摘　　要

全文检索技术是信息处理的各领域中的重要技术。本文对全文检索技术进行了以下几方面的研究：

1、介绍了国内外检索技术的发展过程，讨论了普通文本检索、概念信息检索、超文本信息检索、多媒体信息检索、数据挖掘等的技术特点。

2、研究了全文检索技术的两种主要索引方法的特点和实现过程。其中基于字表的检索方法由于具有无需分词、实现容易的优点，因而在实践中被广泛采用。然后针对该算法存在的“索引库较大、匹配速度不高和查全率较高而查准率较低”等缺点，引入了第二种检索方法：基于词表的检索方法。

3、研究了中文自动分词技术，这是中文全文检索钟的关键技术。对其中的几种方法，如机械匹配法（即 MM 法）、特征词库法、约束矩阵法、语法分析法和理解切分法等做了详细的比较和分析，并归纳出各自特点。其中 MM 法由于实现简单，并且是其它方法的基础，本文对其进行了着重介绍。

4、在 MM 方法的基础上，本文对一种利用基于字、词和词组的混合模型来实现中文全文检索的方法进行了探索和研究。该算法的基本原理是：把所有的单字、词还是词组都作为语词，建立汉语词语二叉树。分词时，读取二叉树右边的内容，并比较左节点的长度，得到有意义的最小长度的语词。然后又在这种算法的基础上进一步讨论了一种改进的 MM 法以减少词语的歧义切分。

5、设计了校园网内 Web 页面的搜索引擎，该引擎的主要特点是：将搜索引擎主要分为前端和后端，后端获取 Web 文档，然后分词，建立和更新索引；前端提取索引库中的内容，向客户提供检索服务。在该系统中利用网络蜘蛛，扫描校园网中所有 HTML 文档，寻找所有与检索关键字相关的页面。并将向量空间的思想运用到其中，即可提取出其中的资源中心，即检索结果。

关键词：全文检索，倒排文件，自动分词，二叉树，搜索引擎，向量空间

Abstract

The full text retrieval (FTR) is the primal technology of disposing the information. The article does some research of the full text retrieval technology.

1、The article summarize the development of the web search technology in the domestic country and aboard. It will refer to not only the common document retrieval in the web, but also the query of concept information, hypertext information, multimedia information and the data mining. These new technology are also introduced briefly. The article lists the specification of the full text retrieval technology, at the same time the deficiencies are also referred and the trends of the future are demonstrated .

2、The paper demonstrates the two index methods of the FTR. Search based on the words list is very simple in the implementation of the algorithm without dividing the words and it is used widely. Because of considerable storage space and larger index database, higher rate in the full searching and the lower rate in the exact searching, the article demonstrates a new retrieval method based on the phrase list.

3、Chinese Words Divided Syncopation Technology is the difficulty of the query technique based on phrase. Some divided syncopation such as mechanical matching method, feature phrase library method, restriction matrix method, syntax analysis method and comprehended syncopation method are emphasized. The MM method is easy to realize and the foundation of other methods, and is introduced emphatically.

4、The article purpose the hybrid modeling based on character, word and phrase as the Chinese FTR using MM method. To reduce de divergent divided syncopation an improved MM method is prompted.

5、The retrieval system adopting the algorithm could search for World wide web pages in school . The search engines could be classified front searching engines and meta searching engines: the meta one get Web document, then slice the word,

establish and update index; the front one extract the content of the index library, provide the users query service. It uses network spider to scanning all HTML documents and find out the pages which is useful. Then it uses the idea of Vector Space Model（VSM） to pick up the result.

Keywords: Full Text Retrieval, Inverted Files, Divided Syncopation, Search Engines, Vector Space Mod

第一章　序言

20 世纪 90 年代，信息技术突飞猛进，Internet/Intranet 席卷全球，产生了大量的文本、声音、图像、数据库等各种形式的电子信息资源。随着大容量的存储介质技术与馆藏信息数字化的发展，各种形式的电子资源经过收集、加工，就可以通过网络提供远程的存取，实现资源的管理与共享。面对越来越多的信息，迫切需要一个高效的检索系统，以便对这些信息进行整理和加工。

1.1 信息检索的发展过程

纵观计算机信息检索系统的发展，可以将其发展过程划分为三个阶段。

第一阶段：1971 年以前建立的许多信息检索系统，其工作方式是传统的批处理检索方式。这一阶段的数据存取与数据通信能力都比较差。

第二阶段：1971 年以后，产生并发展了联机情报检索系统，如 OCLC、Dialog 在线数据库联机检索系统。这一阶段的特点是联机数据库集中管理，具有完备的数据库联机检索功能，但其数据通信能力较差。

第三阶段：以 Internet 的出现为标志，系统大多采用分布式的网络化管理，其信息资源的主要特点是:数字形式表达、多媒体和多载体、内容覆盖全社会领域、分布无序、难于规范化和结构化、内容特征抽取复杂、用户界面要求高等。这些特点导致了信息处理从传统模式向新型模式的转变，如体系结构从终端主机方式到客户/服务器结构方式，网络环境从局域网到 Internet 等开放网，应用接口从封闭界面到 WWW 等，信息结构从结构化到非结构化，系统功能从单纯信息检索到综合信息管理和服务等等。这些变化必将促使信息检索技术的研究和不断发展，以满足人们对提高信息利用能力的需要。

全文检索是信息检索发展的最前沿和目前的最高阶段。

1.2 全文检索技术的发展

全文检索(Full-Text Retrieval)是指以全文本信息为主要检索对象，允许用户以自然语言根据资料内容而不是外在特征来实现检索的先进查询手段。“文海捞针”是对全文检索的形象描述，全面、准确和快速是衡量全文检索系统的关键指标。全文检索技术的出现，导致了信息检索领域的一场革命；比起传统

的标引检索来，全文检索技术提供了全新的、强大的检索功能，是发现信息、分析和过滤信息、信息代理、信息安全控制等应用的主要技术基础。以全文检索为核心技术的搜索引擎已经成为网络时代的主流技术之一。在全文检索研究领域中，基于概念、超文本信息检索最为活跃，并已取得了突破性进展。

1.2.1 基于概念的信息检索技术

基于概念的信息检索是指通过对文献中的原文信息进行语义上的自然语言处理，析取各种概念信息，并由此形成一个知识库。然后，根据对用户提问的理解，检索知识库中的相关信息，以提供直接的回答。

概念信息检索有以下几个特性：

1、具有分析和理解自然语言的能力。可以对输入的原文根据其概念内容进行组织和安排，以析取相关的概念信息和范畴知识。然后，通过记忆机制将它们存储到知识库中，以备检索用。

2、记忆机制能够自动补充与更新。

3、具有用自然语言回答用户提问的能力。

概念信息检索技术的上述特性，使系统的查全率和查准率都得到提高。Web 上的 Excite 搜索引擎就是采用概念信息检索理论设计的数据库，在 Excite 搜索引擎输入检索词“elderly people financial concerns”，系统可将含有“economic status of retired people”和“the financial concerns of senior citizens”等与检索词概念一致的信息作为返回结果，可见系统自动将“elderly people”与“retired people”和“senior citizens”、“financial concerns”与“economic status”进行了概念匹配。由于基于概念的信息检索技术具备了智能检索的一些特性，其系统分析和理解原文内容及用户提问信息的能力较强，因此，备受检索用户的青睐。

1.2.2 超文本信息检索技术

超文本信息检索技术是以超文本网络为基础的文献检索技术。超文本信息组织的特点是正文信息以节点而不是以字符串作为信息的基本单元，节点间通过链进行连接。在检索文献时，其检索技术应能满足节点间的多种链接关系可以动态地选择性激发，根据思维联想或新信息的需要，通过链从一个节点到另一个节点。Internet 上的搜索引擎代表了超文本信息检索技术的发展水平，网

上建立和运行的多个基于超文本信息的全文检索系统如：Al-tavista、Yahoo!、Lycos、Infoseek 等著名引擎，不仅检索速度快，还普遍实现了自动分类、自动摘要、自动索引等功能，使 Web 信息得到有效的组织，极大地方便了用户对 Internet 信息的查找和利用。

1.2.3 基于内容的多媒体检索技术的发展

多媒体信息检索是指对图形、图像、文本、声音、动画等多媒体信息进行检索的过程。目前，一种被称为基于图像内容检索(Content Based Image Retrieval，CBIR)的多媒体检索技术正在成为国际上众多公司、大学和研究机构的研究热点。CBIR 技术是随着大量多媒体信息的出现而产生，是解决多媒体信息检索的有效途径。传统的数据库检索是采用基于关键词的检索方式，早期的图像数据库如 Kodak Picture Exchange System(KPX)、the Press Link Library 和 the Time Archive Collection 沿袭了这种检索方式，采用描述性文本进行检索。由于图像和视频信息的内容具有丰富的内涵，在许多情况下仅用几个关键词难以充分描述,而且作为关键词的图像特征的选取也有很大的主观性。因此，这种传统检索技术有很大的局限性。于是，基于内容检索技术应运而生。它区别于传统的检索手段，融合了图像理解技术，从而可以提供一种从巨容的图像/视频库中，根据人们提出的要求进行有效检索的方法。根据所处理的对象，CBIR 可分为静止图像检索和视频检索两种。

与传统的检索方式相比较，CBIR 具有以下特点：

1、利用反映图像/视频内容的特征来进行检索；

2、是相似度检索，即根据库中各个被检索单元(图像或镜头)与检索要求的相似性程度而返回检索结果；

3、除了利用反应图像/视频内容的特征来进行特征检索外，还提供了多种其它检索手段，如可通过提供样本图像进行相似性检索，也可通过人机交互进行浏览检索等。

在现有的系统中，IBM 的 QBIC(Query By Image Content)系统可以说是第一个真正的功能齐全的 CBIR 系统，它对 CBIR 技术的发展也产生了深远的影响。QBIC 系统提供了对静止图像和视频信号的检索手段。在静止图像检索中，提供了颜色、纹理、草图、形状、多物体等多种检索方法，并提供了根据样本

图像进行相似性检索的方法。在视频检索中，包括了分镜头检测、主运动估计、建立层描述、通过拼接完成代表帧生成等多种视频处理手段，并在此基础上提供了通过物体运动、摄像机运动的附加视频检索手段。

由加州大学圣地亚哥分校开发的 Virage 系统在美国市场上目前是最畅销的 CBIR 系统。Virage 系统提供了将多种检索特征相融合的手段，用户可以定义各检索特征在检索中的权重，从而可根据自己的需要控制检索方向。Virage 系统还提供了浏览检索手段——系统首先从图像库中随机选取一组图像，供用户从中选择与检索要求接近的图像，若这些随机图像中没有满足要求的，用户可要求系统重新选取，直到图像组中有与检索要求相近者。

基于内容的多媒体信息检索技术有着广阔的应用前景，它可广泛用于电子会议、远程教学、远程医疗、电子图书馆、军事指挥系统等方面，大容量图像数据库的检索是其主要应用方向。作为一种新兴的技术，CBIR 目前还处于初级阶段，只能利用一些相对简单的特征来检索，但随着研究的不断深入和发展，其功能也会越来越强大，将成为未来信息社会中不可缺少的技术和工具。

1.2.4 数据挖掘技术的发展

数据挖掘(Data Mining)就是从大量的、不完全的、模糊的、随机的数据中，提取隐含在其中的、人们事先不知道的、但又是潜在有用的信息和知识的过程。数据挖掘是近年来在信息检索技术基础上发展起来的一门技术，是信息检索技术的一个重要分支。还有很多和这一术语相近似的术语，如从数据库中发现知识(KDD)、数据分析、数据融合(Data fusion)等。特别要指出的是，数据挖掘技术不仅是面向特定的数据库的简单检索查询调用，而且要对这些数据进行微观、中观乃至宏观的统计、分析、综合和推理，企图发现事件间的相互关联，以指导实际问题的求解,甚至利用已有的数据对未来的活动进行预测。数据挖掘是一门广义的交叉学科，它汇集了数据库、人工智能、数理统计、可视化、并行计算等多方面的技术。在信息网络化时代，单个的人利用传统的手段几乎不可能处理或阅读整个信息库。同时鉴于信息库中存在着大量无用和冗余的信息，往往使用户所寻找的信息量与信息总量相比非常小，因此如何“去粗取精、由表及里”并迅速、准确以及适量地提供用户所需信息，同时在一定程度上揭示信息与信息之间的关联是文本挖掘的主要任务。

数据挖掘与传统信息检索的区别和联系：

1、传统的信息检索较适合于数据类型同构的信息库。但是对于异构数据组成的信息库，例如多媒体等则不擅长。

2、传统的信息检索需要用户将要寻找的事件以关键词的形式较准确的描述出来，作为查询提交给系统。但往往这与人们通常的思维行为模式背道而驰，再有用户经常并不真地知道要什么信息。

3、由于字义本身与其概念的延伸不在同一级上，造成利用传统信息检索所寻找的信息可能仅仅是字面本身的信息，但往往人们想要的是这个信息的概念及相关的成分，而不仅仅是字面所表达的信息。

4、传统信息检索可以被当作挖掘的底层工具，换言之，传统信息检索关注“词”的处理而信息挖掘则关心“词”的本原(Ontology)。

5、传统信息检索尽管引入布尔运算，作为逻辑算子使用户能够较准确地表达查询。但其结果往往导致或丢失一些信息或产生大量冗余信息。

6、传统信息检索通常是用户从信息库中去找他想要的。而信息挖掘是看信息库中到底存在些什么。目前，信息挖掘的研究和开发以及应用还刚刚起步，但已显示出较好的发展前景，随着信息挖掘的应用与技术的成熟，必将成为信息内容服务业的主流，以支持一个快速的、新兴的 Internet 信息服务市场。

1.3 全文检索的特点

1、方便灵活的全文存储和管理功能。对库的各种操作简便灵活，易于掌握，可对库中的记录进行编辑、修改、裁减、打印、编排。树型多级分类管理可使系统建库数量不限，数据容量可以无限大。

2、丰富全面的检索方式。包括支持布尔检索(与、或、非、异或)、位置逻辑检索(同段、同句、相差几个字以及前后次序有关等)几十种检索方式。全文检索系统是一种存储文献全文或其主要部分并能提供全文检索的源数据库，与书目数据库、事实数据库相比较，主要检索特点是：

（1）包含信息的原始性。库中信息基本上是未经信息加工的原始文本，具有客观性。

（2）信息检索的彻底性。可对文中任何字、词、句进行检索，还可表示

检索词间的复杂位置关系。

（3）所用检索语言的自然性。不作标引，借助截词、邻接等匹配方法，以自然语言检索所需文献。

（4）数据相对的稳定性。全文数据库数据基本上是封闭的，一般不需更新。

（5）检索结果的准全性。利用后控制表及检索技术可以改善检索效果。

3、系统支持 GB 国标码、GBK 大字符集码、BIG5 繁体字码和多语种处理能力。支持中、日、西、俄及其他东方文字或者图像输入和存储。

4、支持多种格式文档资料和各种多媒体信息的管理和检索。包括各种格式化的文档(WPS、TXT、CCED、WORDSTAR)以及 HTML 页面的超文本链接，自动索引格式化的文档和页面，书版排版格式(S2、PS2、S72)直接入库，实现全文检索功能,流行的图像格式(GIF、BMP、TIFF、JPG、PCX)和声音动画文件(WAV、MID、FLC)的存储和播放，Microsoft Office 文件的语音识别、合成，图形和图像处理与传送以及超文本的链接处理技术，图像、图形、视频和音频信息的存储、管理、检索和播放以及各种文字处理软件、图表制作软件产生的格式化文件的存储、管理、检索和输出。

5、采用数据和索引压缩技术,以提高系统的查询效率，降低空间的膨胀率。

6、支持结构化数据和非结构化数据的存储，同时实现数字型、日期型、定长和变长字符型、文本型、文档型(如 MSWord，HTWL 等)和多媒体数据型。

7、系统广泛的兼容性。支持多种硬件平台，如小型机、服务器、PC 机。目前流行的硬件平台有：IBM、SUN、DEC、SGI、Unisys、NCR、Alpha、VAX 等。支持多种操作系统，如服务器上运行的 Unix、Scounix、Windows NT；客户机上运行的 Windows3.X、Windows95、Windows NT、Web 浏览器。中文全文检索系统应能支持以上软、硬件平台中的绝大部分，以保证用户在 Internet 应用方面具有优势，同时使信息服务系统的水平升级和垂直升级简便易行。

8、采用 Client/Server 体系结构，可使系统具有良好的可伸缩性和可选择性，在实际多用户环境中可以获得更高的性能，适合于以网络为中心的计算模式和 Internet 应用。

1.4 全文检索所面临的问题

虽然全文检索技术日趋成熟，文献型的检索系统的开发和使用也相当广泛，一些记录达千万级的大型数据库已经使用多时，在索引结构、检索技术、查询性能、词查（Thesaurus）管理、自动标引、自动摘要和自然语言处理等相关领域均取得了显著进展，但现在信息检索的研究和开发工作也面临着许多挑战：

--无所不在的信息检索。无所不在的信息检索要求把信息检索技术扩展到单面、光盘出版、企业信息库、Web 站点、Internet 搜索引擎、电子商务和数据仓库等各个领域。

--自然语言处理技术。无论从数据挖掘，还是提供更易使用的自然语言查询接口方面，中文自然语言处理是关键因素，但是中文自动标引在 80 年代比较热烈的研究没有取得可用的突破性成果；自动摘要和自动分类系统的可用性仍缺乏实际证明；机器翻译系统仍然是仁者见仁，智者见智。

--检索系统的评价。和其他领域一样，信息检索技术的研究和系统开发需要科学的评价，我国 863 计划已经开始对中文 OCR、自动分词、自动摘要进行统一测试评测，建立检索系统的评测也十分必要。

--多媒体内容检索。我国信息检索的研究主要是针对“数据库记录”和“文字”。对图像、音频和视频信息的基于内容的检索研究需要大大增强。在某些数字图书馆软件系统中已经实现内容图像检索，针对音频和视频信息的检索在国外也取得了很多成果。

--Internet 搜索引擎。全文检索技术是类似于 Altavista 等搜索引擎的核心支撑技术，由于 Web 是以 HTML 作为置标语言，因此相关排序等算法肯定和普通文本的检索不同，同时因为网上信息太多、信息不可能被完全覆盖，对检索的要求也首先是查准，然后是查全，除了文字页面的搜索引擎外，图像、音频、视频信息的搜索引擎也在发展中。

1.5 全文检索的方法

目前所研究的全文检索方法主要有两种：基于统计的方法和基于知识的方法。基于统计的方法是利用查询变量在目标对象中的各统计指标来描述它们之

间的相关度；基于知识的方法要求引入知识库的信息用以分析查询变量，从而检索出具有一定匹配度的信息。基于统计的方法在信息检索中的应用相当普遍。从简单的文本搜索到信息挖掘都能发现它的踪影。为了优化检索结果，部分研究引入遗传、神经网络等算法。实际上，基于知识的方法是在基于统计方法的基础上发展起来的，较为典型的研究为基于内容的检索。尤其在计算机图像和视频等领域，基于内容的检索吸引了大批研究者，其目的是提取对象的特征，并附以识别特征的知识库结构。

1.6 全文结构

全文整体结构如下：第一章介绍了特点和主要方法，指出了当前全文检索所面临的问题。第二章描述了全文检索技术中的主要方法之一—字索引。第三章分析、比较了几种中文自动分词方法的特点和区别。第四章利用前面所述自动分词方法实现基于词索引表的全文检索，并提出一种改进的机械匹配法以提高检索效率。第五章是全文的重点，在该章中利用全文检索技术构造了一个适用于校园网内的搜索引擎。第六章对全文作了小结。

第二章　基于字表的检索方法

汉字全文检索系统和西文全文检索系统相比，在原理和方法上都有相同之处。首先在计算机内部，无论汉字还是西文都是以字节形式存储。两种技术的差别主要是由于汉语本身造成的。与西方文字和文本比较，汉字文本中的词是由一个或多个单字构成。词与词之间无间隔，实词和虚词之间也无间隔，检索的基本单元可以是单个汉字，也可以是词。所以，存在两种基本的检索方法：基于字表的检索方法和基于词表的检索方法。下面，我们来讨论基于字表的检索方法。

2.1. 字表检索系统基本设计

2.1.1. 字表的组织

字表法索引库的主要部分是每个字的字表信息，字表结构如表 2.1 所示，其中字符 i 对应的字表记录了该字符在源文档中所出现的位置 Pix。位置可以采用字符相对于文档头的偏移字符数表示，而不按通常情况采用相对于文档头的偏移字节数，这样可以大大减小位置的数值大小，有利于进一步采用压缩技术。建立字表索引时，需要扫描整个源文档，对出现的每一个有效字符，计算其在文档中出现的位置，并将该位置的值加入到对应的字表中。

...	.....
啊	$P_{11}P_{12}P_{13}$
阿	$P_{21}P_{22}P_{23}$
...	.....
的	$P_{i1}P_{i2}P_{i3}$
...	.....
中	$P_{j1}P_{j2}P_{j3}$
...	.....

表 2.1 字表结构

2.1.2 检索策略

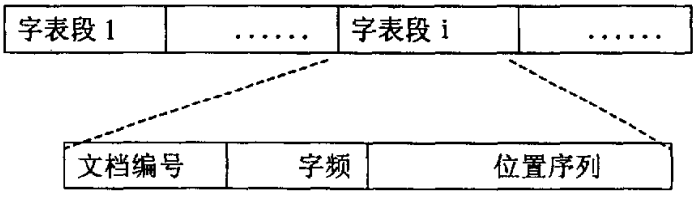
索引库中的一个字表记录了对应字符在源文档中的所有位置信息。考察一个字符串，如两个字的字符串其中 XY（X、Y 表示任意的汉字字符），假设 X 的位置为 P_x，如果字符串在源文档中出现，Y 则的位置 P_y 必定等于 P_x+1（1 为两个汉字间的字符距离）。在索引库中，X 的字表中将包含 P_x，而 Y 的字表中也必然包含 P_x+1。进行检索时，扫描 X 和 Y 各自对应的字表，若文档中有该词出现，则必定有 X 对应的字表中存在位置值 Px，Y 对应的字表中存在位

置值 Py，使得 $P_y = P_x + 1$ 成立，每查到一对这样的位置值，就是检索到字串 XY 一次。扫描完两字的字表，就可检索出该字符串的所有出现。

2.1.3 索引库结构

字表是索引库中最主要的部分，在每个汉字字符对应的字表中，包含该字符出现在所有文档中的全部位置。为了区分每个位置值属于哪个文档，每个字符的字表被分为多个字表段，每段对应一个文档，记录该字符在此文档中的出现位置。字表采用倒排文件结构，如表 2.2 所示。

表 2.2 字表及字表段结构



每个字表段起始部分记录当前文档的编号，随后是该字符在文档中的出现频率，最后是该字符在文档中的所有出现位置序列。每个字符的所有字表段按文档编号递增的顺序排列，如果该字符在文档 k 中没有出现，则不存在文档 k 对应的字表段。

2.2 索引创建及其优化

2.1.2.1 基本的索引创建方法

系统采用的索引创建方法不需要排序，分为如下两步。第一步分析源文档，产生临时的中间文件，我们称为分析过程。当前只处理 GB 码字符，其中包含全部字符，既有汉字，又有一般的数字，标点符号等。GB 码第一个字节的范围是 0XA1~0XF7，第二个字节的范围是 0XA1~0XFE。汉字从“啊”开始，首字节为 176~247，第二个字节为 161~254。根据这种分布规律，可以方便地定位每个字符对应的字表信息。源文档经过处理，其包含的每个字符的对应信息写到一个临时的中间文件。对于每个字符，其在临时文件中的对应信息包括：该字所出现的当前文档编号，在该文档中的出现频率，出现的位置序列和该字符出现在下一个文档中的数据的指针数据在文件中的偏移值。第二步处理临时

文件，依次从临时文件中读取每个字符出现在每一篇文章中的数据信息，生成最终的倒排文件，在这里称为创建过程。生成的最终倒排文件中包含每个字符出现在所有文档中的信息。包含：该字符出现的当前文档的编号，出现频率和相应的位置序列。处理过程如图 2.1 所示：

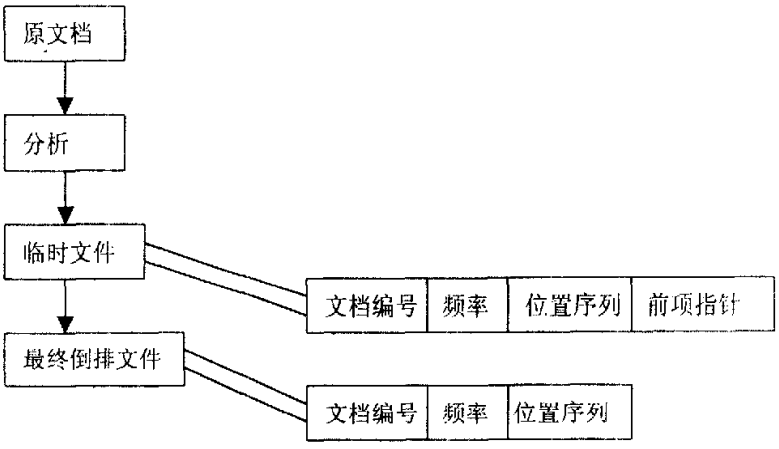


图 2.1 索引创建流程

2.2.2 改进后的索引创建方法

在上述方法中，对于源文件的分析过程本身需要一定的时间，随着处理数据集规模的增大，相应的分析时间增大，但第二步创建过程所需的时间也迅速增大。该过程需要大量的随机读取操作来遍历每个字符对应的所有信息。当数据的规模增大时，遍历每个字符的临时数据的操作变得很慢。这是由于字符对应的每个字表的数据在临时文件中有一定距离，遍历需要不断地移动文件指针来读取这些数据。

利用操作系统提供的虚拟内存技术可以优化索引的创建过程。Windows 操作系统用虚拟内存来动态管理运行时的交换文件。为了提供比实际物理内存还多的内存容量以供使用，Windows 操作系统占用了硬盘上的一部分空间作为虚拟内存。当 CPU 有要求时，首先会读取内存中的资料。当内存容量不够用时，Windows 就会将需要暂时储存的数据写入硬盘。内存映射文件技术是 Windows

NT 提供的一种新的文件数据存取机制。利用内存映射文件技术，系统可以在2GB 的地址空间中为文件保留一部分空间，并将文件映射到这块保留空间。一旦文件被映射之后，Windows NT 将仔细管理页映射、缓冲以及高速缓冲等任务。通过把临时文件映射到虚拟内存中，可以大大加快对临时文件的访问速度。

对于较小的源数据集，分析处理后生成的临时文件也较小，使用内存映射文件可以大大加快创建过程。但当数据规模增大时，该方法的性能迅速降低，甚至比没有使用内存映射文件都差。性能的降低一方面由于机器有限的内存，其小于临时文件的大小。另外一方面，同一个字符相邻的数据在临时文件中距离过大，导致大量的缺页中断，系统性能大大降低。解决该问题的有效方法是把原有的单个的大的中间文件分成多个小的临时文件，在分析过程中生成多个小的临时文件，创建过程依次处理每个临时文件，将其映射到虚拟内存中，可以充分利用直接内存访问的速度，并且减少缺页中断。

2.3 索引的压缩技术

2.3.1 索引的压缩与查询分析

由于全文检索系统通常处理的都是海量数据，经过处理生成的索引数据也是很大的，因此采用一定的压缩策略，可以节约存储空间。另外，全文检索系统执行检索时，通常需要读取一定的索引数据，采用压缩技术，可以减少读取数据所需的时间，从而有可能提高检索速度。在研究索引 I/O 压缩技术时，一方面希望能够减少索引数据占用的磁盘空间，但同时不能降低检索速度，否则系统的性能就会下降。

假设 Tr 为检索时读取未压缩索引数据所需的 I/O 时间， Tc 为内存中实际匹配和查找时间，则针对未压缩索引数据执行检索所需的总时间 T 可以表示为：

$$T = Tr + Tc \quad (1)$$

对压缩索引数据执行检索所需的总时间 T'可以表示为：

$$T' = Tr' + Td + Tc \quad (2)$$

其中，Tr'为检索时读取压缩索引数据所需的 I/O 时间

Td 为解压缩时间。

在检索中，一般情况下读入的部分压缩数据需要解压缩，然后进行匹配和查找，最坏的情况下，所有读入的压缩数据都需要解压，因此会使用更多的时间。合理的压缩技术应该保证检索压缩索引数据时读取索引数据的时间和对索引数据解压缩的时间总和不大于检索未压缩数据时读取索引数据所需的时间，即：

$$T' \leqslant T \quad (3)$$

$$Tr' + Td \leqslant Tr \quad (4)$$

2.3.2 压缩方法

在未压缩的索引文件中，文档编号占用 4B，字频占用 2B，每个位置值占用 2B。在字表中源文档的编号是按递增的顺序排列的，可以采用运行长度编码的方法表示文档编号。对于任何文档编号，只记录其相对于前一个文档编号的偏移值。同样，某个字符在一篇文章中的所有出现位置必然是按升序排列的，也可以采用这种方法进行编码，对每一出现位置记录其相对于前一个出现位置的相对偏移值。采用差值编码，可以减小数值的范围，以便于对这些差值进一步采用短的长度表示。

采用字节对齐的方法压缩索引，对于一个给定的正整数，可以用 1 个或多个字节表示，表示该数的首字节的最左边两位为标志位，指示该数值共占用几个字节，剩余位可以用来存储实际的数，即 1~4B 可以分别用 6，14，22，30b 来保存实际的数。经过压缩，每个文档编号就不必一定要占用 4B。比如，文档编号为 80，用该方法表示为二进制位串：01000000 01010000，只需要 2B。对于字频和位置值，一般较小，不会大于 32768，所以可以采用一位做标志位，指示该数占用 1B 或 2B。

2.4 小结

本章在分析实际需求和现有技术的基础上，研究了字表法全文检索中索引的创建优化及压缩技术：使用虚拟内存技术可以大大提高索引的创建时间，索引的压缩技术可以减少索引文件所占用的磁盘空间，也可以提高检索的速度，但解压也有一定的代价，检索速度因此降低。

第三章 汉语自动分词方法

相对于单字索引，词表法适用于大规模应用，索引库可以组织得比较小，检索速度比较快，而且还可以实现同义词、反义词的概念检索，但其难点在于中文自动分词及分词中歧义的处理。下面我们通过介绍几种主要的分词方法，并讨论在分词中如何进行歧义处理。

3.1 机械匹配法

机械匹配法的基本思想是：事先建立一词库，其中包含所有可能出现的词。对给定的待分词的汉字串 S，按照某种确定的原则切取 S 的子串，若该子串与词库中的某词条相匹配，则该子串是词，继续分割剩余的部分，直到剩余部分为空；否则，该子串不是词，转上重新切取 S 的子串进行匹配。

机械匹配法的数据结构较为简单。一般来说，词库可分为基本词库和专业词库。为了提高查找匹配效率，它们又可细分为单字词库、双字词库、三字词库、四字词库和多字词库等。对机械匹配法，每个词库中的词条都非常简单，只需记录词的内部表示，而不必附带其他信息。词库可根据内部表示的大小组织成一个有序表。这样便于用二分法进行匹配查找。但是，由于整个词库一般较大，无法一次调入内存，结果，一次匹配查找往往需要多次访问外存，执行速度不一定十分理想。对此，一种改进的方法是：按照某种确定的原则(如词的首字)将整个词库分成若干个子词库，使每个词库均可一次调入内存，而每个子词库均按内部表示的大小组织成一个有序表。这样，执行一次匹配查找时，首先确定待查串可能属于哪个子词库，然后把那个子词库调入内存按二分法进行查找。如果相同子词库中词的内部表示长度不一，那么，标准二分法还必须加以修改。总之，词库的设计应以既省空间又能快速执行匹配查找为目标。

词库的建立是机械匹配法成败之关键。这里一个重要的问题是：到底哪些词该收入词库？哪些不应收入词库？词库小了也许不够用，词库大了既费空间又费查找时间，甚至造成大量的歧义切分。一般来说，词库的好坏可通过两个参数来衡量，即覆盖率和利用率。覆盖率是指词库中出现在待切分语料中的词的数量与待切分语料的实际含词量之比，而利用率是指词库中出现在待切分语

料中的词的数量与词库含词量之比。这两个参数都依赖于词库和待切分的语料，并且两者相互制约。为了获得高的覆盖率和高的利用率，一般采用基本词库加专业词库的做法，其中，基本词库中收集那些与语料无关的常用词汇而专业词库则根据语料所属专业来选取。即使这样，也不能保证词库中确实含有特定语料中的所有词。为了对付这种情况，自动分词系统应该为用户提供动态维护(包括扩充)词库的功能。

3.1.1 最大匹配法和最小匹配法

根据字串切取的策略，机械匹配法分为最大匹配法和最小匹配法。根据匹配不成功时重新切取的策略，机械匹配法又分为增字法和减字法。增字法一般与最小匹配法相结合，减字法一般与最大匹配法相结合。

最大匹配法的基本思想是：假设词表中最长的词由 i 个字组成，则每次从句子头上截取一个长度为 i 的字串，令它同词表中的词条依次匹配，如果词表中的却有这样的一个 i 字词，匹配成功，就把这个字串作为一个词从句子头上切分出去。然后再从句子余下的头上截取另一个 i 字字串，重复上述过程，直到句子被切分完为止。如果在词表中找不到一个词条能与当前字串匹配，就从该字串的串尾减去一个字，用 i－1 字长的字串到词表中去查找，若匹配成功同样把该字串作为一个词切分出去；若匹配失败，从该字串串尾再减去一个字，用 i－2 字长的字串去匹配词表，直到匹配成功。

最小匹配法的方法和最大匹配法相反：它是按词表中最短长度的 j 个字(一般为 j＝1）从句子头开始截取字串与词表中的词条进行匹配。若匹配成功，就把这个字串作为一个词从句子头上切分出去。然后再从句子余下的头上截取另一个 j 字字串，重复上述过程，直到句子被切分完为止。若匹配失败，则将字串串尾加一个字，得到 j＋1 字长的字串，与词表中的词条匹配，若匹配成功同样把该字串作为一个词切分出去；若匹配失败，则继续在串尾加一个字，用 j+2 字串与词表匹配，直到匹配成功。

例 1，输入句子“中华人民共和国成立了”。假设词表中有“中华人民共和国”，“成立”，“了”，“中华”，“人民”，“共和国”，“中”，“华”，“人”，“民”，“共和”，“国”，“共”，“和”等词，词表中最长的词是 7 个字（i＝7），最短的词是 1 个字（j＝1）。

若使用最大匹配法：第一次从句首截取的7字字串“中华人民共和国”就匹配成功。句子余下的部分为3字字串“成立了”，词表中没有这样的3字词，字串截尾的新串“成立”，匹配成功。句子余下部分“了”，也匹配成功。于是句子被切分为“中华人民共和国/成立/了”。

若使用最小匹配法：第一次从句首截取“中”，匹配成功；接着截取“华”，匹配成功，继续切分句子余下的部分，最终句子被切分为“中/华/人/民/共/和/国/成立/了”。

例2，输入句子“有个人叫张梦云”。运用最大匹配法得到的结果为“有/个人/叫/张梦云”，最小匹配法的结果为“有/个/人/叫/张梦云”。

可见，最小匹配法的原则是“短词优先”，即认为对于同一个句子来说，切分的词最短时是最佳切分结果；而最大匹配法的原则是“长词优先”，即认为对于同一个句子来说，切分的词数最少时是最佳切分结果。由于大多数汉字均可构成单字词，所以按最小匹配法分词的结果往往因分得太细而不合要求（如例1）。反之，虽然最大匹配法的评估原则在大多数情况下是合理的，但当长词覆盖短词时，也会引起切分错误（如例2）。在实际运用中，我们使用最大匹配法较多。

3.1.2 正向匹配法和逆向匹配法

根据切取子串的方向，机械匹配法又分为正向匹配法和逆向匹配法。前面所讲的是正向匹配法。如果从句子的尾部开始从右向左扫描，便是逆向匹配法。若匹配失败，就要从当前字串中去掉最前头的一个字以形成新串（逆向最大匹配法）；或在当前字串前面加上一个字以形成新串（逆向最小匹配法）进行匹配。

例1，输入句子“中国以新的姿态出现在世界东方”。

正向最大匹配法：中国/以/新/的/姿态/出现/在世/界/东方 （误）

逆向最大匹配法：中国/以/新/的/姿态/出/现在/世界/东方 （误）

例2，输入句子“他使节约粮食进一步形成风气”

正向最大匹配法：他/使节/约/粮食/进一步/形成/风气 （误）

逆向最大匹配法：他/使/节约/粮食/进一步/形成/风气 （正）

例3，输入句子“这件事反应了一个人的精神面貌”

正向最大匹配法：这/件/事/反应/了/一/个人/的/精神/面貌 （误）

逆向最大匹配法：这/件/事/反应/了/一/个人/的/精神/面貌 （误）

例 4，输入句子“美国加州大学的科学家发现……”

正向最大匹配法：美国/加州/大学/的/科学家/发现…… （正）

逆向最大匹配法：美国/加州/大学/的/科学家/发现…… （正）

由以上四例可知，利用正向最大匹配法和逆向最大匹配法切分同一文本，有四种不同情况：情况 1，两种方法切分结果不同，且两种结果均不正确（如例 1）；情况 2，两种方法切分结果不同，但其中有一种结果正确；情况 3，两种方法切分结果相同，但结果都错误；情况 4，两中方法切分结果相同，且是正确的。可见为便于发现歧义切分，可将两者有机地结合起来形成双向匹配法。由于正向匹配法和逆向匹配法对词库的组织有不同的要求，所以，将它们结合时，要重新考虑词库的组织以便两者都能快速执行。

3.2 特征词库法

特征词库法的基本思想是：事先建立一个特征词库，其中包含各种具有切分特征的词。对给定的待分词的汉字串 S，首先根据特征词库将 S 分割成若干个较短的子串，然后对每个子串分别采用机械匹配法进行切分。实际上这是一种“分而治之”的办法。由于每个子串都比 S 短，所以切分速度较快。

特征词库法的理论基础是：虽然汉语的形态标志没有英语等西方语言那样丰富，但是汉语中还是存在一些形态标志的。并且，这些形态标志为汉语的切分提供了重要的依据，所以在自动切分时，应尽可能加以利用。一般来说，各种词缀(包括前缀和后缀)、虚词和重叠词等都可作为切分特征。虽然它们的数量有限，但由于它们的使用频率一般较高，所以把它们单独分离出来先行处理是可行的、有效的。这样做也便于系统的维护。

因为不同类型的特征词往往要求不同的处理，所以特征词库中的词条不但要记录词的内部表示，而且还要记录它的类型。特征词库的规模一般不大，往往可一次调入内存，并且可按使用频率的大小来排列。切分时按频率由大到小的顺序依次处理。

选取特征词的依据是汉语语法中的构词法和构形法等。但是，也应该看到，

汉语中经常出现一些不合常规的例外现象。对此，在建立特征词库时应尽可能全面地加以考虑，最好能预计各种例外情形以便特殊处理。

由于特征词库中的每个词条往往是对若干词的抽象，所以这些词的切分得到了统一的处理。结果，机械匹配法中的词库就不必包含这些词，达到了既省空间又加快查找速度的效果。

上面给出的两种分词方法的一个共同的特点就是孤立地考虑词的形式。然而，出现在汉语中的每个词除了具有形式之外，还具有词性和词义。此外，相邻词汇的词性和词义必须是相容的，否则就会出现不合语法或不合逻辑。换句话说，汉语中相邻词汇的词性和词义之间必须满足一定的约束关系，这些约束关系是判断自动切分结果正确与否的重要依据，必须设法体现在分词方法中。下面将要给出的方法就是沿这条思路对前面给出的方法的改进。

3.3 约束矩阵法

为了说明引入约束矩阵法的背景，下面先解释一下什么叫做歧义切分。所谓歧义切分是指相同的汉字串被切分成不同的词的序列。典型的歧义切分包括交集型歧义切分和组合型歧义切分。

所谓交集型歧义切分是指形为 ABC 的汉字串既可切分成 AB/C，又可切分成 A/BC。如汉字串“计算机房”既可切分成“计算机/房”又可切分成“计算/机房”。所谓组合型歧义切分是指形为 AB 的汉字串既可切分成 AB，又可切分成 A/B。如汉字串“任何”既可切分成“任何”又可切分成“任/何”。

现在的问题是，当某个汉字串具有歧义切分时，怎样在这些切分结果中挑选一个正确的结果作为最终可用的切分结果。约束矩阵法就是为了解决这个问题而提出来的。

约束矩阵法的基本思想是事先建立一个语法约束矩阵和一个语义约束矩阵,其中的元素分别表明具有某词性的词与具有另一词性的词的相邻是否符合语法以及属于某语义类的词与属于另一语义类的词的相邻是否符合逻辑。另外，事先还要建立一个词库，其中包含所有可能出现的词，它们的各种可能的词性和语义类。对给定的待分词的汉字串 S，按照某种确定的原则切取 S 的子串，若该子串与词库中的某词条相匹配，则从词库中取出该词的所有词性和语

义类，然后根据约束矩阵判断这些词性和语义类中是否存在与已切分出来的相邻词相容的部分。若有，则该子串是词，记下它的所有相容的词性和语义类作为后继切分的基础,继续分割剩余的部分，直到剩余部分为空；否则，该子串不是词，转上重新切取 S 的子串进行匹配。

假设不同的词性共有 N 种，不同的语义类共有 M 种，那么语法约束矩阵可实现成 N*N 的布尔矩阵 G，G(I，J)=TRUE 表示具有词性 I 的词后直接紧跟具有词性 J 的词是合语法的。同样，语义约束矩阵可实现成 M*M 的布尔矩阵 S，S(I，J)=TRUE 表示具有语义类 I 的词后直接紧跟具有语义类 J 的词是合逻辑的。

约束矩阵法的前提和基础是存在对词的性质的分类和对词的语义的分类。但是，词的分类问题，特别是词的语义的分类问题并不那么简单。按什么原则分？该分多细？都是值得研究的问题。并且，为了不同的目的而进行的分类往往结果不一。当然，在这里，分类的目的是为了便于形成对切分有效的约束矩阵。在确定了词的分类后，约束矩阵的形成既可以现有的语法和语义规则为基础，又可以对大语料库的分析结果为基础。

在约束矩阵法中，词库中的词条不但要记录词的内部表示，而且还要记录它的各种可能的词性和语义类(为便于动态维护，这些信息可用链表结构来实现)。由于兼类现象在汉语中十分普遍，并且大多数词都有许多不同的义项，所以每个词条所需空间将比前面给出的两种方法大得多。但是，约束矩阵法的一个优点是，分词的结果不但把连续的汉字串分割成了词的序列，而且它还给出其中每个词的词性和语义类。如果分词系统是另一个更大的汉语处理系统的子系统，那么，那个更大的汉语处理系统便可充分利用这些信息进行语法和语义分析。另一方面，确实应该看到，由于约束矩阵法只利用了相邻词汇的约束关系，而汉语中大量存在跨词汇的约束关系，所以它的作用是十分有限的。当分词系统是某个包含语法分析系统的更大的系统的子系统时，一种理想的做法是：把语法分析系统和分词系统溶为一体。这就是下面将要给出的语法分析法。

3.4 语法分析法

引入语法分析法的背景与约束矩阵法相同，它们的不同之处仅在于前者通

过语法规则给出全局约束，而后者仅通过约束矩阵给出局部约束。

为了说明语法分析法的作用，下面考察一下几个汉语句子的切分问题。对汉语句子“他在计算机房基建投资”。按机械匹配法，它既可切分成“他/在/计算机/房/基建/投资”，又可切分成“他/在/计算/机房/基建/投资”。到底该选哪个作为切分结果，机械匹配法无法确定。但是，只要对它们进行语法分析，就不难发现前者不合汉语语法，后者符合汉语语法。所以应以后者作为切分结果。相反，汉语句子“他在计算机房调试程序”应切分成“他/在/计算机/房/调试/程序”。因此，相同的汉字串“计算机房”在不同的语言环境中可有不同的切分，对特定的语言环境到底采用哪种切分可借助语法分析来确定。同理，“何时何地任何职”应切分成“何/时/何/地/任/何/职”，而“任何人都应遵法守纪”应切分成“任何/人/都/应/遵法/守纪”。事实证明：借助语法分析来提高切分正确率是完全可能的。

语法分析法的基本思想是，事先建立一套汉语语法规则，其中的规则不但给出某成份的结构(即它由哪些子成份构成)，而且还给出它的子成份之间必须满足的约束条件。另外，事先还要建立一个词库，其中包含所有可能出现的词和它们的各种可能的词类。对给定的待分词的汉语句子 S，按照某种确定的原则切取 S 的子串，若该子串与词库中的某词条相匹配，则从词库中取出该词的所有词类，然后根据语法规则进行语法分析(包括语法分析树的构造和约束条件的检查，这时不但要使用该词的所有词类，而且还要使用前面已分析部分的结果)。若分析正确，则该子串是词，记下语法分析的结果作为后继切分的基础，继续分割剩余的部分，直到剩余部分为空；否则，该子串不是词，转上重新切取 S 的子串进行匹配。

这里首先需要确定语法规则的内部表示。为了加快分析速度，一般将整个语法规则库分成若干个子库(如根据规则右部的第一个分量或最后一个分量来分类)。每个子库中的规则又可按使用频度来排序。一条规则实际上就是一个产生式加上一个关于该产生式右部各分量的约束条件。约束条件可实现成布尔函数。

语法规则的形成是自然语言形式的结果，是用计算机分析和处理自然语言的前提和基础。历史上，正是为了实现自然语言的形式化而建立了形式语言理

论。另一方面，在用形式语言理论来描述和处理自然语言的过程中所遇到的各种问题又不断地促使新理论的提出和完善。这种理论研究与实际应用的循环不但使理论得到进化，而且也使人们对自然语言的认识不断深化。但是，到目前为止，这一循环还远没有达到饱和状态。也就是说，理论结果和实际需要之间还有很大的距离。具体体现在：为描述和处理自然语言而提出的形式语法规则还不能完全覆盖丰富多彩的自然语言现象。结果，语法分析法的应用将不可避免有其局限性。理想的系统应为语法规则的增删和修改提供手段。

另外，语法分析法要求保存分析时产生的所有中间结果(语法分析树)，故它的空间开销要大些。不过，由于分词的最终结果包括一棵语法分析树，所以后继处理中就不必再进行语法分析了。

3.5 理解切分法

经验告诉我们，人们在阅读汉语文章时，并不存在孤立的切分阶段，而是一边切分，一边理解。面对一段含有歧义切分的汉字串，人们总是采用一种可理解的切分而放弃不可理解的切分。一般来说，切分是理解的基础，理解是切分之目的，可理解性是判断切分正确与否的标准。切分与理解的这种相互依赖关系决定了要想提高切分结果的正确率,就必须在切分法中引进“理解”成份。理解切分法就是这样一种具有“理解”成份的切分法。它与语法分析法的关系是，后者是前者的基础。但是，除了进行语法分析外，它还要进行语义分析。

理解切分法的基本思想是：

1、事先建立一个词库，其中包含所有可能出现的词和它们的各种语义信息

2、对给定的待分词的汉语句子 S，按照某种确定的原则（例如按正向最大匹配法）切取 S 的子串

3、若该子串与词库中的某词条相匹配，则从词库中取出该词的所有语义信息；若不匹配，则转 1，按某种原则重新切区子串（例如减去子串尾的最后一个子），并继续匹配词库，直到匹配成功

4、调用语义分析程序进行语义分析(包括形成理解结果和检查约束条件，这时不但要使用该词的所有语义信息，而且还要使用前面已分析部分的理解结

果)。

5、若分析正确，则该子串是词，记下理解结果作为后继切分的基础，继续分割剩余部分，直到剩余部分为空；否则，该子串不是词，转3重新切取S的子串进行匹配。

这里涉及到理解结果的内部表示的问题。常见的表示方法有基于格语法的语义框架法、语义网络法、概念结构法、功能描述法等。理解结果的形成由对应的语义分析程序来负责。词库中需记录哪些语义信息以及它们的表示形式，这些问题都根据语义分析程序的需要来确定。由于理解切分法的最终结果包括理解结果的内部表示，所以，它为后继的处理提供了一个很高的起点。

但是，也应该指出，为了有效地实现理解切分法，还有许多理论问题需要研究。并且，即使采用理解切分法也不能解决所有的歧义切分问题。例如，汉语句子“乒乓球拍卖完了”既可切分成“乒乓球/拍/卖/完/了”又可切分成“乒乓球/拍卖/完/了”。并且两者都是可理解的。这时，除非具有进一步的语用和语境知识，否则即使人也无法判断该采用哪一种切分。

3.6 小结

自动分词的几种方法：机械匹配法、特征词库法、约束矩阵法、语法分析法、理解切分法各有其特点。机械匹配法是这几种分词方法中的基础。为了解决机械匹配法中的分词歧义的问题，最简单的办法是同时采用正向最大匹配法和逆向最大匹配法进行双向切分。但由于切分字串和匹配词表的次数为一般方法的两倍，速度较慢，需要过多内存，所以一般不予采用，而且，双向切分只能找出大部分的歧义，而不能在两种切分结果不同时选择出正确的切分结果，也无法完全实现分词无歧义。

特征词库法主要是通过将待分字串分解，建立特征词库，以实现字串的快速切分；约束矩阵法是在分词的同时分析已切分的字串与前后字词的逻辑关系，如果符合约束条件就是词，否则便不是，来实现无歧义切分；语法分析发是通过对已分字串与其所在句子的语法关系是否合理，来实现词语的正确切分；理解切分法在分词的同时需要进行文章的语义分析。这几种方法不光实现起来较复杂，而且其分词的准确性很大程度上取决于所采用的约束条件的充分

性和语法、语义规则的完整性，所以就实现的可行性来说，机械匹配法中的最大匹配法（MM 法）不失为一种简单、有效的分词算法。如果能对 MM 法进行适当的改进，以提高分词的准确性，则它将在中文全文检索领域得到更广泛的应用。在下面一章里，我们将在最大匹配法的基础上，提出一种改进方法，以实现尽可能无歧义的检索。

第四章 基于词表的检索方法

4.1 预处理算法

4.1.1 建立词库

语词：在这里我们把字、词、词组统称为语词。词包括单字词、两字词、多字词。

思想根源：借用了汉语中词组本位的思想和汉语树库的构建。

L：为词库

T_i：为词库中基于每个汉语单字的字、词、词组的语词树

N_j：为每个语词树中的语词节点

$L=\{T_1, T_2, T_3\cdots\cdots\cdots\cdots\cdots\}$

$T_i=\{N_1, N_2, N_3\cdots\cdots\cdots\cdots\cdots\}$

如图 4.1：

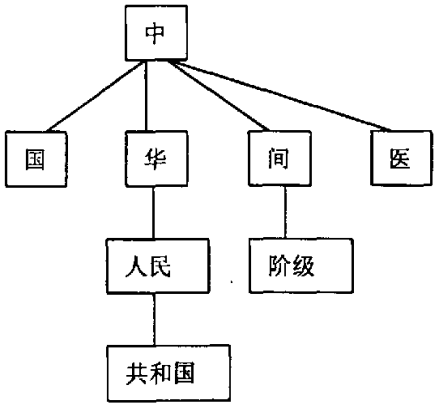


图 4.1 一般树结构语词树

考虑到二叉树存储结构的优点是：结构简单，可以方便的构造任意的二叉树，可以方便的实现二叉树操作中的查询、插入、和删除操作。我们把每个单字可能构成的各种词语普通树(图 4.1)用二叉树(图 4.2)的形式表示出来，虽然可能增加树的深度，但却降低了算法的复杂度，减少了搜索时间。

由于根节点是汉语单字，所以在词的扩展中不需要再增加整棵的树，只需要增加部分树枝即可，只要计算机的容量足够大，词库的更新不再是问题。

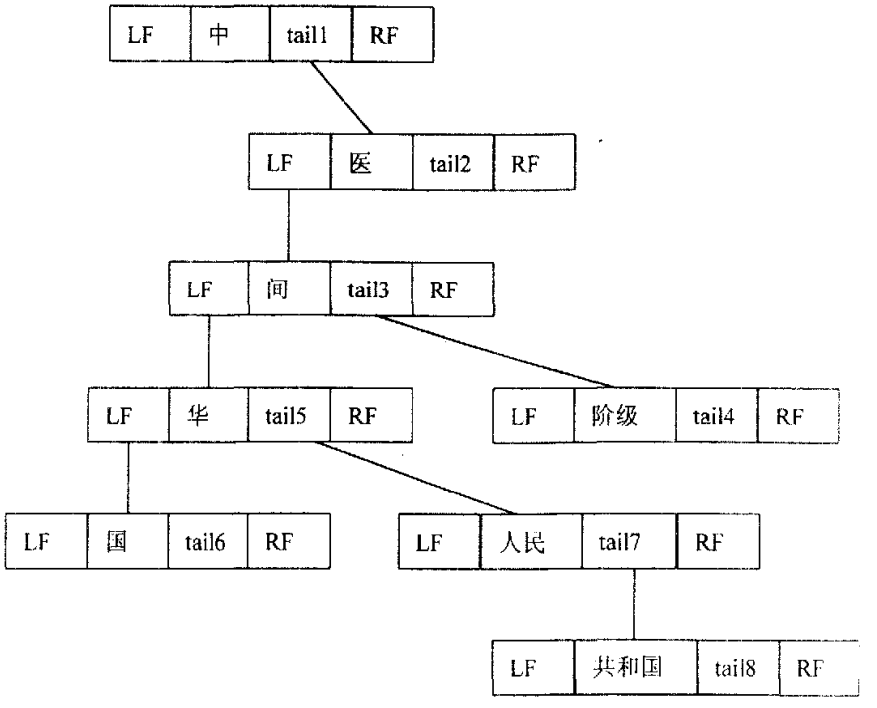


图 4.2　二叉树结构语词树

每个节点及其左子树是一个语词层级，其排列顺序是按照文本比较的降序排列的，在同一语词层级内，我们可以采用二分法快速定位；每个节点的右子树对应此节点的下一个语词层级，从而在分词时能够层层递进找到匹配的词。

每棵树的根节点即词首汉字的地址排放按照 $h(cc)=(91*c1)\wedge c_2$ 的地址排放(c_1，c_2)为汉字 cc 的高、低字节的机器内码值，∧为异或运算符。

词典的建立采用 c 语言，二叉树每个节点的数据结构可以表示如下：

```
struct Treeinfo
{
string Data;
int   tail;
```

Treeinfo*Left;

Treeinfo*Right;

}

结构中，Data 表示一个词干，如：‘阶级’；tail 表示对应词组分词文档库所在的地址，如 tail4 表示‘中间阶级’这个词组对应的分词文档库的地址；Left 指向对应节点的左子树；Right 指向对应节点的右子树。之所以增加地址 tail 项，是为了提高文档分词结束后存入分词文档库的速度，也是为了提高查询的速度。

4.1.2 建立停用词库

在汉语中有一些虚词如“啊”、“的”、“了”等词，并无具体的含义，在检索中意义不大，不会用作检索关键词，我们称作停用词。这些停用词虽然个数很少，但出现频率很高，在索引中占用较大的存储空间，而在检索时却很少用到。所以如果建立停用词库，不对停用词建立索引，将提高检索效率。

4.1.3 算法

分词时，从句子流中读入一个汉字，然后根据根节点地址 h(ri)找到对应的单个汉字，因为根节点的左指针为空，所以直接由其右指针指向的节点数据 datastr1 的长度 Ncl 从句子流中读入 Ncl 个汉字 filestr1，如果 filestr1 比 datastr1 小，则继续取左指针对应语词 datastr2 长度的 filset2 与 datastr2 比较；如果 filestr1 与 datastr1 相等，则停止对左子树的比较，取右指针对应语词 datastr2 长度的 filestr2 与 datastr2 比较；如果 filestr1 比 datastr1 大，则表明这个单汉字对应树的比较从 datastr1 的父节点结束。以此类推到分词结束。算法描述如下。

设句子流中某个句子 $Fi=a_1a_2\ldots a_ja_{j+1}\ldots a_n$，设当前分词指针 p 指向汉字 a_j

（!）据 $h(a_j)=(91*c_{10})\wedge c_{20}$($c_{10}$，$c_{20}$为汉字 a_j 的高、低字节的机器内码值)找到单汉字 a_j 对应的语词树。

（2）存储节点对应的数据 $datastr_{k-1}$，扫描节点左指针指向的节点数据 $datastr_k$，计算 Ncli=Len($datastr_k$)，把 Fi 中 $filestr_k=a_ja_{j+1}\ldots{}_ja_j$+Ncl 读入进行分析

（3）如果 $datastr_k$=NULL 则转（5）

如果 strcomp($filestr_k$，$datastr_{k.1}$)=1 则转(5)

如果 strcomp($filestr_k$，$datastr_{k-1}$)=0 则 k = k +1 转(4)

如果 strcomp($filestr_k$，$datastr_{k-1}$)=-1 则转(2)

（4）存储节点对应的数据 $datastr_{k-1}$，扫描节点右指针指向的节点数据 $datastr_k$，计算 Ncli=Len($datastr_k$)，把 Fi 中 $filestr_k=a_ja_{j+1}...a_j$+Ncl 读入进行分析，转（3）

（5）把新分出的词到 $datastr_{k-1}$ 结束加上分词符，顺序放入已分词文档数据库，此数据库包括了位置字段，等待后续处理。

4.1.4 待分字串的预处理

在机械分词过程中，把句子中的字串和词表中的词条进行匹配是必须进行且不断重复的步骤。我们注意到，如果词表是按增序排列的话，具有同样前缀的词相隔不远。由此想到，我们可以在开始分词前进行一个预处理过程，按某一算法对句子进行扫描，保存下字段的内部结构信息而不仅仅是对某一长度的字串进行匹配。通过选择合适的算法，在预处理过程中就能完成分词过程中所有的数据库访问操作。在实际分词的时候可以直接用到这些结果。这样做的好处是：因为它保存了所有的切分可能，所以预处理过程可以不加改变地用于各种机械分词算法，便于算法的实现；而且利用它也可以找出所有可能的歧义字段。

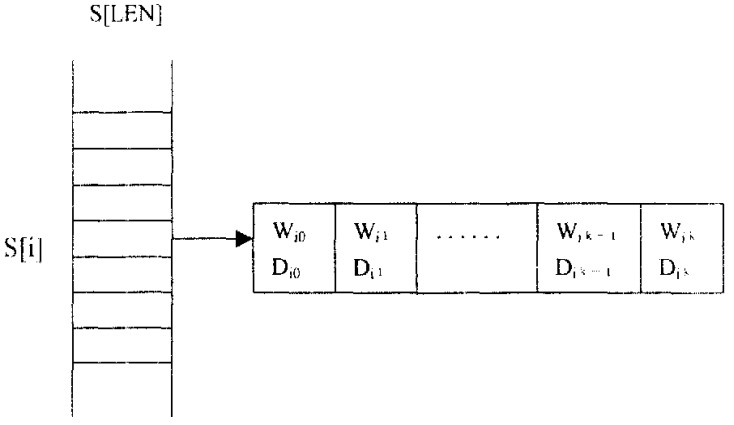


图 4.3 用来保存字段结构信息的数据结构

在预处理过程中，需要一种符合要求的数据结构来保存字段的内部结构信息。我们选择的数据结构形式如图 4.3 所示：

其中　数组 S[LEN]的内容是待分析的句子，S[i]为句子中的第 i+1 个字；

C_i 指向首字为 S[i]的字段的内部结构信息；

W_{ik} 表示 S[i]到 S[i+W_{ik}]组成一个分词单位；

D_{i0} 是 Wik 所表示的分词单位的属性，如它在词库中的位置，词性等。例如，如果句子中有字串“中华人民共和国”，通过增字扫描可以得到词典中首字为“中”的分词单位有“中”、“中华”、“中华人民”、“中华人民共和国”，除去首字的字串长分别为 0、1、3、6，在内存中的表示如图 4.4 所示：

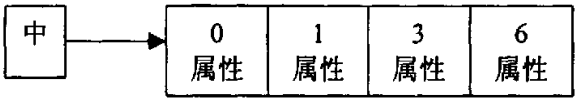


图 4.4 首字为“中”的字段的内部结构信息

有了这些信息,在实际分词时就很容易从“中”开始，在句子中截取一定长度的字串来得到符合条件的词条。

利用以上的数据结构，我们可以实现预处理过程，其算法大致描述如下：

```
Pretreatment                    //函数中常量 LEN 为句子长度，常量 MAXWORDLEN 为词表
                                的最大词长
{
  for（i=0；i<LEN；i++）
  {
    k=0postion=Index（S[i]）；//检索首字索引，结果存入变量 postion 中
    for（j=0；j<MAXWORDLEN；j++）
    {
      if（SearchWord（postion,S[i],S[i+j]））//判断 S[i]到 S[i+j]的字串是否为词。
                                          //另外，当 i+j>=LEN 即超出句子尾
                                          //部时还应进行相应的调整
        {
```

```
                Wik=j;
                k++;
                StoreData（Dik）;                    //保存该词条的属性
            }
        }
    }
}
```

4.2 索引的构成

索引库由两级构成：第一级为词语级索引，是图 4.2 中右子树的“RF”部分；第二级为文档级，是该词语在某一文档中出现的位置。其具体结构如图 4.5：

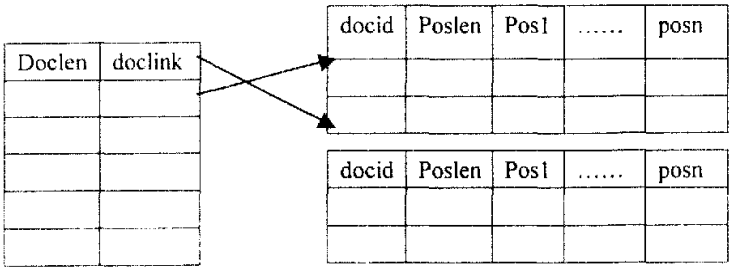


图 4.5 中文索引库结构

其中：doclen 为该词语对应索引表 2 的长度；

doclink 为该汉字对应索引表 2 的地址指针；

docid 为该词语所出现文档的序号；

poslen 为词语出现的频率；

pos1...posn 为词语在文档中出现的位置

检索的时候，不需要将所有索引表预先调入内存，可以采取一种请调策略。一般只词语级索引表 1 调入内存，而索引表 2 依旧在辅存；根据索引表 1 中对应索引表 2 的位置信息调入索引表 2。这样，虽然整个索引库较大，但只占用较少的主存空间，检索速度较高。为了进一步降低索引库的大小，提高检索速度，还可以对索引表 2 进行适当的压缩：

1、posn 为字符在文档中出现的位置，一般用相对于文档头的字节数表示，而这里用字符数表示。这样，每个位置信息所需要的存储空间比原先要减少大约一半。

2、位置信息不再用相对于文档头的偏移字符数表示，而是用该位置相对于该字在本文档中前一出现位置的偏移字符数表示。因为，若 pos1，pos2……posn 表示相对于文档头的位置信息，则 pos1……posn 必然是以升序排列。为了节省空间，pos1 存放该汉字相对于文档头的偏移字符数，pos2 存放该汉字相对于 pos1 的偏移字符数，即 posn 为该字相对 posn-1 的位置。

4.3 改进 MM 算法的基本思路

前面描述了最大匹配法(Maximum Matching Method，简称 MM 法)的基本过程。MM 法的评价原则是“长词优先”，然而现有的最大匹配法，不论顺向还是逆向，增字或减字，都是在局部范围进行最大匹配，即每次最大匹配的范围都是最先 i 个或最后 i 个字符。这样并没有充分体现“长词优先”的原则。例如以下句子：

句子 1：“当中华人民共和国成立的时候”

句子 2：“当他看到小孩子时”

如果用正向的 MM 法进行分词，第二个句子的结果是：“当/他/看到/小孩子/时”，切分是正确的。但第一个句子的结果却是：“当中/华人/民/共和国/成立/的/时候”。显然，“当中华人民共和国”是歧义字段，这里的切分是错误的。

如果用反向的 MM 法即 RMM 法进行分词，第一个句子的结果是：“当/中华人民共和国/成立/的/时候”，切分是正确的。但第二个句子的结果是：“当/他/看到/小孩/子时”，“小孩子时”又成为了歧义字段。

可以看到，以上两种分词方法都在一定情况下产生了歧义切分。这里歧义产生的原因是没有充分体现“长词优先”的原则。“中华人民共和国”和“小孩子”都是句子里最长的词，但是在某些情况下被切分开来。为了对这种情况进行改进，我们提出一种改进的 MM 法，其基本思想如下：

假设词表中最长的词由 i 个字组成，句子长度为 N，为了便于讨论，假设是采取归左原则进行切分。

先从句子第1个字开始截取一个长度为i的字串(即句子的开头i个字),令它同词表中的词条依次匹配。如果在词表中找不到一个词条能同当前字串匹配，就从句子第2个字开始截取一个长度为i的字串重复以上过程。如果还找不到，则依次从第3，4，……N－i个字开始截取长度为i的字串进行匹配。如果在某一次匹配中查到词表中确有这样一个i字词，匹配成功，就把这个字串作为一个词从句子中切分出去，把原句中位于这个字串左右两边的部分视为两个新的句子，递归调用这一过程。如果所有的匹配都不成功，说明句子中没有长度为i的词，则开始寻找长度i－1的词。重复这个过程直到整个句子被切分。

例如对刚才的句子1："当中华人民共和国成立的时候"，设i=7，首先截取的字串是"当中华人民共和"，匹配不成功，接着截取字段"中华人民共和国"，匹配成功，把它切分出来。原来的句子变成两个子句："当"和"成立的时候"。对它们再分别进行以上过程，直到所有的词被切分出来为止。

同样，对于句子2："当他看到小孩子时"，首先被匹配成功的字串显然是这个句子中最长的分词单位"小孩子"。可以看到，本算法在整个句子的范围内寻找最长词，充分体现了"长词优先"的原则，成功地处理了MM法和RMM法不能正确切分的句子。

本算法的大致描述如下：

```
Seqment（S[0]，S[LEN-1]）//对S[0]到S[LEN-1]的字串即全句进行分词常量LEN
                          为句子长度，常量MAXWORDLEN为词表的最大词长
{for（j=MAXWORDLEN-1；j>=0；i--）
  {for（i=0；i<LEN；i++）
    {if（Match（S[i]，j））          //判断是否有首字为S[i]，长为j+1的字段即搜索wik
                                    中是否有值为j
      {WordSegment（S[i]）；        /把找到的首字为S[i]，长为j+1的字段切分出来
        If（i>0）                   //判断S[i]是否为句子的首字
        Segment（S[0]，S[i-1]）；   //对S[0]到S[i-1]的字串进行分词
        If（i+j<LEN－1）            //判断S[i+j]是否为句子的尾字
       Segment（S[i+j+1]，S[LEN-1]）；//对S[i+j+1]到S[LEN-1]的字串进行分词
        Break；                     //找到了首字为S[i]的词条,所以中止对原句的扫描
```

```
        }
      }
    }
  }
```

以上算法通过对函数 Segment（ ）的递归引用最终完成分词。为了便于讨论，在这里采取的是归左原则（对连续性交集，需左部结合），实际实现的时候我们采取的是归右原则，这样分词精度更高。

4.4 检索的实现

4.4.1 练习部分

1）搜集数据，建立全文数据库

2）筛除停用词，建立汉语词库。其中词库由二叉树构成，二叉树的根结点是常用汉字，根结点和右子树构成以该汉字开头的常用词语

3）为每一个词语建立文档级索引表

4.4.2 检索部分

（1）读入检索字串，并调用 Pretreatment 函数，在内存中生成如图 4.3 的每一个汉字的字段结构信息，并以最大匹配法切分字串

（2）指针指向检索字串中第一个汉字，根据字段结构信息调入对应的二叉树

（3）遍历二叉树

（4）若匹配成功，则根据节点中 RF 信息调入对应文档级索引表，转 7）

（5）若匹配失败，指针在检索字串中右移一位，重新以最大匹配法切分字串，调入对应二叉树，重复 3），4）

（6）若依旧失败，将用于匹配的字串数减 1，重复 2），3），4），5）

（7）根据索引表中文档信息将文档序号加入检索管理链表，并保存文档中的摘要

（8）将剩余字串按以上方法继续匹配，并把匹配得到的文档序号加入检索管理链表；

（9）按检索要求对链表中的文档进行“与”或“或”操作

（10）若链表为“空”，则输出“无相同匹配”；若链表不为“空”，则输出检索结果

4.5 字索引与词索引的比较

从前面的介绍可以看出，字表法和词表法两种检索方法各有特色，适用于不同的领域。现将它们的优缺点比较如下：

1、**查全率** 在字索引检索中，由于是将检索字串逐字匹配文档中的语料，所以可以将所有匹配成功的文档检索出来，查全率较高。而在词表法中，检索字串匹配汉语树之前必须进行分词，自动分词不可避免的会产生分词歧义，则会影响到匹配的正确性，所以查全率较低。

2、**查准率** 由于自索引方法是将检索字串机械匹配文档库，而不理解字串的涵义，势必将影响到检索的查准率。例如，输入检索字串“中华”，可能将含有诸如“……其中华人……”、“……当中，华人……”的文档检索出来。在词表法中，匹配之前的自动分词使得每一个匹配的词语具有独立的涵义，不会出现上述检索错误，所以具有较高的查准率。

3、**检索复杂度** 由于分词的原因，词索引法明显复杂于字索引法。

4、**索引库的大小及检索速度** 以单个汉字建立的索引库中包含了所有常用汉字，占用的索引空间较大，检索所需要的时间较长；以二叉树建立的汉语词库中的索引表，所占存储空间小，检索速度快。

通过比较可见，字索引比较适用于数据量较小，对查准率要求不高，但在查全率上要求严格的系统；词索引适用于具有较高的查准率、海量数据的大型系统中。

4.6 本章小结

MM 法是词索引检索中主要用到的一种分词方法。本章以 MM 法为基础实现以词为单位的全文检索。在下一章中将以本章中提到的全文检索算法为基础构造搜索引擎。

第五章 利用全文检索技术实现校园网内的搜索

随着 Internet 的飞速发展，各种网上应用不断发展。目前网上中文信息的数量增长迅猛，使得基于全文检索技术的中文搜索引擎越来越多。而针对校园网的检索系统却相对缺乏。校园网面对的是学校中的学生和老师用户，其主要作用是方便用户进行资料的查询，或是学术讨论，或是友谊联系等等。相对来说，在校园网中，数据量和数据的类型没有 Internet 上的丰富，对查准率的要求高于查全率，所以我们使用第四章提出的基于最大匹配法的检索方法。

5.1 校园网搜索引擎的产生环境

5.1.1 将 Internet 网上的搜索引擎用于校园网的弊端

近十年来，中文信息在 Internet 网上的迅速发展，产生了大量的中文搜索引擎，比较著名的有：中文雅虎、搜狐、百度搜索、天网搜索、google 搜索等。这些搜索引擎主要是针对 Internet 网上海量、无序的信息而设计的；对于校园内的局域网来说，信息量远不如 Internet 上丰富和复杂，如果将以上搜索引擎用于校园网内的信息检索，无疑是“杀鸡用牛刀”，而且会严重影响到检索质量。其原因如下：

1、搜索引擎需要的存储空间过大。以上所谈到的检索系统检索的范围是整个互联网，所以其中的文档数据库将相当大，而且不管是字索引还是词索引其索引库所占的存储空间都将很大。

2、搜索引擎算法复杂。例如，检索时需要遍历 Internet，搜索引擎必须通过某种算法选择一个页面作为初始 URL 开始访问；在校园网中则无需进行页面选择，可以直接将学校的主页作为初始页面。再如，更新 Internet 上的文档数据库时也是需要利用某种算法更新，并且更新后的数据库中的页面不可能覆盖整个 Internet 上的中文网页；而校园网的 IP 地址在一个有限的范围内，我们可以每隔一定的时间间隔（如一天、一星期或一个月）对文档数据库中的所有页面进行更新，检索时访问到校园网内的所有页面。

3、检索的速度慢。文档数据库和索引库的大量数据必然降低检索速度，复杂的搜索算法也将花费更多的时间。

5.1.2 构造校园网搜索引擎的可行性

基于以上的原因，我们认为单独为某一个校园网构造搜索引擎是有一定实际意义的。同时不论从理论上还是经济上来说，为某一个特定的网络建立搜索引擎也是切实可行的：

1、理论上来说，校园网的搜索引擎比互联网上的搜索引擎结构内简单，运用全文检索技术，在现有搜索引擎的算法上适当简化，便可实现。

2、从经济上来讲，校园网的搜索引擎因其简单的结构，实现起来并不需要太多的软件和硬件资源，可在微机上完成检索。

5.2 搜索引擎工作原理

搜索引擎要想完成搜索任务，必须解决两个关键问题：一是如何建立和维护全文搜索索引，二是在此基础上如何实现快速有效的检索机制。索引的组织是全文检索系统的基础，它决定着检索算法的复杂程度和检索的速度。

建立索引数据库要用到网络蜘蛛。初始化时，网络蜘蛛一般指向一个URL（Uniform Resource Locators，统一资源定位)池。在遍历 Internet 的过程中。按照深度优先或广度优先或其他启发式算法从URL池中取出若干URL进行处理，同时将未访问的URL放入URL池中，这样处理直到URL池空为止。对Web文档的索引则根据文档的标题、首段落甚至整个页面内容进行，这取决于搜索服务的数据收集策略。网络蜘蛛在漫游的过程中，根据页面的标题、头、链接等生成摘要放在索引数据库中。如果是全文搜索，还需要将整个页面的内容保存到本地数据库。用户最关心的是搜索结果是否能够满足自己的需要，尤其是当搜索引擎可以获得的信息资源非常多的情况下。目前，搜索引擎仍不能很好地理解人的查询请求。所以，目前采取的一种常见的策略是将用户的查询请求分解成若干关键字，根据这些关键字计算Web文档跟用户请求的匹配程度，从而挑出若干匹配的文档。

所以，建立索引库需要解决如下问题：

1、网络中各页面如何处理

2、网络中各节点如何处理

3、如何提高查全率和查准率

5.3 HTML 文档的扫描分析

网络上一般是 HTML（超文本标记语言）文档，虽然 HTML 文档是一种纯文本文档，可以在写字板中打开、编辑，但与正规的文本文档之间还有着很大的差别。HTML 文档中除了包含了正文文本之外，还包含了大量的用于定义文档的标题、字符集等属性信息，控制文本的显示格式和表现效果，以及引入超级链接或各种媒体的置标命令。

5.3.1 置标命令的识别处理

HTML 文档中的置标命令都是以字符“〈”开始，后跟控制命令和各种参数，最后以字符“〉”结束。例如下面的置标命令定义了文本中的字体、颜色属性：

〈FONT SIZE=3 COLOR=BLUE〉中文〈\FONT〉搜索引擎

其中“中文”两字将以 3 号蓝色字符在浏览器中显示，而后面的“搜索引擎”则用系统的默认字体和颜色显示。

有些置标命令，如〈FONT〉，必须成对使用，需要在相应的位置上给出结束置标。结束置标中的控制命令为符号“/”加上相应的命令串，如〈/FONT〉。置标命令与相应的结束置标命令之间为置标命令所要控制的字符串。而有些置标命令则单独使用，不需要结束置标，如换行命令〈BR〉。

在实际处理时，需根据置标命令的语义，把置标命令分为两类：一类是不起分隔作用的置标命令，另一类是起分隔作用的置标命令。前一类置标命令包括〈A〉、〈B〉、〈I〉、〈EM〉、〈T2〉、〈BIG〉、〈SUB〉、〈SUP〉、〈FWT〉、〈SMALL〉、〈STRONG〉、〈STRIKE〉等及它们对应的结束置标命令。这类置标命令在语义上不起分隔作用，两个字符中间出现这样的置标命令，仍应认为是两个连续的字符。

例如下面的字串：

〈FONT SIZE=3 COLOR=BLUE〉个人〈/FONT〉电脑

“个人”和“电脑”之间被标志命令隔开，但在分词处理时仍应认为“个人电脑”是一个词。起分隔作用的置标命令包括第一类置标命令以外的绝大多数。被这类置标命令隔开的两个字，应该视为不连续。如：

个人〈P〉电脑

“个人”和“电脑”之间被分段命令〈P〉分开，表示“个人”和“电脑”

两个词分别属于不同的段落，应视为两个独立的词汇。

根据置标命令的以上特点，在提取 HTML 文档中的信息时，遇到第一类置标命令，只要把置标命令去掉即可，遇到第二类置标命令时则要用空格代替它。

5.3.2 HTML 文档的扫描算法

HTML 格式的文档由两个部分组成：文件头和文件体。文件头中包含文档的标题，以及其它相关属性，这些内容不显示在浏览器的页面内。文体件是 HTML 文档的主要部分，描述的是在浏览器中显示的内容。

文档分析模块主要提取出以下 3 种信息：

（1）文档标题：通过在文件头中提取出置标命令〈TITLE〉与〈/TITLE〉之间字串而得到，保存在临时库中的 title 字段中；

（2）文档内容：通过提取出置标命令〈BODY〉与〈/BODY〉之间的所有正文文本而得到，保存在临时库中的 body 字段中；

（3）新的链接：通过提取出置标命令〈AHREF=“字串”〉中引号部分的字符串得到，保存在 URL 队列中。

对 HTML 文档的基本扫描过程如下：

（1）分析文件头，读取〈TITLE〉与〈/TITLE〉之间字串，作为文档的标题，并保存到临时库中的 title 字段中；

（2）跳到文件体；

（3）顺序扫描每个字符，直到文件结束，对扫描到的每个字符，如果是置标命令开始，则：

a）读出该置标命令，并跳到该置命令的下一个字符；

b）分析该置标命令。

如果是超链命令〈AHREF=“字串”〉，则：

①读取其中的 URL 字串，存放在临时的 URL 队列中；

②将位置计数器的值置为置标命令〈A〉后的一个字符所在的位置。

否则，如果是起分隔作用的置标命令，则：

①写一个空格到 body 字串中；

②将位置计数器的值置为当前置标命令之后的位置；

否则，保持位置计数器不变；

否则，当前字符为正文字符。

a）识别出该字符，如果为汉字编码，则要读取连续的两个字节，保存在body 字段中；

b）将位置计数器的值置为原值加上该字符的长度(汉字为 2，ASCII 字符为 1)。

5.4 节点的遍历

传统的文本信息组织是线性的、顺序的。从物理上看，它是以字符、行作为基本单位。从逻辑上看，它是以字、句、段、节、章作为单位。由于文本是顺序组织的，对它的检索、插入、修改、删除等操作十分方便。而在 Web 页面中，信息是以超文本的形式存在。从本质上说，超文本是一种管理文本信息的技术，它将文本信息存储在许多结点上，用链将这些结点连成一个网状结构。逻辑上，结点表示信息单元、片段或其组合。链表示结点间关系，如同义、反义等。

搜索引擎为了建立索引库需要遍历网络上的节点。一般从某一初始页面的URL 开始，取回该页面并将其送给 Store 服务器，由其压缩存储于数据中。然后，取出该面中的所有 URL，存入 URL 队列中，再依次按某种次序取出下一 URL 进行访问。不断重复这一过程，直到 URL 队列空。

遍历过程为：

1、访问校园网的主页，将该页面 HTML 文档内容压缩后保存在文档库中

2、取出该页面中的所有 URL，将其作为主页 URL 的子节点存入 URL 队列

3、随机选择其中一个 URL 访问，保存该页面内容到文档库中，该页面中的 URL 到 URL 队列中

4、以深度优先的原则重复第三步，直到访问的 URL 的 IP 地址超出了校园网的地址范围

5、重复第三、四步，另外选择一个 URL 访问，直到将校园网中的所有节点保存到以图的形式存在的 URL 队列中

6、在以上步骤中，若访问到重复的页面，就停止该次访问，返回该页面的上级页面，重新选择其它页面访问，并将该次错误访问记录下来，以后将不

重复错误访问

5.5 选择主要页面

以上方法检索可以具有较高的查全率，但将返回大量无用的页面。为了提高系统的查准率，将链接分析技术运用到搜索引擎中，找到检索的资源中心。

5.5.1HITS 算法

链接分析的目的就是开发和利用网页之间的链接关系，挖掘深层隐藏信息，找到页面之间的关联关系，超级链接表明的是页面之间的引用关系。Kleinberg 于 1997 年提出了基于 WWW 的链接分析算法 HIFS(Hyperlink Induced Topic Search)。在 HITS 算法中，对某个主题，算法为某个网页集合中的每个网页文档 p 计算两个权重值：authority 值和 hub 值，分别代表该文档作为该主题权威中心（authority）或资源中心（hub）的可靠性。用 A(p)和 H(p)表示网页 p 的 authority 值和 hub 值，其中 A(p)定义为所有指向 p 的页面 q 的中心权重 H(q)之和，H(p)定义为所有 p 所指向的页面 q 的权重 A(q)之和，迭代关系如下：

$A(p)=\sum H(q_i)$　（其中 q_i 是所有链接到 p 的页面）

$H(p)=\sum A(q_i)$　（其中 q_i 所有 p 所链接的页面）

HITS 算法常常和已有的文本检索系统配合使用：假设某个文本检索系统（例如搜索引擎）收到查询请求后返回一个按照相关度排序的相关页面集合，HITS 算法取该集合的前 r（比如 r=200）个页面作为算法的根页面集合（root set）R，然后将 Web 中指向这 r 个页面和从这 r 个页面指出的页面都扩展进来，得到算法迭代所需的封闭页面集合 R'，将 R' 中的每个网页视为图中的一个顶点，则这些页面之间的超链接就可看成图的边。HITS 算法虽然不能找出所有的相关页面，但是 hub 和 authority 之间是一种相互增强的关系，一个好的 hub 必然指向许多好的 authority，同样一个好的 authority 必然被许多好的 hub 链接。

5.5.2 应用于关键资源提取的链接分析算法

研究发现，HITS 算法在许多情况下得到的结果并不能让人满意，主要是由于下面 3 个原因：

（1）不同的主机之间的互增强关系。这种互增强关系表现为，有些时候同一个主机上面许多页面可能指向第二个主机上的同一个页面，这将导致第一个主机上面的枢纽分数和第二个主机上的权威分数被抬高；反之亦然。由于我们假设每一个主机的页面都是属于同一个作者或者组织，而前面所说的这种情况就无形中加大了一个作者在迭代计算中所起到的作用。

（2）自动产生的链接。由于某些原因，一些自动的链接生成工具往往被用来产生大量的超链接，这就破坏了超链接本身的客观性，由这种自动产生的链接计算出来的权威和枢纽值肯定是不能反映实际情况的。

（3）无关节点。在有些情况下，HITS 算法的扩展后的根集合中包含许多与查询主题无关的页面，如果这样的页面在 Web 子图中的链接稠密的话，迭代运算的结果就是主体漂移，使得一些权威分数和枢纽分数和高的页面是与查询无关的。为控制主体漂移，进行如下修改：

（1）将与查询主题无关的节点从 Web 子图中去掉，不让其参加迭代运算

（2）根据关联度修正不同的页面节点的权值对前面的公式进行如下修正：

$A(p)=\sum H(q_i)\times auth\_wt(q_i,p)$　（其中 q_i 是所有链接到 p 的页面）

$H(p)=\sum A(q_i)\times hub(p,q_j)$　（其中 q_i 所有 p 所链接的页面）

在全文检索的基础上直接应用该算法，根据 hub 值的高低来确定关键页面，具体算法如下：

（1）对于某一个查询结果，取全文检索返回的前 m 个返回结果作为初始集合，记为 M；从 M 中取前 r（r<m）个结果构成构成根集合 R；

（2）对于 R 中的每一个顶点，可以根据超链接的关系按照如下规则在 M 中进行扩展：

规则 1：对于 p∈R，如果 q，(q,p)是超链接，且 q∈M，则将 q 扩展进根集合 R

规则 2：对于 p∈R，如果 q，(p,q)是超链接，且 q∈M，则将 q 扩展进根集合 R

(3)对于集合 R,用向量 H、A 分别记录其中所有网页的 hub 值和 auhtority 值。

（4）利用 HITS 算法计算 R 中每个元素的 hub 值和 authority 值。

（5）取 hub 值前 k 名的页面放入集合 K 作为关键资源输出

5.6 搜索引擎的结构

检索系统分为前端和后端，前端向客户提供检索服务，后端获取 Web 文档，并建立和更新索引。系统结构如图 5.1：

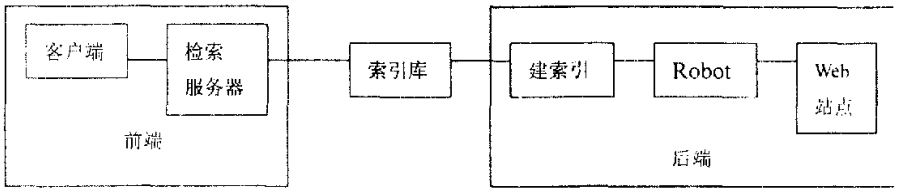


图 5.1 检索系统结构

5.6.1 获取文档数据库

和其他的全文检索应用系统一样，本系统首先需要构造一个适用于检索的文本数据库。所以获取文档数据库的步骤为：

步骤一：利用 Robot（机器人）程序定期遍历校园网内所有站点，获取相应的 Web 页面，并存储于本地计算机中

步骤二：识别和处理置标命令，将 Web 页面中的 HTML 文档转换为正规的文本文档存放

步骤三：保存全部地址信息到 URL 队列中

这样，就得到了整个网络中的全文档数据库。

5.6.2 建立 Web 页面索引

基本的索引创建方法分为两步：

1 分析原文档，产生临时中间文件，称为分析过程。对原文档进行处理，把其包含的每个词的对应信息写入一个临时中间文件。对于每个词，其在临时文件中的对应信息包括：该词所出现的当前文档的编号，在该文档中的出现频率，出现的位置序列和该词出现在下一文档中的数据指针（数据在文件中的偏移值）。

2 处理临时文档，依次从临时文件中读取每个词出现在每篇文章中的数据

信息，生成最终的倒排文件，这里称为创建过程。生成的最终倒排文件中包含每个词出现在所有的当前文档中的信息，包含：该词出现的当前文档的编号，出现频率和相应的位置序列。

5.6.3 搜索引擎中的数据结构

通过以上的变换，系统后端包括以下数据结构：

1、类似于第四章所示的常用汉字二叉语词树，每个节点的结构如下：

LF	WORD	RF

其中：

LF 指向该节点的左子树，左子树是以该树的根结点汉字构成的另一词语的后续部分根结点的 LF 部分为空；

WORD 是能与根结点汉字构成词语的汉字；

RF 指向该节点的右子树，右子树是该词语的后续部分，叶子节点的 RF 部分也是一个地址指针，指向右子树所构成词语的词语引表

每个常用汉字一个二叉树，存放在辅存上，利用请调策略调入主存。

2、一张汉字级索引表，常驻内存，记载每一个常用汉字的二叉树地址，其结构如下：

汉字编码	二叉树地址

3、二叉树上的每一个叶子节点对应一个词语索引表，该表用于记载对应词语所在页面序号，存放语辅存，也利用请调策略调入主存，结构如下

页面序号	出现频率	位置序列

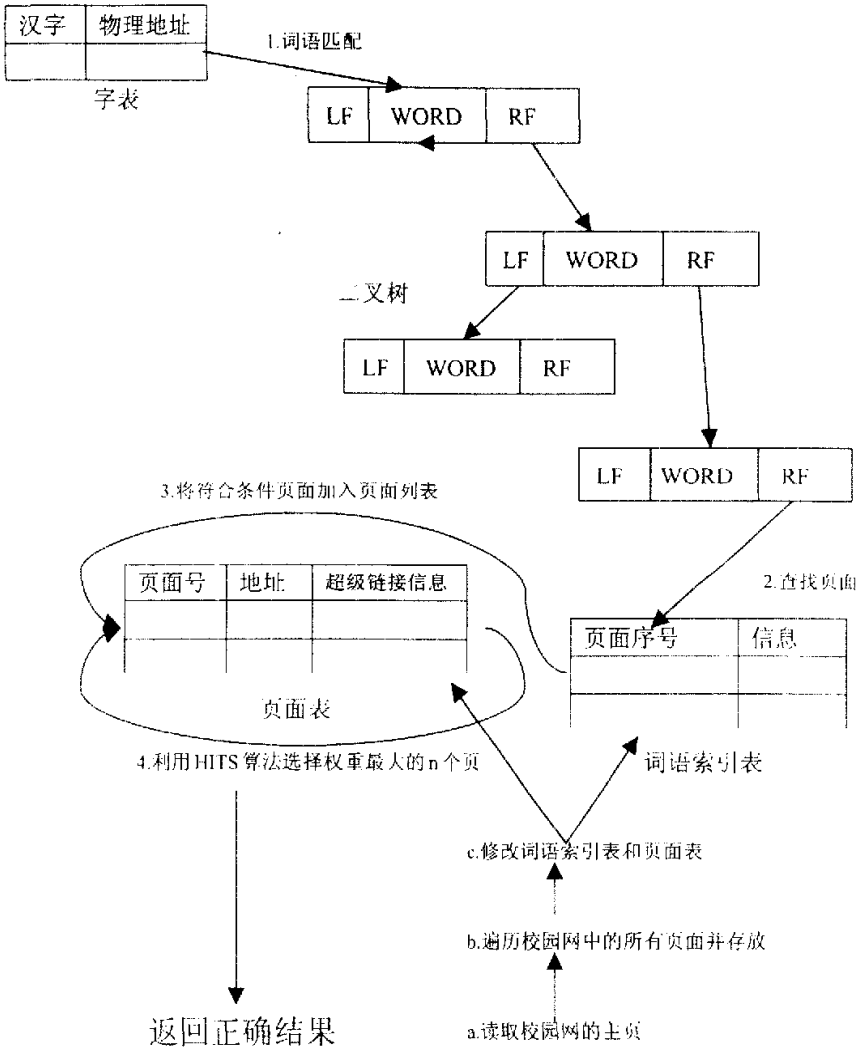
4、所有网内页面的页表，该表常驻内存，其结构如下：

序号	物理地址	超级链接信息

其中超级链接信息记载的是该页面链入和链出的页面序号，通过反复查找该表可检索出对应的资源中心。

5.6.4 检索的实现

检索系统的实现可以如下图表示：



其中：a、b、c 为后端工作过程，1－4 为前端工作过程。如果查询输入的关键字为一个较长的字串，则需要将字串按第四章所谈的改进的 MM 法进行分词，对每一个词进行如下匹配，在将多个词语检索的结果作“与”或“或”操作。

5.7 小结

在校园网络内进行有限IP内的信息检索是全文检索技术的又一重要应用。校园网内的搜索引擎不同于普通的用于 Internet 的搜索引擎，它结构简单，构造方便，具有较高的查全率和查准率，可以针对不同的网络结构特制不同的搜索引擎。本章所提出的局域网内的搜索引擎算法具有一定的普遍意义，使用与大多数的校园网络，并能得到简洁准确的查询结果。

第六章 结束语

全文检索技术是当今信息检索发展的最高端和最前沿，它从上个世纪发展至今，在文本检索方面已比较成熟。近年来，随着 Internet 的迅速发展，网络上信息日趋繁多和复杂，对信息检索的要求越来越高。全文检索技术以其较高的查准率和查全率，较简单的检索算法被广泛应用于网络信息的检索中，如：搜索引擎，数字图书馆文献检索，针对用户的各种信息的检索，等等。但世上并无万能东西，全文检索技术也有其缺点和不足，这也是广大学者尽力研究，努力改进的部分。由于汉语自身的特点，使得汉字全文检索远比西文全文检索复杂，汉字全文检索技术也远不如西文全文检索技术发展得充分、成熟。汉字全文检索技术发展的障碍主要在于以下几点：

1、高的查全率和高的查准率不可能同时拥有。使用单汉字检索方法，具有较高的查全率和较低的查准率；使用词检索方法，具有较高的查准率和较低的查全率。

2、在词检索方法中，分词易产生歧义。

3、不论是字检索还是词检索，倒排索引文件都会占用较大的存储空间。特别是字检索，虽然检索算法简单，单庞大的索引文件极影响检索速度。

4、检索会产生大量的重复或无用的符合检索条件的文档，不利于用户从中读取有用的信息。

针对以上问题，本文主要做了以下工作：

1、介绍了国内外全文检索技术的发展动态，分析了全文检索技术的优缺点，指出了今后的发展方向。

2、分别介绍了字索引和词索引的主要方法。

3、将全文检索技术运用于搜索引擎中，实现校园网络中的检索。

本文旨在研究中文全文检索技术，并尝试构造适合于一定范围的搜索引擎。今后还需对如何更大程度的提高检索速度和检索效率作进一步研究。

参考文献

1. 苏广利. 因特网信息检索工具的十大发展方向. 图书馆建设，2001，1：75-76
2. 雷春明，焦玉英. Web 页面信息检索智能代理模式研究. 现代图书情报技术，2001，第 3 期：30-32
3. 黄崑，符绍宏. 自动分词技术及其在信息检索中应用的研究. 现代图书情报技术，2001，第 3 期：26-29
4. 高迎，王丽君，王锡钢. Simutem：一个中文信息检索系统. 鞍山师范学院学报，2001，第 3 期：82-85
5. 储节旺，鲍克忠. 网上信息检索目标与策略的转换. 情报理论与实践， 2002，第 25 卷第 1 期：55-57
6. 曹元大，贺海军，涂哲明. 中文 Web 文档全文检索系统的设计及实现. 北京理工大学学报，2002，第 22 卷第 1 期：68-71
7. 周前，肖建华. 全文检索中的文本学习技术研究. 湖南工程学院学报，2001，第 11 卷第 2 期：64-67
8. 曹元大，贺海军，涂哲明，王琴. 全文检索字索引技术的研究与实现. 计算机工程，2002，第 28 卷第 6 期：260-262
9. 刘志勇. 网络环境下信息检索效率的评价. 大连大学学报，2002，第 23 卷第 1 期：110-112
10. 张开舟，张惠惠. 万维网信息检索系统开发技术. 情报学报，2002，第 21 卷第 1 期：42-47
11. 周涛. 两种全文信息检索系统的比较研究. 情报理论与实践，2002，第 25 卷第 2 期：138-140
12. 陈华辉. 一个中英文全文搜索引擎的设计与实现. 计算机应用研究 2001，第三期：131－133
13. 陈淑燕，瞿高峰. 全文检索系统的数据库设计. 延安大学学报（自然科学版），2001，3. 第 20 卷 第 1 期：31－34
14. 郑庆华，张炜. 超文本全文检索技术的研究与实现. 西安交通大学学报，2001，4 第 35 卷 第 4 期：377－381

15. 张俭恭，陈定权. 汉字全文检索系统的关键技术与实现. 现代图书情报技术，2001，第 2 期：16－18

16. 李志蜀，李果. 中文搜索引擎的原理剖析及开发实现技术. 计算机应用研究，2001 第 11 期：96－99

17. 杨建林. 全文检索研究. 情报理论与实践，第 23 卷 2000 年第 1 期：12－13

18. 苏新宁. 超文本技术在全文检索系统中的实现. 情报学报，2000 年 12 月，第 19 卷第 6 期：582－585

19. 马迎春. 全文检索系统概述. 情报科学，2000 年 12 月 第 18 卷第 12 期：1132－1135

20. 董春晓. 万维网上的全文检索技术及其发展. 情报理论与实践 第 23 卷 2000 年第 1 期：53－55

21. 李广建，黄永文. 基于 WWW 的全文检索系统设计与实现. 现代图书情报技术，2000 年第 2 期：26－28

22. 裘江南，马克芬. 一种基于 Web 的全文检索系统的建立方法. 现代图书情报技术，2000 年的 2 期：32－34

23. 顾春庆，于玉，顾永立，胡运发. 汉字全文检索的实现与探讨. 计算机工程，1998 年 2 月 第 24 卷第 2 期：69－72

24. 赵曾贻，陈天娥，朱兰. 一种基于语词的分词方法. 苏州大学学报（自然科学），2002 年 7 月第 18 卷第 3 期：44－48

25. 蒋微. 中文搜索引擎的自动分词算法. 电脑开发与应用，2002 年第 15 卷第 6 期：26－27

26. 陈天娥，赵曾贻. 基于字、词、词组的中文搜索引擎分词系统. 武汉工业学院学报，2002 年第 3 期：37－40

27. 郭辉，苏中义，王文，崔骏. 一种改进的 MM 分词算法. 微型电脑应用. 2002 年第 18 卷第 1 期：13－16

28. 赵新民. 搜索引擎的中文信息处理技术. 现代情报，2002 年 5 月第 5 期：98－100

29. 陈桂林，王永成，韩客松，王刚. 一种改进的快速分词算法. 计算机研究与发展，2000 年 4 月第 37 卷第 4 期：418－424

30. 欧振猛，余顺争. 中文分词算法在搜索引擎应用中的研究. 计算机工程与应用，2000. 8：80－83

31. 谭琼，史忠植. 分词中的歧义处理. 计算机工程与应用，2002，11：125-128

32. 邹海山，吴勇，吴月珠，陈阵. 中文搜索引擎中的中文信息处理技术. 计算机应用研究，2000，第 12 期：21-24

33. 闫引堂，周晓强. 交集型歧义字段切分方法研究. 情报学报，2002，第 19 卷第 6 期：637-643

34. 秦洪晶. Internet 中文信息检索技术. 青海大学学报，2000，第 13 卷第 4 期：86-89

35. 丁丰，董娜，林碧琴，袁保宗. 自然语言处理系统中自动分词的研究. 北方交通大学学报，1999，第 23 卷第 6 期：31-33

36. 严威，赵政. 开发中文搜索引擎汉语处理的关键技术. 计算机工程，1999，第 25 卷第 6 期：5-7

37. 李盛涛，吴丽辉，于满泉，潘文锋，余智华，王斌程，学旗. 主题 Web 信息采集的研究与分析. 语言计算与基于内容的文本处理. 清华大学出版社. 2003 年 7 月：488－494

38. 傅国宏，王晓龙. 基于词形的汉语文本切分方法. 情报学报，1999，第 18 卷第 3 期：235-240

39. 刘说，王斌，杨志峰，张鑫. Web 关键资源发现中的链接分析技术. 语言计算与基于内容的文本处理. 清华大学出版社. 2003 年 7 月：495－500

40. 郑延斌. 自动分词中的歧义处理. 微型机与应用，1998，第 6 期：9-10，49

41. 邹育理. Web 环境下的信息检索. 大学图书情报学刊，2001，第 3 期：14-16

42. 鲁松，白硕，黄雄，张健. 基于向量空间模型的有导词义消歧. 计算机研究与发展，2001，第 38 卷第 6 期：662-667

43. 张英福，郝志娟. 计算机检索策略初探. 图书馆学研究，2000，6：63-65

44. 陈建秋，邓飞其，刘发贵. 智能化搜索引擎分析与探讨. 广州大学学报（自然科学版），2002，第 1 卷第 3 期：39-42

45. 何军，周明天. 信息网络中的信息过滤技术. 系统工程与电子技术，2001，第 23 卷第 11 期：76-79

46. 李蕾，王楠，张剑，钟义信，郭祥昊，贾自燕. 中文搜索引擎概念检索初探.

计算机工程与应用，2000，6：1-4
47. 董春晓. 万维网上的全文检索技术及其发展. 情报理论与实践，2000，第 23 卷第 1 期：53-55
48. 金燕，李建华，杨宇航. WWW 上的全文信息检索技术. 计算机应用研究，1999，第 1 期：40-43
49. 朱靖波，姚天顺. 中文信息自动抽取. 东北大学学报（自然科学版），1998，第 19 卷第 1 期：52-54
50. 郭祥昊，钟义信，杨丽. 基于两字词簇的汉语快速自动分词算法. 情报学报，1998，第 17 卷第 5 期：352-357
51. 尹锋. 汉语自动分词研究的现状与新思维. 现代图书情报技术，1998，第 4 期：22-26
52. 杨雅群，张建中，刘兵. 超文本超媒体技术及其发展. 电子展望与决策，1997，第 4 期：38-41
53. 刘伟权，钟义信. 自然语言处理与全文情报检索. 情报理论与实践，1997，第 20 卷第 1 期：43-46
54. 余盛可. 超文本中的迷路问题. 计算机研究与发展，1994，第 31 卷第 5 期：24-28
55. 肖云，孙茂松，邹嘉彦. 利用上下文信息解决汉语自动分词中的组合型歧义. 计算机工程与应用，2001.19：87－90
56. 杨雅群，张建中，刘兵. 超文本超媒体技术及其发展. 电子展望与决策，1997 年第 4 期：38－41
57. 潘有能. 一个自动分词分类系统的实现. 情报学报，2002 年 2 月第 21 卷第 1 期：38－41
58. 丁承，邵志清. 基于字表的中文搜索引擎分词系统的设计与实现. 计算机工程，2001 年 2 月第 27 卷第 1 期：191－193
59. 严海兵. Internet 搜索引擎检索功能的研究. 苏州城市建设环境保护学院学报，2001 年 3 月第 3 卷第 1 期：58－62
60. 王丽君，高迎，王锡钢. 中文检索系统中查询的扩展. 小型微型计算机系统，2002 年 7 月第 23 卷第 7 期：894－896

读研期间发表的论文

1. 遗传算法在网络流量及带宽分配中的应用，中南民族学院学报（自然科学版），2001，9

2. 利用全文检索技术实现 Web 也的搜索，数理医药学杂志，2003 年第 16 卷第 5 期

致　谢

光阴荏苒，我结束了在华中师范大学两年的学习生活。在这难忘的两年时间里，我得到了众多老师和朋友们的无私帮助，是他们的鼓励和支持使我有了战胜困难、完成学业的勇气。

首先，我要对我的导师何婷婷副教授表示衷心的感谢和敬意。在学习中，她以严谨的治学态度、丰富的学术知识对进行了我不厌其烦的启发和指导。她渊博的学识给我树立了学习的榜样，而她那一丝不苟、孜孜以求之的工作作风更让我在获得知识的同时也明白了做人的道理。

感谢肖德宝教授、谭根稳书记、胡金柱教授、冯刚教授、陈利副教授、魏长华教授、魏开平副教授和李晓燕教授等所有老师对我的精心栽培和谆谆教诲。

最后，我要特别感谢我的班主任吴爱莲老师，她热情的支持和关怀给了我进取的力量，我唯有以自己今后的努力和对社会的奉献来回报所有关心和支持我的人。

作者 于波
2003 年 12 月