主题搜索引擎的研究

李瑞芳,杨 娜
(沈阳化工学院 计算机科学与技术学院,辽宁 沈阳 110142)

摘 要:介绍了将开源的全文检索工具包 Lucene 嵌入到自己的搜索引擎中来满足开发主题搜索引擎的需求。并基于 Lucene 中文分词的不足设计了一个比较完善的中文分词器,然后将其引入具体应用中,并且与传统搜索引擎在性能上进行了比较。

关键词:Lucene;全文检索技术;主题搜索引擎;索引;中文分词
中图分类号:TP393 **文献标识码**:A

Research of thematic search engine

LI Rui Fang, YANG Na
(Department of Computer Science and Technology, Shenyang Institute of Chemical Technology, Shenyang 110142, China)

Abstract: In order to meet the require of developing thematic search engine, this paper introdued the method to embed open-source Lucene search toolkit into its own search engine. Because of the inadequacy of Chinese word segmentation based on Lucene, the paper designed a more perfect Chinese segmentation, then employed it in the application, and compared with traditional search engine in terms of performance.

Key words: Lucene; full-text retrieval technology; thematic search engine; index; Chinese word segmentation

国际互联网的迅速发展使得以 Internet 为载体的中文电子信息愈来愈多,传统搜索引擎采集索引查询内容不断扩大,这不但使搜索引擎面临巨大的困难,而且越来越不能满足主题用户的需求。例如,为了获取数条相关信息,用户不得不在大量的失效信息、甚至垃圾信息中费力寻找。目前人们对搜索引擎的首要关注点已经从如何找到更多的信息转向如何快速找到准确、有用的信息。因此,人们希望在企业应用中或者个人产品中加入自己的搜索功能。这样不仅可以对企业发布的信息建立索引,也可以对企业计算机内长期积累的电子文档资料建立索引,实现方便快速查找。

在 Lucene API 的基础上开发面向主题的搜索引擎[1]是一种有效、低成本的选择,因为 Lucene 全文数据库采用倒排文件索引技术[2],所以查询速度优于关系型数据库,而且可以免费下载。基于 Lucene 的优势已有很多企业将其应用到自己的搜索引擎中,如 Eclipse 开发环境的内部搜索引擎就是用 Lucene 构建的。但由于 Lucene 自带的中文分词只能将中文切成单字不能实现词语的切分,因此,符合需求的中文分词器有待人们去开发,并将其加入中文分词模块来实现更高效的检索。

1 全文检索引擎 Lucene

1.1 Lucene 简介

Lucene 是 apache 软件基金会 jakarta 项目组的一个子项目[3],是一个开放源代码的全文检索引擎工具包,它不是一个完整的全文检索引擎,而是一个全文检索引擎的架构,提供了完整的查询引擎和索引引擎,它为数据访问和管理提供了简单的函数调用接口,可以方便地嵌入到各种应用中,实现针对应用的全文索引/检索功能。

Lucene 的 API 接口设计得可以通用,输入输出结构都很像数据库的表⇒记录⇒字段,所以很多传统的应用文件、数据库等都可以方便地映射到 Lucene 的存储结构/接口中。总体上看,可以先把 Lucene 当成一个支持全文索引的数据库系统。

1.2 Lucene 系统结构

Lucene 系统结构[2]如图 1 所示。从图 1 中可以看到,Lucene 的系统由基础结构封装、索引核心、对外接口三大部分组成。其中,直接操作索引文件的索引核心又是

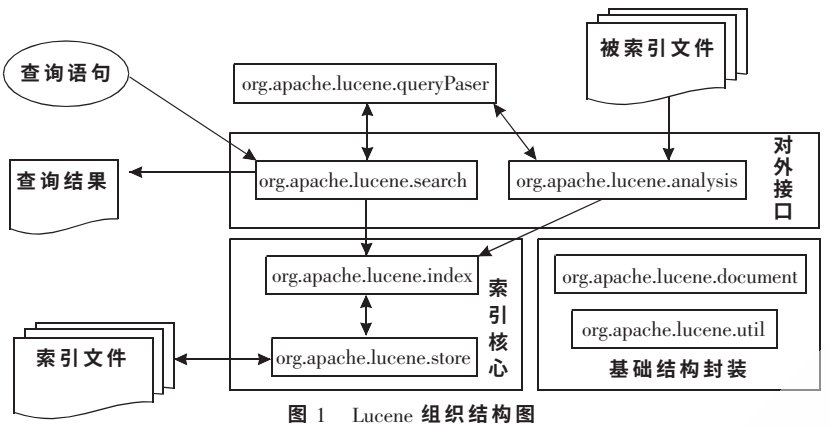

图 1 Lucene 组织结构图

系统的重点。Lucene 将所有源码分为 7 个模块（在 Java 语言中以包来表示），各个模块所表示的系统部分见图 1。需要说明的是：org.apache.lucene.queryPaser 是 org.apache.lucene.search 的语法解析器，不被系统之外实际调用，因此没有当作对外接口看待。从面向对象的观点来考虑，Lucene 应用了最基本的一条程序设计准则：引入额外的抽象层以降低耦合性。首先，引入对索引文件的操作 org.apache.lucene.store 的封装，然后将索引部分的实现建立在 org.apache.lucene.index 之上，完成对索引核心的抽象。在索引核心的基础上开始设计对外的接口 org.apache.lucene.search 及 org.apache.lucene.analysis。

1.3 Lucene 程序运行机制

Lucene 系统功能强大，实现复杂，但从根本上主要包括 2 个主要功能：(1)建立索引库[4]，也就是将待索引的纯文本内容经切分词后索引入库；(2)检索索引库，即根据查询条件从索引库中找出符合条件的文档。

在研究建立索引库时首先要知道如图 2 所示的 Lucene 索引机制的架构。

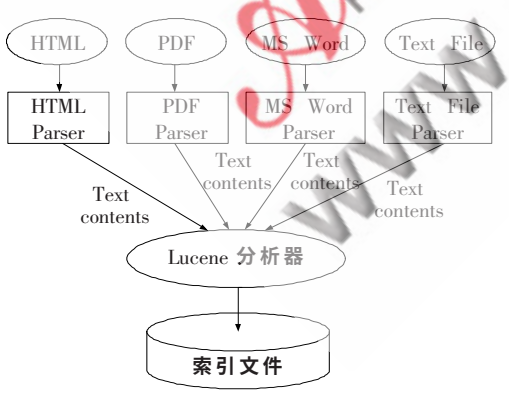

图 2 Lucene 索引机制架构

从图 2 可以看出，Lucene 索引过程分为 3 个主要操作阶段：将数据转换成文本；分析文本；将分析过的文本保存到索引库中。首先，Lucene 使用各种解析器对各种不同类型的文档进行解析，如对于 HTML 文档，HTML 解析器会做一些预处理的工作，过滤文档中的 HTML 标签等，然后输出文本内容，接着 Lucene 的分词器从文本内容中提取出索引项以及相关信息。

检索索引库的运行逻辑如下：

(1)输入查询条件，如用户希望查询到含有词“编程”和“入门”但不含词“Java”的记录，则输入条件为“编程+入门-Java”；查询条件输入搜索器(lucene.search)，搜索器里有 1 个查询解析器(lucene.queryParser)，搜索器调用这个查询解析器来解析查询条件。

(2)查询条件“编程+入门-Java”被传送到查询解析器中，解析器将对“编程+入门-Java”进行分析，首先分析器解析字符串的连接符，即加号和减号，然后调用语言解析器(lucene.analysis)对每个词进行切词，一般英文将按空格来切词，最后得到的查询条件表示为：“编程”AND “入门”AND NOT “Java”。

(3)查询器根据查询条件检索事先已建立好的索引库，得到查询结果，并返回结果集 lucene.search.Hits，Hits 类似于 JDBC 中的 ResultSet。

2 中文分词

因为 Lucene 提供的两个中文分析器(ChineseAnalyzer 和 CJKAnalyzer)只能将中文切成单汉字，这对于绝大多数中文用户来说很不方便，因此有必要开发适合自己的中文分析器。本系统基于字符串匹配的分词技术实现了一个中文分词器。它是按照一定的策略将待分析的汉字串与 1 个“充分大的”词库中的词条进行匹配。若在词库中找到某个字符串则匹配成功(识别出 1 个词)。按照扫描方向的不同，串匹配分词方法[5]可以分为正向匹配和逆向匹配；按照不同长度优先匹配的情况，可以分为最大(最长)匹配和最小(最短)匹配。本研究使用的方式是采用基于字典的双向匹配算法，即需要进行正向和逆向 2 次匹配。传统的方法都是先进行 1 次正向匹配，然后再进行逆向匹配，每次只有 1 个方向在进行匹配，而另 1 个方向的匹配过程需要第 1 个结束后才开始。本文的设计方法是把多线程技术引入到中文分词程序当中[6]。因此可以在正向匹配时，同时并发地执行逆向匹配，提高了运行速度和执行效率。以下为正向匹配和逆向匹配都将调用的核心程序：

载入处理完的文本，以字符数组的形式存储起来。建立 2 个方法：(1)查找在文本中出现的词，并且返回词的长度；(2)用来统计词出现的次数和控制程序进度。

方法 1：public int zhaoCi (char [] ch, int juli, TreeMap<String, Integer>[] tmp)。

ch 数组表示的是文本内容，juli 是词的长度，TreeMap 数组存储不同长度的词。

方法 2：public synchronized void ciPinTongJi (HashMap<

String, Integer>hm, char[]ch, int size, TreeMap[] tmp)。

方法 2 是并发执行的，正反匹配都需要这个方法。HashMap 存储字典中词的第 1 个字，ch 数组表示的是文本内容，size 是文件的长度，TreeMap 是数组存储不同长度的词，按照 2 个字、3 个字、多个字的存储顺序存储；而且方法 2 是对外的接口，方法 2 调用方法 1，并利用方法 1 返回的结果得到分词的结果和词频结果。当从文本读入 1 个字时，使用 contain( )来判断 HashMap 中是否存在这个字的映射，如果存在就取得长度等于字典中最长词的一段内容，在 TreeMap 数组中进行查找，如果在 TreeMap 中找到对应的映射则对应的键值加 1，输出时在词后面加上分割符号'\'，然后继续重复前面的步骤，直到文件结束，退出；如果 TreeMap 中不存在，那么 $i+1$，读取下一个字，重复前面的步骤，直到文件的结尾，退出，程序结束。

正向匹配程序流程图如图 3 所示。

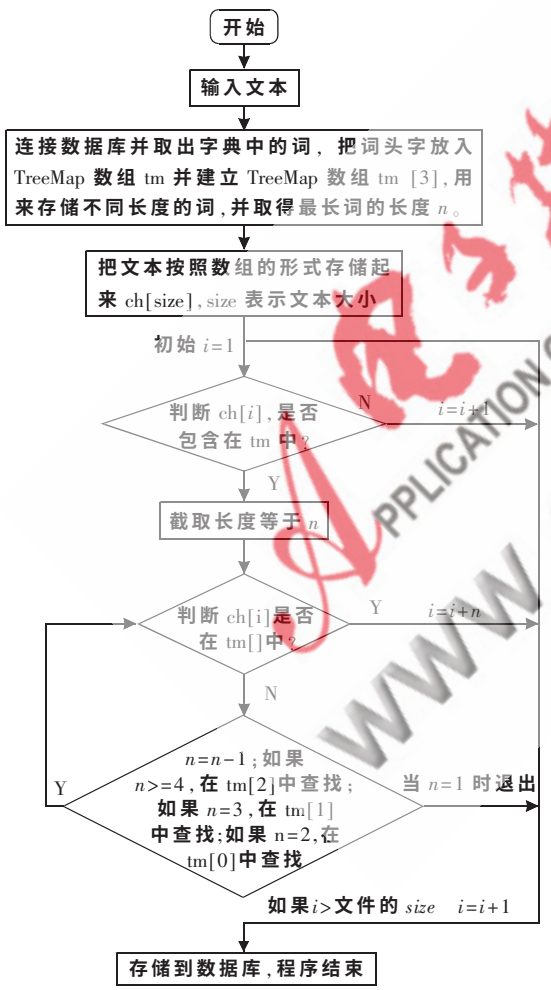

图 3 正向匹配流程图

3 全文检索引擎 Lucene 的应用

Lucene 本身只是一个组件，若想让 Lucene 真正起作用，还得在 Lucene 基础上进行必要的 2 次开发[7]。下面的方案是对 Lucene 的应用研究，在本系统实现过程中要解决的关键问题有：数据加工及文本数据库的实现；全文数据索引；全文数据检索和结果处理。

3.1 运行环境

操作系统：Windows NT/2000/xp；开发语言：Java、JSP；开发环境：MyEclipse6.5；API 插件：Lucene2.3.2 (Jakarta Lucene 是一套免费的开放源代码，由 Apache Jakarta 开发)；Web 服务器：Apache 的 Tomcat6.0。

Lucene 在 Java 环境下运行，因此首先要安装 jdk 并设置环境变量 JAVA_HOME，还要安装 tomcat6.0。到 Lucene 的官方网站 http://jakarta.apache.org/Lucene/下载 1 份拷贝(笔者下载的是最新版 2.3.2)，下载后将得到一个名为 lucene-2.3.2.zip 和 apache-ant-1.7.0-bin.zip 的压缩文件，将其解压即可。

3.2 系统结构

该应用分为 3 部分：(1)数据库发布平台，包括服务器、Java 环境、Lucene API、中文分词模块；(2)HTML 文件倒排档生成系统；(3)服务器端执行的 JSP 程序和用户界面。系统结构如图 4 所示。

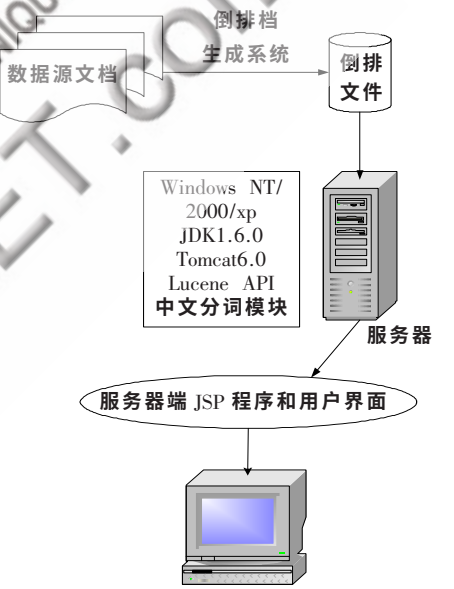

图 4 系统结构图

3.3 Lucene 的扩展

对于 Lucene 组件包，为了能够支持中文，要进行修改。首先将改写后支持中文的分析包 IK-Analyzer.jar 加入到发布包 Analysis 包中。解开 Lucene.zip，在解开的目录 src\demo\org\apache\Lucene\demo 下打开 IndexHTML. java。在第 1 处"import org. apache. lucene. analysis. standard. StandardAnalyzer;" 下面加 1 行 "import org.apache. lucene. analysis. IKAnalyzer;"，把第 2 处"writer=new IndexWriter (index, new StandardAnalyzer (), create);"注释掉，换成"writer=new IndexWriter(index, new ChineseAnalyzer(), create);"解开 Luceneweb. War，释放出 configuration. jsp 和 result. jsp 以及 web. xml。编辑 configuration. Jsp，找到 indexLocation 变量，赋值成"/index"(或者用户自己建立的索引的目录名称)；编辑 result. Jsp，找到"Analyzer analyzer=new StopAnalyzer ();" 删除或者注释

(下转第 6 页)

```
c=b;
d=a;
```

通过 File->Compile to dll 将该 M 文件转化为 dll，在 Debug 目录下可找到 Exchange2.lib 和 Exchange2.dll 这 2 个文件，将其放入 TestMatcom 工程目录下并在 TestMatcomDlg.cpp 中添加以下代码：

```
#pragma comment(lib, "Exchange2.lib")
extern "C" int DLLX_stdcall Exchange2_v (char* emsg, int nlhs, Mm* plhs[ ], int nrhs, Mm* prhs[ ]);
```

MATCOM 将 M 文件里面的函数 Exchange2 转化成了 Exchange2_v 函数，该函数的第 1 个参数 emsg 用于传递 1 个消息字符串，可赋值为 NULL；第 2 个参数 nlhs 是 Exchange2 函数定义的输出参数的个数；第 3 个参数 plhs[]是指向输出参数的指针数组；第 4 个参数 nrhs 是 Exchange2 函数中定义的输入参数的个数；第 5 个参数 prhs[ ]是指向输入参数的指针数组。

在消息响应函数中添加如下代码：

```
Mm a, b, c, d;
a=zeros(1,3);
b=ones(1,3);
b.r (1,2)=100;      /* 将矩阵 b 的第 1 行第 2 列元素赋值为 100 */
Mm* Input[2]={&a,&b};
Mm* Output[2];              //不用为其分配空间
//调用 dll 函数
Exchange2_v(NULL, 2, Output, 2, Input);
//查看第 1 个输出参数的结果
Mm lookOutput1=*Output[0];
//查看第 2 个输出参数的结果
Mm lookOutput2=*Output[1];
```

即实现了调用 dll 里的 Exchange2_v 函数。

(下转第 12 页)

(上接第 3 页)

掉，改成“Analyzer analyzer=new org. apache. lucene. analysis. IKAnalyzer();”。这样就扩展了 Lucene 的中文分词的功能。

Lucene 并没有规定数据源的格式，而只提供了 1 个通用的结构(Document 对象)来接收索引的输入，因此输入的数据源可以是：数据库、WORD 文档、PDF 文档、HTML 文档……，只要能够设计相应的解析转换器将数据源构造成 Docuement 对象即可进行索引。本设计实现了 doc、ppt、xls、pdf、txt、xml 解析转换器将其文本信息提取出来。

3.4 搜索性能的比较

经过多次测试取平均值，本设计在搜索主题信息的平均速度上比 Google 要快，虽然数量上不如 Google 检索的多，但在信息符合度上明显比其强。这样就已经符合主题用户，不一定要多只要精而且节省时间的需求，这对于当今效率优先的市场来说是非常有竞争力的。应用 Lucene 的搜索引擎的检索速度与计算机的配置有关，配置较好的计算机的搜索时间相对要少。以检索关键字编程为例，Lucene 与 Google 性能比较结果如表 1 所示。

表 1 性能比较

	搜索记录个数	平均时间/s	符合度/%
Lucene	180	0.225	98
Google	20 400 000	0.298	45

全文检索引擎 Lucene 所构建的搜索引擎的搜索个数是由磁盘存储的信息量的多少决定的，搜索时间除了第 1 次检索有点慢，以后的时间耗费明显少于通用搜索引擎。虽然通用搜索引擎提供的信息量大，但是并不是所有的信息都符合用户的需求，用户要在大量的信息中筛选有用的信息要花费大量的时间，可见主题搜索引擎的优势，本设计基本符合预期的结果。

本文提出了一种解决中文全文检索的方法，嵌入到 Lucene 中可以应用到搜索引擎、中小企业网站站内检索、个人用户桌面搜索引擎建立、特定文档检索数据库建立等，从而实现对目标文档方便地检索管理，提高检索效率。并且通过对全文检索引擎 Lucene 的研究以及在 Lucene API 上的扩展，可以开发出多种应用程序，如：网站内容搜索系统、可检索的邮件系统、海量文献数据搜索系统。为了开发出性能指标更高的搜索引擎可以根据现有的排序算法或自定义排序算法自行开发结果排序模块加入到 Lucene 中来进行测试比较，这些都有待于继续研究。

参考文献

[1] 聂颂.具有自动分类功能的主题搜索引擎的研究[D].天津：天津大学，2004：7-9.
[2] 车东.在应用中加入全文检索功能——基于 Java 的全文索引引擎 Lucene 简介[EB/OL].http://www.chedong.com/tech/lucene.html.2005-07.
[3] Lucene[EB/OL].2002.http://lucene.apache.org/java/docs/index.html.2002.
[4] 曹元大，贺海军.全文检索字索引技术的研究与实现[J].计算机工程，2002.28(6)：286-288.
[5] 黄昌宁.中文信息处理中的分词问题[J].语言文字应用，1997(1)：72-78.
[6] 李志蜀，李果.中文搜索引擎的原理剖析及开发实现技术[J].计算机应用研究，2001(11):98-101.
[7] 肖创柏.基于全文检索技术的商业信函处理系统的设计与实现[J].计算机应用研究，2004(1)：150-152.

(收稿日期：2009-07-03)