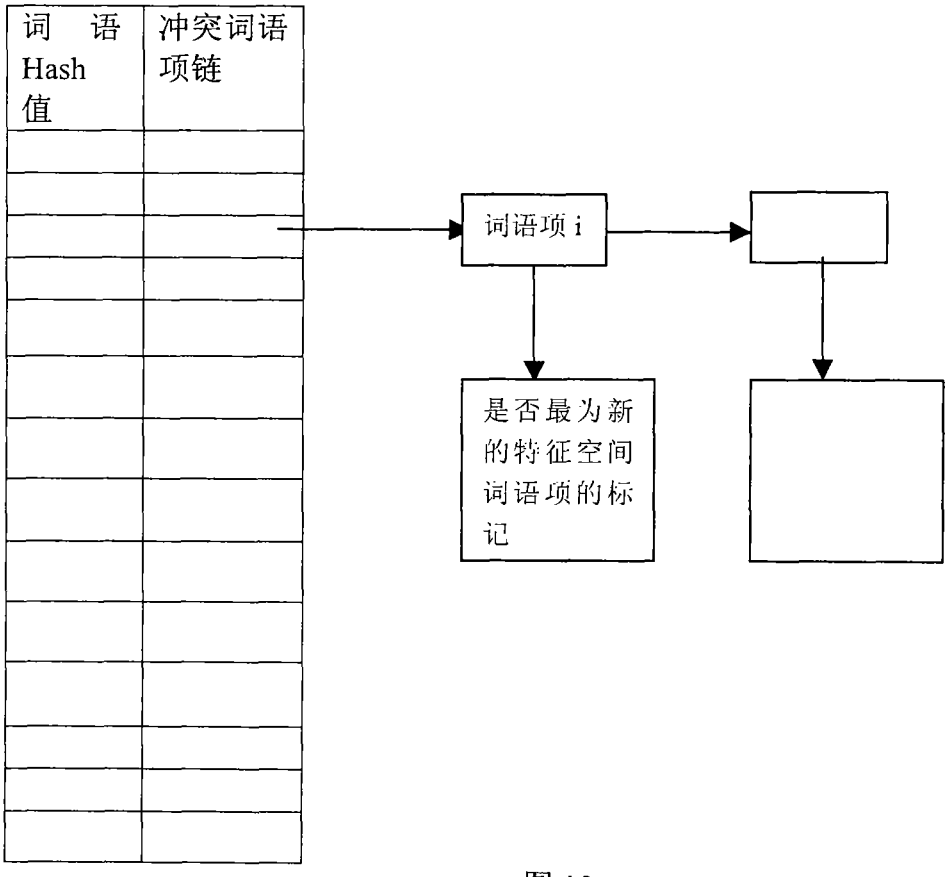


图 10

```
struct sw
{
    string content; //词语
    double* statistiscal;//词语统计值指针
    sw* next; //冲突链下一项
}
typedef long int signiture;
struct CibiaoItem
{
```

```
    signiture index;//词语的 hash 值
     struct sw*    link;
}
class Hash
{
 CiabiaoItem []Hasharray;
 long int _size;
public:
   signiture   Find(string);
   bool Insert(sw*);
   operator[](signiture);
}
```

3.4 主要算法过程

3.4.1 文本预处理

如前所述，文本预处理是指对文本进行词法分析，输出文本的词语序列。分词包括未定义词的识别和排歧。未定义词泛指没有在词典中出现过的词，包括a)命名实体(各种机构名、人名、地名) b)通过离合重叠词缀等方式构成的词典上没有的词 c)对应于新的概念的"新词", 比如"超女" 。因此,随着词典的不断进步扩大和新概念的出现,可以认为,未定义词的范围在发生变化。排歧是指在多种词语切分的可能中选择一个最可能正确的结果。在本文前述的分词方法中，就是计算候选词语序列的概率，取概率最大的切分结果。

在本文的工作中，我们采用中国科学院计算技术研究所开发的开源词法分析程序 ICTCLAS 进行文本的预处理。这个分词程序的原理是采用统计的途径进行分词，具体就是隐马尔科夫模型（HMM）。更准确的说，这个隐马尔科夫模型是层叠的隐马尔科夫模型，并结合了 N-最短路径的方法。首先用 N-最短路径的方法对文本进行粗切分，保留歧义的结果。然后将之输入连续的两个训练好的HMM进行命名实体的识别和排歧处理。再后，把识别出的未定义词按照和在普通词典中出现过的词同等的地位进行处理，即输入 Class-based HMM,最后将结果输入下一层HMM进行词性标注。

经过词法分析后的文本不再是连续的字序列，而是形如“词 / 词性”这样格式的词的序列。具体的程序实现的问题陈述如下。

1) 调用 ICTCLAS 的方式。直接的方式是由人工在程序界面上进行操作。另

M进行命名实体的识别和排歧处理。再后，把识别出的未定义词按照和在普通词典中出现过的词同等的地位进行处理，即输入 Class-based HMM,最后将结果输入下一层HMM进行词性标注。

经过词法分析后的文本不再是连续的字序列，而是形如“词 / 词性”这样格式的词的序列。具体的程序实现的问题陈述如下。

1) 调用 ICTCLAS 的方式。直接的方式是由人工在程序界面上进行操作。另外可以选择把已有程序作成 windows 平台上的 console application 的应用的动态链接库 dll。也可以把 ICTCLAS 作为一个新的进程进行调度。

2) 依次处理训练语料库和测试语料库的文本文件。经过 ICTCLAS 的处理后，文本文件成为了词的序列的文件。本文工作中还调用到了操作系统的文件方面的接口函数，把语料库中分好词的文本文件重新编号整理成了“类别编号_类别中文本号”这样格式的文件名。以方便后续处理。

3.4.2 特征降维

特征降维主要围绕上述数据结构展开，参见图 8.图 9 图 10。

图 8 的 hash 表表项是在扫描训练语料库的文本的过程中逐步形成的。算法如下。

```
for(int i=0;i<m;i++) //遍历每一类文本
{
for(int j=0;j<a[i];j++) //遍历每一类文本中每一篇文本 a[i]记录 i 类中文本数量
    {    while (word=GetWord(i,j))
    {
         if((sw* lk=Hash.Find(word))==0)
         //已经扫描过的文本中还没有出现该词语
         {
             sw* link=new sw;
             link->content=word; //将该词语填入词表
             link->remain=0;
             link->statistical=new double[m];
```

```
                //记录各个类别文章中该词语出现的均值
                  link->next=0;
                  for(int k=0;k<m;k++)
                  {
                        link->statistical[k]=0;
                  }  //初始化各个项为 0
                    link->statistical[i]++;//当前文本中出现这个词 于是该词相应类统计计数增 1
                    Hash.Insert(link);
              }
              else //词表中已有此项
              {
                    link->statistical[i]++;//该词相应类统计计数增 1
              }
        }
        }
    }//之后形成图 9 的状态
    然后收集每个词语项的各个类的统计信息成为过滤指标值。程序如下。
    for(int inter=Hash.start; inter<Hash.end;inter++) //对 hash 词表中每一项
    {
        for (sw* lk=Hash[iter].link;lk!=NULL;lk=lk->next) //对 hash 词表中每一项（冲突）的每个词语
        {
            double sum=0;
            double bar=0;
            double critia=0;
            for(int k=0;k<m;k++)
            {
                lk->statistical[k]=lk->statistical[k]/a[k];
```

```
        } //每个类别的均值
        for(int k=0;k<m;k++)
        {
            sum+=lk->statistical[k];
        }
        bar=sum/m; //收集各个类别统计信息：计算各个类别均值的均值，作为方差计算的准备
         for(int k=0;k<m;k++)
        {
            critia+=(lk->statistical[k]-bar)^2;
        }
        delete []lk->statistical;
        lk->statistical=new double(critia); //计算好方差填入词表项 此时数据结构状态如图 10 所示
    }
}
```

再后按这个统计值由大到小排序，取前 k 个词语作为保留特征项。

```
Hash::TagRemain()
{
    sort(k，remainword);
  // 对 hash 表中存入的词语项按其统计值域即 statistical 域由大到小进行排序，取前 k 大入新数组
    for(int t=0;t<k;t++)
    {
      string content=remainword[t].content;
       Hash[hash(content)].Find(content)->remain=1;
    //在 hash 表中标注应该保留相应词项
    }
    for(int inter=Hash.start; inter<Hash.end;inter++) //对 hash 词表中每一项
    {
```

```
        for (sw* lk=Hash[iter].link;lk!=NULL;lk=lk->next)
      //对 hash 词表中每一项（冲突）的每个词语
        {
            if (lk->remain==0) Hash.contract(lk->content);
      //对不保留的词语进行压缩以减少 hash 表的冲突
        }
    }
}
```

最后形成结果见图 10。可存入外存形成特征文件，作为计算文本向量 index 的特征维依据。

3.4.3 向量生成

读入特征文件，即标记了保留词的词语的 hash 表，按照 hash 表中标记了的词语的顺序计算出文本在这些词特征项上的出现频率。下面给出计算文本集合的文本的向量的程序。

```
for(int i=0;i<n;i++) //遍历每一类文本
{
    for(int j=0;j<c[i];j++) //遍历每一类文本中每一篇文本
    {
        long int wordtotal=0;
        while (word=GetWord(i,j)) //读每篇文本的每个词语
        {
            wordtotal++; //记录所有词次数
            sw* lk=Hash.Find(word);
            if(lk->remain= =1)
            {    if (lk->statistical= =NULL)
                lk->statistical=new double(0);
                 else
                   (*lk->statistical)++; //相应词语项出现在该篇文章中出现计数
```

加 1

```
            }      //end if
        }  //end while
         for(int iter=Hash.start; inter<Hash.end;iter++) //对 hash 词表中每一项
        {
            for (sw* lk=Hash[iter].link;lk!=NULL;lk=lk->next) //对 hash 词表中每一项（冲突）的每个词语
                {      if(lk->remain==1)
                    {   (*lk->statistical)=(*lk->statistical)/wordtotal;//计算词频
                        write(Termfile,i,j,(*lk->statistical))//每一维写入文件 形成相应文本的向量
                        (*lk->statistical)=0; //清 0，为下一个文本向量计算做准备
                    } //end if
                } //end for
        } //end for hash item
    } //end for each text in a class
} end for each class
```

3.4.4 训练学习机器，计算文本的类别

用训练语料库中的文本向量文件训练学习机器——支持向量机。因为本文工作中采用二分类的支持向量机，因此对于每个类别都训练一个相应的支持向量机。约定，称对应于类别 i 的支持向量机为向量机 i。支持向量机学习的原理已在前述章节中详细分析过了，这里不再鳌述。

下面描述对每一类文本训练相应支持向量机的实现过程。本文工作中，我们采用在自然语言处理开放平台（www.nlp.org.cn）上公布的 SVM。这个支持向量机基于著名的 SVM^{light}。采用支持向量机有诸多好处[42]。对第 i 类的支持向量机，修改第 i 类别的文本向量编号为 1，表示属于这个类别，修改除了第 i 类以外的文本向量的类别编号为 0，表示不属于这个类别。这样对支持向量机 i 就有 a[i]

个正例加$\sum_{j}^{j\neq i} a[j]$个反例，将这些向量合并为一个文件，作为训练文件 tr_i 输入支持向量机。再将测试语料库的向量表示文件也作类似的处理形成测试文件 t_i 输入训练好的支持向量机。支持向量机将输出结果文件，指明测试文本以多大的信度属于这个类别。

依次对十个类别训练十个支持向量，得到十个结果文件。对每个测试文本向量，对比其在十个结果文件中的记录，采用投票的方式决定测试文本属于的类别。若向量仅在 i 类结果文件中对应结果为 1，则属于类别 i；若同时存在若干类结果文件中同一向量的对应结果为 1，则取信度最大的那个类别作为那个向量对应的测试文本的类别。若一个向量在所有的类别结果文件中对应结果都为 0，则指定向量对应的测试文本的类别为在类别结果文件中信度最小的那个类别。最后就得到了由学习机器所判定的所有测试文本的类别。

3.4.5 统计性能评价指标值

对比由学习机器判定的类别和测试文本实际属于的类别，按前述章节对每一个类别计算分类性能指标——准确率 p 召回率 r。$p=\frac{a}{a+b}$，$r=\frac{a}{a+c}$。其中 a,b,c

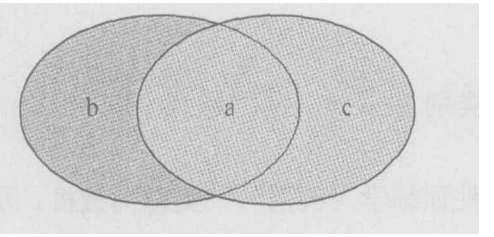


图 11

的含义如图 11 所示。

左边的椭圆代表程序判定属于某个类别的测试文本。右边的椭圆代表原本属于某个类别的测试文本，其交集表示由程序和人工同时判定属于某个类别的测试文本。定义针对某个类的准确率和召回率的折中评价指标 $f=\frac{2*p*r}{p+r}$。

对每个类得到的这些指标分别进行平均，得到宏准确率 M-P=$\sum_{\text{所有的类别}}$ p，宏召回率 M-R，M-R=$\sum_{\text{所有的类别}}$ r，宏 f 值 M-F，$\text{M-F}=\frac{2*\text{M-P}*\text{M-R}}{\text{M-P}+\text{M-R}}$。

这样就得到了用本文新提出的分类方法加支持向量机形成的分类系统的分类性能的量化指标。

3.5 性能评价的试验结果及其分析

上面提到的性能指标，是指用训练好的分类器对新的测试文本进行分类，结果正确的总体效果。在测试文本集合中，对于每一个指定的类，分类器分类的结果是有 b+a 篇文本属于这个类别，而由专家判别的实际 b+a 篇文本中只有 a 篇文本属于这个类别，那么对于这个类别分类的准确率为 pi=a/(a+b), 而实际由专家判别的属于这个类别的文本有 c+a 篇，因此对于这个类别分类的召回率为 r_i=a/(a+c)。所谓召回率的意思是，有多大比例的属于这个类别的文本被分类器识别，所谓准确率的意思是，有多大比例的分类器识别属于这个类别的文本被认为是真正属于这个类别的。

这样把每个类别的 pi 进行平均，就得到宏平均准确率 M-P，对每个类别的 ri 进行平均就得到宏平均召回率 M-R。但是通常这两个指标值不能同时都很高，实际的系统是二者的某种折中。通常采用加权平均公式进一步计算得到一个性能的综合平均指标 M-F1。

当然，性能的优劣和很多因素相关。首先是训练语料库中文本的质量和数量。质量是一个难以量化的指标，直观的讲，就是指定属于某类的文本的代表性。而训练文本数量多可以缓解代表性不够的影响，因为这可以减轻那些代表性不强的文本的影响度。并且大数量的训练文本可以使得后续的学习机器得到充分地训练，能够看到足够多的经验，强化典型，淡化例外。这里我们关心的是主流，而不是例外。可能在某些应用背景下，需要对例外作专门的分析，但是在本文工作的文本分类任务中，以及本文工作所描述的应用中，主要考虑样本的典型性，而忽略对于例外的分析。影响到测评性能的因素还有测试语料库和训练语料库整体的相似性，这在训练样本量不大时，表现会明显一些。另外影响性能的因素还有在文本分类的每一个步骤。比如甚至在预处理阶段，分词的性能不好也对最终分类效果造成不好的影响。特别是在当原始的词语特征空间采用词典时这种可能的不好影响程度更大。因为也许会因为分词的错误，把一个词分开，或者把属于不同的两个词的前后紧邻的两个字划分为一个词，这样就不能体现词典空间定义的

词在文本中的实际分布了。

当然，在这么多前述的影响因素中，讨论什么是最佳维数时，采用的方式是固定其他所有的因素，观察分类在测试集合上的性能，最终确定应该保留的维数。用这样的维数计算训练语料库中的文本向量，对分类器进行训练，并用这样方式下训练好的分类器对测试文本进行分类。当然，测试文本也用相同维数和维序的向量表示。表 1 和表 2 展示了采用本文提出的基于均值-方差的特征提取算法和支持向量机分类的文本分类在各个特征维下的性能指标值。

为了对比工作的效果，本文做了对比试验。用不同的特征选择方法替换模块中的均值-方差特征选择方法，重新遍历一遍上述过程。为了尽量减少偶然性，在三种方法下都作出用多种维数表示向量进行分类的结果。同时，为了排除" 均值-方差特征选择方法特别有利于支持向量机学习"的可能，作者同样换用了 Knn 学习机器重复上述过程，又得到了在 knn 学习机器下的不同的特征选择方法的分类性能对比图。也就是，为了对比特征抽取算法对最终分类性能的影响，我们分别采用三种不同的特征选择算法和两种学习机器一共六种组合的分类程序过程得到四组性能指标在不同特征降维上的分布图，为后续研究打下了基础。 参见图 12，它给出了采用方差降维，和一般的降维方法，分别在支持向量机和 knn 分类算法下，分类性能随维数变化的走势图。其中 DF 方法和 CHI 方法的曲线几乎重合，而且三种方法与采用 SVM 和 Knn 结合时，表现了大体相同的走势，因此我们只截取了在 SVM 下的情况，于是形成了图中三条曲线（其中一条对应 CHI 几乎与 DF 的重合，因此不太明显）。从中可以看出：1）本文工作提出的方法对于分类算法没有特别的偏好，至少对于典型的两种学习算法如此。2）分类性能在中部取得最好，是因为这种特征降维方法体现了除噪的特点。

表 1 RtoD 表示向量降维的比例，D 表示向量的维数，M-P、M-P、M-F1 分别表示宏准确率 宏召回率 和宏-F1 性能指标（MSV）

RtoD	D	M-P	M-R	M-F1
0.1	2826	0.8763	0.856	0.866
0.3	8478	0.9123	0.898	0.9051
0.5	14130	0.9168	0.902	0.9093
0.7	19782	0.9377	0.93	0.9338
0.9	25434	0.9378	0.93	0.9339
1	28260	0.67	0.8	0.7293

表 2 D 表示向量的维数，M-P、M-P、M-F1 分别表示宏准确率、宏召回率和宏-F1 性能指标（PFC） （见下页）

D	M-P	M-R	M-F1	D	M-P	M-R	M-F1
100	0.903	0.884	0.8934	1500	0.8498	0.83	0.8398
200	0.9126	0.894	0.9032	1600	0.8514	0.836	0.8436
300	0.916	0.9	0.9079	1700	0.8574	0.838	0.8476
400	0.9252	0.912	0.9186	1800	0.8617	0.842	0.8517
500	0.6434	0.668	0.6555	1900	0.8644	0.846	0.8551
600	0.6753	0.69	0.6826	2000	0.8654	0.846	0.8556
700	0.6624	0.6815	0.6718	2100	0.8654	0.846	0.8556
800	0.7988	0.78	0.7893	2200	0.8679	0.848	0.8578
900	0.8089	0.788	0.7983	2300	0.8693	0.848	0.8585
1000	0.8138	0.796	0.8048	2400	0.8666	0.844	0.8551
1100	0.8274	0.808	0.8176	2500	0.8722	0.852	0.862
1200	0.8335	0.812	0.8226	2600	0.8722	0.852	0.862
1300	0.8421	0.822	0.8319	2700	0.8728	0.852	0.8623
1400	0.8405	0.824	0.8322	2800	0.8763	0.856	0.866

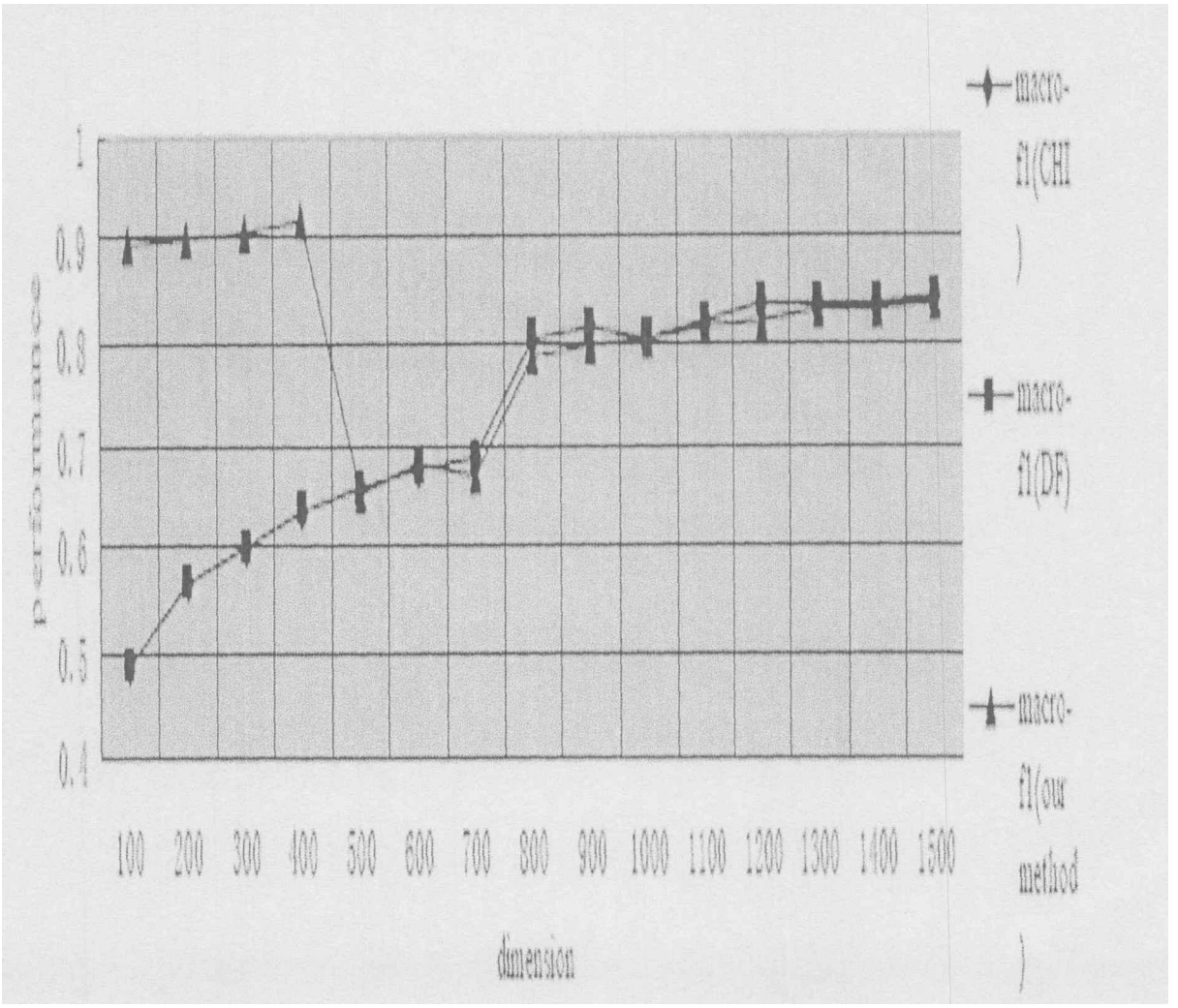


图 12 三种特征抽取算法在支持向量机上分类的性能与降维后维数的关系（Knn

上与 SVM 上几乎重合）

3.6 小结

本章主要从工程实践的角度叙述了在按照本文所述方法和框架下实现文本分类程序的若干主要设计及实现问题。并且，测试了不同方法组合下，分类的性能，展示了各自分类的结果，分析了出现这种现象的可能原因，证实了本文所述工作对于有效文本分类的价值。

第四章 文本分类的应用

4.1 在网上信息检索中的应用

对新到文本自动指出其所属的类别的文本分类本身可视为自然语言处理的一种应用。对于推动相关理论的研究具有重要意义。此外它也可以作为在更大应用框架下的一种技术和任务，作为进一步新的引申应用的基础。其中一种颇有潜力的应用就是在网上信息检索中的应用。

目前 web 页的数量海量，类型各异，并且不断动态变化和增长，为了有效的查找用户需要的信息，需要将传统的信息检索向 web 信息检索（搜索引擎）方向发展。搜索引擎面临比一般信息检索更大的困难，在于新的信息检索需求呈现下列新的特征：1）待检索的文本数量是海量的 2 ）待检索的文本是异构的 3）待检索的文本数量是动态变化的 4 ）待检索的文本可能有重复 5）待检索的文本可能是跨语言的 6 ）包含有指向其他 web 文本（资源）的链接 7 ）在网络环境下，面对的是多用户不同的检索需求和表达不严格的信息需求。面对相同的查询，可能因为用户的信息需求不同，或同一用户不同时间的需求不同，而对于同样的检索结果列表有不同的满意度 8）网络上的搜索引擎对时间有比较严格的要求，需要在大量查询的情况下对查询快速反应。在这样不同的特征和难点面前，除了考虑采用更紧凑的数据结构和高效的算法外，本文考虑在这样的应用背景下引入文本文类。

如果能够事先对文本进行分类，则面对用户简短而充满歧义的检索需求就可以在相应的类别（主题，更好的反映文本的信息功用）中进行查找，这样大大提高了检索的效果，同时大大提高了检索的效率[42, 43]。这就相当于在现有的倒文档结构中实行了分段管理，按用户可能需要的文本类别返回给用户，更有针对性，因而会提高检索的效果。比如，用户提交检索关键词“微软”，可能是希望查找其产品帮助，可能是希望查看其招聘信息，这两种信息需求反映了文本的类别信息，一类是计算机技术类，一类是招聘信息服务类。将文本类别信息加入检索的倒排文档数据结构中，可以使得返回的文章按类别排列，便于用户快速找到自己

真正需要的信息，并且便于用户进一步在检索结果中进行相关检索。

另一方面，用户的兴趣可能不在于提交一个对系统的查询，而是愿意花一些时间来浏览（browsing） web文本空间中用户感兴趣的内容[45]。或者是当其对所希望了解的领域知之甚少时，希望系统能有一个提醒。因为搜索引擎假设用户知道自己要检索什么，所以用户可以比较轻松的给出查询的关键字。然而在现实的信息利用中，还存在这样一种情况，即用户不知道查询的具体关键字，只知道大概的领域；如果有网页文本的预先分类，则可以使用户顺着这样的目录层次找下去，逐步导引用户到所需要的信息领域。这样的分类体系是用户浏览网页很好的起点。一个很好的例子是yahoo!的层次目录。与之类似还有Open Directory Project，它用在Alta Vista, Netscape,HotBot,Lycos等系统中。但是他们都还是人工进行分类，而不是机器自动进行。因此，文本分类（这里指机器的自动分类）大大降低了人力，提高了效率。不用再聘用专门的“通才型”专家一篇篇阅读进行人工分拣，而且毕竟这种方法在海量的动态变化的web 文本信息面前是相当吃力的。

4.2 在特定领域的应用

其他方面的应用，比如主动信息服务和信息过滤（Information Filtering）。邮件过滤，为保证内容安全而进行的网页过滤属于信息过滤的范畴，其中重要的基础模块就是文本分类。主动信息服务在本文的工作中被具体化到了目前一个空白而有较大使用价值的招聘信息服务中，后续有进一步的描述。过滤可以看作是特殊的二分分类问题，而主动信息服务就是要按照用户预先设定的主题（相对静态）将由 Spider 搜集到的网页中的文本进行有顺序的路由。而对文本进行路由可以看作是文本分类的另一种表述[46]。文本分类除了具有上述的应用价值，其任务的完成也推动了相关理论的发展和相关语料库资源的完善。TREC 会议中的一个基准测试任务就是过滤。这一点可以在下述文本分类的方法体系中得到证明。

文本分类和信息路由（ Routing ）过滤(Filtering)的关系是这样的。

从研究的目标性质来看，文本分类的分类体系比过滤和路由要复杂。它不局限于单类判别——“是”还是“不是”，而是多个类别，并且这些类又时还呈现父类和子类的树形结构关系。但信息过滤或路由还会考虑用户的过滤标准的变化，即相当于考虑分类体系的变化，而分类不用太关心分类体系的变化。当然，

来自统计理论的研究，也逐渐支持这种分类体系的自形成（详细参见 5.1 节）

具体应用如下。

1. 邮件过滤

邮件过滤是文本分类任务一个很直接的应用。因为除了文本来源的因素外，二者基本相似。可以认为邮件过滤是文本分类的特例。即是一个二分类的过滤问题[46,47]。

文本分类的任务是根据预先确定好的类别体系，将待分类文本分到相应的类别中去。从文本分类角度来看，垃圾邮件过滤就是要求将邮件分为垃圾、非垃圾两类中的一类，是一个二值分类问题。我们可以将电子邮件经过预处理提取出邮件正文的文本内容，利用文本分类的算法识别垃圾邮件，这也是目前垃圾邮件过滤技术研究的一个趋势。但垃圾邮件过滤与一般的文本分类在很多方面又有所区别，主要表现在：

1) 对文本分类，每个类别的内容一般不会经常改变。如，一个文本现在是体育类，将来也还属于体育类。而对垃圾邮件过滤，“垃圾邮件”类别是和用户密切相关的，更注重个性化，用户对垃圾邮件的判别准则会随时间改变，而且垃圾邮件本身的内容形式也在不断的变化。因此在垃圾邮件过滤中要给用户提供自学习、反馈的机制，适应新情况。

2) 无论对邮件服务器还是用户客户端，邮件过滤都对实时性要求比较高。因此要尽可能的采用计算简便、速度快的文本分类算法。

3) 在分类效果上，人们最不希望将非垃圾邮件误判为垃圾邮件而过滤掉，因此对垃圾邮件类别的分类准确性要求较高。

电子邮件有自身的结构特点。邮件的协议和内容格式也是由RFC（Request For Comments）的几个文档规定。从电子邮件的结构出发，寻找垃圾邮件的特征，在发件人、收件人、邮件头、邮件正文内容等各方面展开邮件过滤工作，是垃圾邮件过滤常采用的基本方法。 传统采用的方法黑白名单设定规则的方法。包括在白名单中的发件人发送的任何邮件都认为是合法邮件，黑名单中的发件人发送的任何邮件都认为是垃圾邮件。这是目前电子邮件过滤中广泛使用的技术。通常做法是收集一个黑、白名单列表，可以是电子邮件地址，也可以是邮件服务器的域名、IP 地址，收到邮件时对发件人进行实时检查。这种名单一般由比较有信

誉的组织提供，如中国互联网协会（http://www.isc.org.cn）定期在主页上公开垃圾邮件服务器 IP 地址名单。个人也可以根据需求定义和维护自己的黑、白名单。或者设置一些规则，只要符合这些规则的一条或几条，就认为是垃圾邮件。这些规则通常有：信头分析、群发过滤和关键词精确匹配等。

利用文本分类进行邮件过滤是新的基于内容过滤的方法。

2. 招聘信息的实时服务

这是本文独立提出的一种实际生活中的新应用。它是结合信息过滤和网上信息检索的前期工作——网页采集——的一种特定的应用系统。而信息过滤将以本文工作的文本分类作为核心。

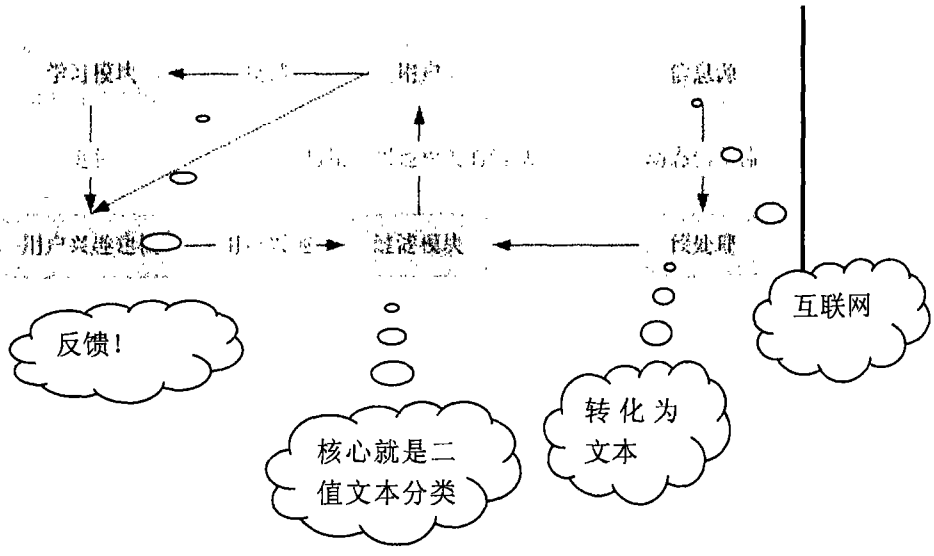


图 13 信息过滤

信息过滤系统，如图 13，包含下面四个方面的因素[49]。

一是操作的发起方式，可以有主动和被动两种。也就是由用户发起还是由过滤系统自动“奉送”。

二是操作所发生的地点。也就是选择是在信息源过滤还是在用户端过滤或者由第三方过滤服务器来代理。

三是指过滤的方法和途径。也就是采用基于内容的方式还是规则的方式。

四是得到关于用户的信息的途径。也就是是显式的方式还是隐含的方式。

显然，作为文本分类的应用，在过滤方式上采用的是基于内容的过滤。在过滤模块中与一般的文本分类不同的是，在文本分类中，分类的体系是确定的，而且常常有多个层次性的类别。但是在这里的过滤模块中，关注的类别就是用户兴

趣所指定的类别，是个单类别的问题。但是这个类别会随着用户的兴趣而发生变化，这也就产生了上文所示图中的反馈环节。这和邮件过滤应用是类似的。但是在招聘信息的实时服务这样的应用中，这个用户的兴趣在相当长的时间内不会发生太大的变化。因此更简化了反馈部分的设计。另外，对于上述的第二个方面，作为可能的商务应用可以同时考虑三种方式。如果过滤在客户端发生，那么过滤软件可以以软件产品的方式出售；如过滤在信息源或者第三方发生，则可以把过滤作为一种服务向设定不同兴趣的用户提供。另外作为招聘信息的实时服务软件，当然应该采用主动的方式，即采用类似“中断”的机制，而不是“程序查询”的机制。也就是说，当有新的符合某个用户设定的兴趣（转化为某个类别）的招聘信息产生（被程序收集到）后，即刻发送给相应的用户。可以想象，这个过滤系统的核心就是驻留在信息源或者客户机或者第三方代理上的一个“门卫分流”。它随时把收集到的信息按不同的类别进行过滤和分发[46]。

剩下的问题是如何收集网络上的招聘信息，以构造适合输入过滤系统的核心文本分类程序的文本。通常可以采用预先设定主页的方式，或者采用搜索引擎中采用的网页 spider。就简化问题而言，手工设定网页并不是很难的事情。可以把主要的招聘门户网页，BBS 设定其中。当然，为了做到尽量不遗漏，可以采用链接分析的方式不断的把新的链接加入其中。这也引起一个新的问题，即如何去重。

改进的系统还包括不但按用户兴趣（很可能是行业类别）进行过滤，在招聘信息多的情况下，可以按用户设定的排序准则进行排序。比如按薪资和工作地点排序等等。

第五章 展望、小结

5.1 分类体系的改进

中国餐馆过程一种统计随机过程。在模式识别中很有潜在的应用前景。在文本分类中也有很好的应用。目前本文工作尚停留在单层分类体系下，而采用中国餐馆过程的理论，可以把分类拓展为多层次的分类，并且分类的体系可以动态的自动形成——这类似于聚类。目前，中国餐馆问题在国内的研究极少，是很有前景的一个研究改进方向。

顺便指出，这个随机过程之所以称为中国餐馆过程缘起于中国餐桌是圆形的，暗示过程的平等均匀性。

5.2 反馈的引入

分类的效果好坏，通过在测试语料库上统计性能指标，已经可以看到，希望这个指标能够指导人们改进设计文本分类的方法，不断进步。这是由研究人员长期参与的一件事情。

考虑在用户使用文本分类器的实际过程中，对于一个文本，程序将它标记为某个类别返回给用户，如果可以收集到用户的反馈信息指明分类是否正确，将是比较好的事情。将这个文本向量连同用户认为正确的类别标号在系统的离线时间，重新加入作为新的训练语料，改进相应的学习过程成为增量学习，会不断的提高使用的性能，更成为特定用户的文本分类器，成为个性信息服务的一个可能的基础。

5.3 降维后空间大小的确定

降维后的特征空间大小常常是依据经验或者出于计算代价的考虑，由程序员初始设置一个值，按这个值对原始特征空间进行降维。之后，可能针对实用的文本分类程序，会进行试验，由人手工调整这个值，直到在测试集合上达到认为满意的测试效果。本文工作目的之一是在两个阶段分别采用各种不同的特征降维和

机器学习方法对文本进行分类，得到其在开放测试文本集合上的分类性能指标，观察并描述这个性能与特征选择保留的维数的关系，验证本文工作提出方法的优势，并作为后续研究的基础。同时，也是使用试验的方法确定最终能够保留的特征维数。因此，能否考虑在类似的研究基础上，发现降维后应该保留的特征数量同学习效果的关系，由程序自动调整确定维数，将是值得研究改进的发展方向。

5.4 效率的提高

分类的效率主要考虑训练好的学习机器对新到文本输出的响应时间，主要是按照特征文件计算新到文本特征向量的时间（也就是 indexing 的时间）和训练好的学习机器响应输出的时间。而训练好学习机器的时间是离线进行的，包括特征降维和训练学习器的时间，所以对此要求不甚严格。因此，对特征降维算法的时间性能要求不如文本预处理和 indexing 的时间性能要求高，因为后二者是会对每篇文本都进行的处理，而且对于新到文本是在线方式完成的。下面主要讨论 indexing 过程效率的提高问题。

Indexing 主要是指按照特征降维后形成的特征文件中词语序列，计算它们在文本中的出现等统计数据指标，作为文本向量每一维的值。因此查找算法的时间复杂度很重要。本文工作采用 hash 表存储保留的特征维，按其 hash 值有序，扫描一遍文中的词语，就可以得到 hash 表中各个应保留的词语特征项的出现，因此得到文本的特征向量。

有没有更为紧凑的算法和数据结构来实现这个逻辑，是有待进一步探讨的。比如用双数组的方式等等。

5.5 小结

本文主要介绍了作者硕士生期间在文本文类及其应用方面的主要工作。这项工作的理论基础和技术支持是自然语言处理技术和机器学习；完成的任务是对文本进行分类；其应用方面包括在性能和服务模式上改进搜索引擎，扩展至新的应用领域，比如邮件过滤和内容安全；以及在特定生产生活方面信息服务的应用。

文本分类分为两个大的阶段，分别采用自然语言处理和机器学习作为基础，在本文的第二章有详述。其中，重点介绍了本文提出的特征降维算法，揭示了它

的思路，描述了它的算法，分析了它的优势。文本分类的性能的评价是指导分类方法改进的重要参考，也在第二章给出了详细叙述。第三章详细描述了按第二章设计思路，工程实现文本分类的体系结构和主要过程和数据结构；并按本文工作提出的方法和其他典型方法进行文本分类，统计了其在相同测试文本集合上的指标，并做出了分析。第四章探讨了文本分类在各个方面的可能应用。第五章针对目前文本分类的理论和实现的问题进行了分析，提出了可能的改进方向；并对全文进行了小结。

本文工作研究了新的文本分类方法所依据的技术，提出了文本分类中一种新的算法，并同时用其他方法作为对比，分别统计了其分类的效果性能指标。在此基础上，探讨了文本分类新的应用方向和改进研究的方向。

参考文献

[1].李晓明,王继明. 搜索引擎——原理、技术与系统. 北京：科学出版社，2005. p1

[2].Sebastiani Fabrizio Sebastiani.Machine Learning in Automated Text Categorization. Proceedings of THAI-99, European Symposium on Telematics, Hypermedia and Artificial Intelligence. 1999.

[3].朱德熙.语法讲义.北京:商务印书馆.1982.11.

[4].俞士汶.计算语言学概论.北京:商务印书馆.

[5].JamesAllen 著,刘群等译.自然语言理解.第二版.北京：电子工业出版社.2005.1.

[6].刘群.计算语言学讲义.中国科学院研究生院.2004.www.nlp.org.cn.

[7].张春霞,郝天永.汉语自动分词研究的现状及困难.系统与仿真学报,V01.17，No.1,Jan,2005.

[8].高军.无监督的动态分词方法.北京邮电大学学报,1997,20(4):66-69.

[9].尹锋，林亚平.情报神经网络的设计与应用.情报学报，1996，(3).

[10].韩客松,王永成,陈桂林.汉语语言的无词典分词模型系统.计算机应用研究,1999,16(10):8-9.

[11].李家福,张亚非.一种基于概率模型的分词系统.系统仿真学报,2002,14(5):544-550.

[12].徐从富.隐马尔可夫模型.浙江大学人工智能研究所.2005年9月修改补充.

[13].L Rabiner.A tutorial in Hidden Markov Models and Selected Applications in Speech Recognition Proceeding of the IEEE,February 1989,77,No.2:257-285.

[14].李家福,张亚非.基于EM算法的汉语自动分词方法.情报学报2002.21(3):269-272.

[15].金凌,吴文虎,郑方,吴根清.距离加权统计语言模型及其应用.中文信息学报,2001,15(6):47-52.

[16].于江生.隐ｍａｒｋｏｖ模型及其在自然语言处理中的应用.北京大学计算语言学研究所.

[17].Shai Fine, Yoram Singer, and Naftali Tishby. The hierarchical Hidden Markov Model: Analysis and applications. Machine Learning, 32(1):41, 1998.

[18].Hua-Ping ZHANG QUN LIU Xue-Qi CHENG Hao Zhang Hong-Kui Yu Chinese Lexical Analysis Using Hierarchical Hidden Markov Model.Seconf SIGHAN workshop

affiliated with 41th ACL. Sapporo Japan, July, 2003:pp. 63-70

[19]. Torkkola. Discriminative features for text document classification. Pattern Anal Applic 18 August 2003, (2003) 6: pp. 301 - 308.

[20]. E. Montañés, J. R. Quevedo and I. Díaz. A Wrapper Approach with Support Vector Machines for Text Categorization. IWANN 2003, LNCS 2686, 2003: pp. 230-237.

[21]. 李凡，鲁明羽，陆玉昌. 关于文本特征选择新方法的研究. 清华大学学报，2001，41(7)：98-101.

[22]代六玲，黄河燕，陈肇雄. 中文文本分类中特征抽取方法的比较研究. 中文信息学报，2004，18(1)：26-32.

[23]姚天顺，朱靖波，张俐，等. 自然语言理解— — 种让机器懂得人类语言的研究. 北京：清华大学出版社，2003.

[24]. FabrzioSebastiani. Machine Learning in Automated Text categorization. Consiglio Nazionale delle Ricerche, Italy ACM Computing Surveys, Vol. 34, No. 1, March 2002: pp. 1 - 47.

[25]. Yiming Yang and Jan O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. Proceedings of 14th International Conference on Machine Learning, San Francisco, 1997:pp. 412-420.

[26]. K. Aas and L. Eikvil. Text categorization: A survey. 1999. http://www.nlp.org.cn

[27]. Dunning TE . Accurate methods orthe statistics of surpriseand coincidence. Computational Linguistics, 1993:61-74.

[28]. Dunja M ladenic1 ladenic M D, Grobelnik M. Feature selection for unbalanced class distribution and native bayes. http: llWWW. CS. cmu. edu / textlearningl

[29]. 朱褒，阮彤，于庆喜. 文本分割算法对中文信息过滤影响研究. 计算机工程与应用，2002(13)

[30]. STUART RUSSELL, PETER NORVIG 著. 姜哲 金奕江 张敏等译. 人工智能——一种现代方法. 第二版. 北京：人民邮电出版社，2004-6-1.

[31]. TOM MITCHELL 著. 曾华军 张银奎译. 机器学习. 北京：机械工业出版社，2003-1-1.

[32]. V. Vapnik 著. 张学工译. 统计学习理论的本质. 北京：清华大学出版社, 2004-6-1.

[33]. Rosenblatt. Principles of neurodynamics. NewYork:Spartam Books, 1962.

[34]. V. Vapnik, A. Chervonenkis. Uniform convergence of frequencies of occurence of

events to their probabilities. Dokl. Akad. Nauk SSSR, 181, 1968:915-918.

[35]. Vapnik V and Chervonenkis A. Theory of Pattern Recognition. Nuaka, Moscow, 1974.

[36]. Rissanen. Modelling by shortest data description. Automatioca, 14:465-471.

[37]. 王珏. 机器学习研究讲义. 2004. 7. 北京.

[38]. R. Duda and P. Hart. Pattern classification and Scene Analysis.

[39]. Richard O. Duda, Peter E. Hart and David G. Stork. Pattern classification. 2ndEdition. John Wiley & sons. 2001.

[40]. Nello Cristianini, John Shawe-Taylor 著, 李国正 王猛 曾华军译. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. 北京:电子工业出版社, 2004-3-1.

[41]. CHRISTOPHER J.C. BURGES. A Tutorial in Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery. Boston: Kluwer Academic Publishers. nus. edu. sg/~mpessk/svm/svm.

[42]. Thorsten Joachims. A Statistical Learning Model of Text Classification for Support Vector Machines. Proceedings of SIGIR-01, New Orleans, USA, 2001.

[43]. LanHuang. A Survey On Web Information Retrieval. http://citeseer. ist. psu. edu/huang00survey. html

[44]. C. van Rijsbergen. Chapter3 AUTOMATIC CLASSIFICATION, Information Retrieval. 2nd edtion. Butterworths, London, 1979.

[45]. Ricardo Baeza-Yates & Berthier Ribeiro-Neto. Modern Information Retrieval. 北京: 机械工业出版社, 2005-3.

[46]. 王斌. 信息过滤综述. http://www. nlp. org. cn.

[47]. 刘洋等. 垃圾邮件的智能分析、过滤及 Rough 集讨论. http://www. nlp. org. cn.

[48]. 潘文锋. 中国科学院硕士学位论文: 基于内容的垃圾邮件过滤研究.

[49]. Hanani, U., Shapira, B., Shoval, P. Information Filtering: Overview of Issues, Research, and systems. User Modeling and User-Adapted Interaction. 11 (2001) 203-259.

发表的论文

1. Yi Wang, Xiaojing Wang, "A New Approach to Feature Selection in Text Classification" Proceding of the Fourth International Conference on Machine Learning and Sybernatics, Guangzhou, China, Aug, 2005, IEEE Press. (EI Indexed, IEEE catalog Number: 05EX1059, ISBN: 0-7803-9091-1)

2. Yi Wang, Xiaojing Wang, " A New Formal Definition of Language for Natural Language", Proceeding of the 11th Joint International Computer Conference, Chongqing China, Nov. 2005, World Scientific Press (ISTP Indexed, IDS Number: BDF35, ISBN 981-256-532-9).

致谢

感谢我的指导老师王晓京研究员。如果说硕士阶段在研究工作方面还有些成绩的话，这和王老师的关心，指导，帮助，支持和鼓励是分不开的。王老师开放的教导方式使学生受益匪浅。从选题，到国际会议上发表论文，到讨论试验结果，项目申请、答辩，在学习上，王老师为学生默默的无私的奉献着，用心血浇灌学生成长。王老师经常加班不吃晚饭和学生讨论问题。记忆最深的是在国际会议上发表论文，英文撰写，交稿时间又很紧急，王老师从医院输液回来帮学生修改文章。此外，在生活上，王老师也很关心学生成长，经常和学生谈心，了解学生的思想和生活情况，并给予最大的支持和帮助。在硕士学位论文完成之际，满怀感激，感谢王老师的关爱，教导和为学生付出的一切!

感谢一年级导师张海盛研究员。帮助我从一名懵懂的大四学生转变为一个新的研究生。

感谢我所在实验室的何希琼研究员，钟勇研究员和冯勇研究员。谢谢他们无论是硬件还是软件上，给予我的好的环境和支持。

感谢研究生一年级的老师们，他们有的教授我知识，有的关心我成长，让我度过了非常难以忘怀的收获颇丰的一年。感谢教体系结构的董占求老师，容忍我旁听，并使我学到了很多。感谢任课老师刘群老师和王珏老师，回答我种种细节问题，他们的教授使我受益匪浅。

感谢研究生部的汤老师和吴老师，她们像朋友一样的关心，使我的研究生生活更加有意义。

感谢孟庆春师兄的无私帮助。感谢师兄师姐，同学和学弟学妹们。万武南，杨洁，粟伟、曹晟、缪海燕、田乐等等。让我感到团队的力量和温暖，和他们一起学习讨论的时光愉快而难以忘怀。

感谢朱嘉鲁，佘丽，覃安，曹科，余璟明。他们不厌其烦的和我讨论问题，给我以帮助和启发。

感谢自动推理实验室和软件室所有的师兄师姐和同学们。

感谢刘永红，赵万鹏，谭思亮，唐樨谨，张亭，李庆峰，他们乐于帮人，经常忍受我“索要资源”的麻烦。感谢曾琼，李运娣，宗芳，钟秀琴，她们经常充当我“忠实听众”，忍受

我“好为人师”的发表意见。感谢我的朋友刘萌，李军，蔡鸿鹏，范丽，于馥香，兰毅，唐家强，李晓钰，周洁，谢之旻，文强，黄恩溢，刘琰宁，魏毅，他们是我成长的养料，生活的益友。

感谢所有的同学朋友。虽然直接在学术上的帮助不多，但是和他们相处的时光快乐是开展工作必不可少的条件。

感谢我大学老师郭平教授、王铮教授，是他们最初给我展现了计算机世界的神奇，引导我逐步入门。

感谢我的父母，给予我无私的爱和鼓励。