·信息资源开发与利用·

基于双层 PDF 和 Lucene 技术的全文检索研究与实现

向 禹 吴世明
（中南大学档案馆，湖南 长沙 410083）

〔摘 要〕通过建设双层 PDF 全文数据库、创建索引和全文检索等实现过程来阐述相关技术的研究和运用。以建设全文数据库为基础，研究结构化信息与非结构化数据的合并管理，对目录数据和全文数据的同步索引，基于 Lucene 技术，实现档案管理系统的一站式智能化档案全文检索，提升档案查全率。

〔关键词〕双层 PDF；全文检索；档案管理；Lucene
DOI:10.3969/j.issn.1008－0821.2014.06.015
〔中图分类号〕TP391 〔文献标识码〕B 〔文章编号〕1008－0821（2014）06－0075－04

Research & Realization of Full Text Retrieval Based on Double PDF and Lucene Technology

Xiang Yu Wu Shiming
（Archives of CSU，Changsha 410083，China）

〔**Abstract**〕Through the construction of double PDF full text database，created the research and application process to the relevant technical indexes and full－text retrieval etc. With the construction of double PDF archives full－text database as the foundation，combined with management of structured information and non structured data，synchronization index on the catalog data and text data，based on Lucene technology，realized one－stop intelligent archives management system of full text retrieval，to enhance the search rate.

〔**Key words**〕double PDF；full text search；archives management；Lucene

由于档案的凭证性、惟一性和不可替代性，导致用户和档案行业更注重查全率。传统的档案管理手段，由于对标引和著录标准的理解、执行和操作、人员责任心等方面的差异，导致著录信息和检索效果不尽人意。基于 Lucene 技术，依托双层 PDF 文档，对结构化和非结构化信息合并管理，在档案管理系统中实现一站式全文检索，具有很重要的现实意义。

1 档案检索研究现状

传统档案检索，主要是对档案信息著录和标引进行研究，编制检索目录和目录检索系统，常见的检索工具有主题、分类、字序、文号等多种方式，检索系统有简单检索、复合逻辑组配表达式检索等。著录和标引质量提高，检索工具完备均能提高查全率，但存在缺陷，且效率较低。要实现高查全率，必须研究在档案文档中实现内容检索。

Lucene 是一个非常优秀的全文本型检索框架[1]，在文本型的全文检索方面得到广泛的支持和运用；然而，对纸质档案进行数字化扫描加工，最好的存储方式仍为图片格式的非文本型文档，要实现全文检索并非易事；基于图像的检索技术的研究也还不成熟，效果并不理想。

2 全文检索思想与技术

档案资源数据有多种类型：一是结构化数据，有固定格式和长度，如数据库或者元数据，数据表格等；二是非结构化数据，特点是不定长和无固定格式，如 Word、PDF、JPG 等文档；三是半结构化数据，如 XML、HTML 等，这类数据比较灵活，可根据需要按结构化处理，也可按非结构化处理，在使用 Web Service 方式的系统集成对接

收稿日期：2013－11－29
作者简介：向 禹（1976－），男，信息技术部主任，副研究馆员，硕士，研究方向：数据库技术、信息处理技术、档案信息化建设，发表论文 10 余篇，

时，协议中采用的数据传输格式大多为 XML。对于结构化的语句，采用 SQL 语句很容易实现检索。非结构化的数据，通常称作全文数据，检索方式有两种[2]：一种是顺序扫描法，对每一个文档都从头至尾进行扫描，搜索出包含检索词的文档，如 Windows 系统中的查找功能，但这种方式，搜索效率低，速度慢；另一种方式便是我们要重点讨论的全文检索。

2.1 全文检索思想

由于结构化的数据格式是有规律的，用算法容易实现很高的检索效率。全文检索的基本思想便是：把全文数据中信息提取出来，重新进行组织成索引，使其结构化规律化，再按一定的算法对其进行检索。从过程上来看，可简单地分为索引和检索两个过程，但在实际处理过程中，包含的模块构成有：前端查询平台、中文分词、解析引擎、后台管理等。

2.2 双层 PDF 技术

非结构化的数据，又分为文本型和非文本型。对于文本型或者超文本型的文档，全文检索的研究应用已经比较广泛和成熟。而非文本型的文档无法直接实现全文检索，双层 PDF 文档技术便是解决这一问题的最佳方式之一。

双层 PDF 文件是一种包含 Text 层和 Image 层的多层结构 PDF 文件，两层内容位置上相对应，Image 层是原始图像，保留了原始档案的效果；Text 层是 Image 层的 OCR 识别结果，支持选择、检索和复制等功能。通过程序控制可实现两个图层的任意显示和切换，可实现检索词的精确定位。双层 PDF 文档可以是图像型通过档案数字加扫描加工而成；也可以是文本型，通过文本文件如 WORD 转换。

2.3 全文检索引擎 Lucene

Lucene 是目前最为流行的基于 Java 开源全文检索工具包[3]。它并不是一个完整的搜索程序[4]，不能直接嵌入系统中使用；而是一个类库，一种思想和架构。Lucene 提供简单的工具包，方便软件开发人员在应用系统中实现全文检索功能。Lucene 具备五大优点[5]：索引文件格式独立于应用平台；可分块索引，为增量文件建立小索引，通过与原索引合并，提升效率；面向对象的架构，便于扩充；独立的文本分析接口，与语言和文件格式无关；具备强大的查询引擎，包括布尔逻辑、分组查询、模糊查询等，开发人员无需再编写代码。

Lucene 的源码由 7 个模块（包）组成：分词模块、索引管理、检索管理、数据存储管理、查询分析器及公用类库。为了对文档进行索引，Lucene 提供了 5 个基础的类，Document、Field、IndexWriter、Analyzer、Directory。全文检索系统功能强大，实现起来也比较复杂，但从实现过程来看，主要分为索引和检索两大功能。

3 全文检索的实现

主要运用 lucene 技术，基于 PDF 文档，对中文分词、解析引擎、索引、过滤、专业词库等方面进行了重构与优化，由前端查询、索引模块、分词、搜索引擎、后台管理等模块构成。通过全文检索的分词系统、索引系统、引擎系统将海量数据快速展现在用户面前，并支持多关键词、同义词、近义词等检索。

3.1 创建双层 PDF 全文数据库

建设双层 PDF 全文数据库是实现全文检索的基础，减少对纸质档案的使用，从某种意义上来说，也保护了纸质档案。市场上已经有许多产品或者生产线，可以实现档案的双层 PDF 数字化加工。在档案数字化加工过程中，将纸质档案扫描加工后的图片转换处理成双层 PDF 文档，在挂接到档案管理系统中的相应案卷和卷内文件目录之后，原文的存放地址信息自动存入数据库的原文关系表中，通过 ID 号（Rec_id）与案卷和卷内文件目录 Rec_id 对应，对档案文档的 Text 层文本内容及其元数据等相关信息建立永久联系，形成数据包。

基于节约成本和利于管理考虑，对双层 PDF 文档进行了格式固化，它的 Image 层是图片格式，与原文件保持一致，可以阅读和打印；Text 层支持内容自由复制。为了使系统处理数据方便，我们通过后台程序把上传的其他文本格式的文档也自动转换成双层 PDF 文档。双层 PDF 全文数据库不仅为档案在线编研、数据挖掘、开展档案主动服务等打下了基础，全文数据直接利用还使档案得到保护和永久性保存。

3.2 创建索引

数字化加工后的双层 PDF 文件和数据包通过调用全文检索子系统内核函数建立对应的索引文件，抓取和解析数据。本系统中的创建索引的过程，实际上就是将双层 PDF 文档中的 text 层、文档对应的卷内目录和案卷目录及有关元数据（也可以说是结构化和非结构化数据）信息提取并创建索引文件的过程。目前，主要有 3 种索引技术：签名、后缀数组、倒排文档。Lucene 采用的是倒排文档，倒排文档的性能和效率都非常高，因而被广泛采用。

索引过程可分为 4 个阶段：准备待索引文档和数据；对文档进行语法分析和语言处理形成一系列词（Term）；经过处理形成词典和倒排文档（索引）；通过存储过程将索引写入索引库。具体过程分析如下：

（1）建立索引器 indexWriter，生成 index 对象，把 Document 对象加到索引中来。

（2）建立信息字段对象 Field，描述文档的属性，如文件标题和内容可以用两个 Field 对象分别描述。

（3）建立文档对象 Document，用来描述文档，内容可

以从 DOC、EXCEL、TXT、HTML、XML 等文档及关系型数据库等多种途径获得，一个 Document 对象由多个 Field 对象组成的。也可以把一个 Document 对象看作数据库中的一个记录，而每个 Field 对象就是记录的一个字段。我们通过编写 Object2DocumentUtil. class 类来实现数据对象与 Document 对象的转换。

在文档被索引之前，首先需要对文档内容进行分词处理，这部分工作就是由 Analyzer 类来完成。Analyzer 类是一个抽象类，它有多个实现，针对不同的语言和应用需要可以选择适合的 Analyzer，本系统中采用的是 StandardAnalyzer。Analyzer 把分词后的内容交给 IndexWriter 来建立索引。在分词时，如果用来进行索引的文档不是纯文本，先使用 OCR 或者其它技术转换成纯文本才能再进行操作。值得注意的是，同一索引，用来建立索引与查询的分词器必须是同一个，才能保证得到正确的查询结果。

(4) 将 Field 添加到 Document 里面，再将 Document 添加到 IndexWriter 里面。

(5) 优化 indexWriter 对象，Directory 类代表了本系统索引的存储位置，它是一个抽象类，有两个实现：一个是 FSDirectory，它表示一个存储在文件系统中的索引的位置；其次是 RAMDirectory，它表示一个存储在内存当中的索引的位置。

创建索引的方法如下：

```
public void createIndexWriter( String sDir) { try{
boolean flag = true; //标记是否重新建立索引,true 为重新建立索引,false 表示增量索引
File indexDir = new File( sDir) ; //索引文件存放目录
Directory dir = SimpleFSDirectory. open( indexDir) ; // 创建 Directory
Analyzer sAnalyzer = new StandardAnalyzer( Version. LUCENE_30) ; //分词器
indexWriter = new IndexWriter( dir,sAnalyzer,flag,MaxFieldLength. UNLIMITED) ; //索引工厂
} catch( Exception e) { logger. error( "indexWriter Exception:",e) ; } }
```

索引创建成功后便生成索引文件，一个索引 (index) 存放在一个文件夹中，比如文书类的行政档案卷内文件索引为 E: /index/T319。基于 lucene 技术的全文检索，根据不同的配置会产生不同的文件，本系统中的行政卷内文件索引如表 1 所示。

_0. cfx 复合文件，当 compound 启用时，被多个段 (Segment) 共享的单独文件集添加进独立的 compound 文件，扩展名为. cfx；

_2. fnm 保存域 (Field) 的元数据信息，一个段包含多个域，每个域都有元数据信息；

_2. frq 词语频率数据文件，记录了词语所在文档的文档列表 (docID) 和应该词语出现在文档中的频率信息；

表 1 全文索引文件

名 称	大 小	类 型	修改日期	属性
_0. cfx	45 644KB	CFX 文件	2013 - 4 - 20 5: 01	A
_2. fnm	1KB	FNM 文件	2013 - 4 - 20 5: 01	A
_2. frq	6 328KB	FRQ 文件	2013 - 4 - 20 5: 01	A
_2. nrm	106KB	NRM 文件	2013 - 4 - 20 5: 01	A
_2. prx	10 383KB	PRX 文件	2013 - 4 - 20 5: 01	A
_2. tii	14KB	TII 文件	2013 - 4 - 20 5: 01	A
_2. tis	731KB	TIS 文件	2013 - 4 - 20 5: 01	A
Segments. gen	1KB	GEN 文件	2013 - 4 - 20 5: 01	A
Segments_2	1KB	文件	2013 - 4 - 20 5: 01	A

_2. nrm 标准化因子文件，计算 Term 权重值，Term 在文档中的出现次数和 Term 的普通程度都影响到权重值。为检索出的文档按相关性排序提供基础。

_2. prx 词语位置数据文件，记录了每一个 Term 出现在所有文档中的位置信息；

_2. tii 词典索引文件，_2. tis 词典文件；

Segments. gen、Segments_2 段的元数据文件，保存了段的属性信息。一个索引可包含多个段，段与段之间独立，添加新文档可生成新的段，不同的段可以合并。

3.3 索引管理

查看索引，读取指定路径索引中是否存在；索引中包含的文档，词条情况，是否需经过优化等；最后一次修改的时间，路径信息，含有的文档数目等；读取索引词条相关基本信息。

删除索引，删除指定序号的文档之后，自动删除对应的索引文件，编写方法 delete(IndexData indexData) { } 来实现；恢复被删除的文档及索引。

更新索引，更新索引中的某个文档；索引同步处理，用户可根据需要自己定制创建索引时间，可定时或实时更新。增量索引、保存索引、修改索引分别编写方法 incrementIndexWriter(String sDir) { } 、save(IndexData indexData) { } 、update(IndexData) { } 调用 lucene 相应的类，在更新索引时，采用方法 closeIndexWriter ( ) 来关闭 IndexWriter。特别是在 update 方法中采用 indexWriter. updateDocument (term,Object2DocumentUtil. object2document (indexData)) 来实现，当数据量很大时，采用“删除再创建”效率更高，updateDocument 等价于 delete (indexData) + save (indexData) 。

3.4 检索过程及结果处理

全文检索在程序内部实际上是一个复杂的过程，通过分析，可总结为以下步骤：用户输入查询语句；词法分析和语言处理；搜索索引，得到符合条件的文档；对结果的相关性进行排序；将查询结果返回给用户界面。

采用计分器 QueryScorer qs 来记录结果的相关性（权重值），根据权重值大小在界面上进行排序；采用 Lucene 处理关键字高亮显示；Highlighter 利用段划分器 Fragmenter 将原始文本分割成多个片段，片段默认的大小为 100 个字符，将包含检索词的片段显示在检索结果中，便于用户浏览查看选择。系统还需进行特殊字符过滤、多重排序、结果分页等处理。

3.5 原文浏览

通过检索过程，在用户界面得到了查询结果。接下来，需要浏览 PDF 原文，并查出检索词在原文中的具体位置。我们使用 Acrobat Reader，结合档案管理系统，来实现检索词在原文中的自动定位。Reader 软件本身对双层 PDF 文档的查找、文本复制、双层切换等功能都提供了支持，“搜索”窗口允许在多个 PDF 查找项目。

在全文检索页面，浏览 PDF 全文是通过在页面内嵌套 PDF 控件的方式实现。通过程序传递参数给 PDF 控件，实现检索词在文档中的定位。PDF 控件代码:

```
<OBJECT id = ´PDF´classid = ´clsid: *****´border = ´0´
        WIDTH = ´100% ´height = ´100% ´ >
        < param name = ´_Version´value = ´65539´ >
        < PARAM name = ´_ExtentX´value = ´20108´
        >
        < PARAM name = ´_ExtentY´value = ´10866´
        >
        < PARAM name = ´_StockProps´value = ´0´ >
        < PARAM name = ´SRC´value = ″ < % = read-
        Path% > ″ >
```

而在档案管理系统内部，案卷和卷内目录链接的全文，需要点击链接，通过管理系统内嵌的阅读器来打开，与全文检索页面的实现有些区别。

4 一站式智能检索设计

档案管理系统必须具备专业检索和一站式智能检索等检索途径，专业检索提供更为复杂的逻辑表达式组配，适合档案人员处理复杂用户需求时使用；而一站式检索带来的是便捷的用户体验，档案用户不必了解具体的档案分类和细节，通过一个检索入口便可以获得所需的信息。

包含全文检索的一站式检索具备异构档案资源库和分布式资源库处理能力，对结构化与非结构化信息合并管理，对目录数据和原文必须进行同步索引。首先通过 JDBC（Java Data Base Connectivity）连接数据库找到要索引的门类，通过卷内文件目录和案卷目录的 ID 号（Rec_id），查找原文关系表中的 Rec_id，原文表中的这条记录有文件存放路径（File_path）等信息，然后根据信息找到对应的原文（双层 PDF 文档），这样便可以对目录数据和原文进行同步索引。接下来，指定生成 Index 目录。而在检索时，只需要对索引进行访问，便可以很快的在各类档案目录库和全文库检索到用户需要的信息。

5 实现效果

通过测试，系统自动从索引中检索出相关的信息，如果检索词包含在文档中，系统还使检索词在文档中自动定位，免去翻页查找的麻烦。若要缩小检索范围，只需要再增加检索词，检索词之间默认为逻辑“AND”的关系，检索结果按相关度排序，根据文档片段值的大小，将包含检索词的文档片段内容显示在检索结果界面，供用户浏览。

运行表明，基于双层 PDF 文档技术的一站式全文检索，提高了工作效率。通过对跨数据表，跨数据类型，案卷、卷内目录数据和双层 PDF 的 Text 层同步索引，查询时访问索引而不访问数据库，有效减轻数据库和系统的压力。系统可以支持1 000万级的数据，毫秒级的响应时间，每秒 500 人的并发访问；可以适应不同的操作系统平台，支持多种数据库接口；具备通用搜索引擎的构架和功能，用户可任意输入检索信息，可多关键字、关键词组合搜索。

全文检索是档案管理系统中很重要的检索途径，弥补了目录检索的不足，也解决了目录著录不全、不规范等问题，大幅度提高了查全率。全文检索无须编制任何检索目录，完全实现智能化、高效率检索，极大地提高了工作效率。虽然不同的档案管理系统可能会采用不同的编程语言和技术架构来实现，对 Lucene 规范中的技术取舍、采用和配置各有不同，但遵循 Lucence 架构的双层 PDF 全文检索的总体实现思想大同小异。双层 PDF 全文数据库为档案编研和数据挖掘提供了资源[6]；也为档案信息聚合（RSS）的研究、定向主动的档案信息服务研究或者更深层次的档案服务成为可能。

参 考 文 献

[1] 管建和，甘剑峰. 基于 Lucene 全文检索引擎的应用研究与实现 [J]. 计算机工程与设计，2007，(2)：489－491.

[2] forfuture1978. Lucene 学习总结之一：全文检索的基本原理 [EB/OL]. http://forfuture1978.iteye.com/.

[3] 胡长春. 基于 Lucene 的中文自然语言搜索引擎 [D]. 上海：上海交通大学，2009：32－35.

[4] 解鹏飞. Lucene 搜索引擎技术在国家海洋数字档案馆示范系统中的实现及应用 [J]. 海洋环境科学，2008，(8)：117－121.

[5] yingsuixindong. 全文检索引擎 Lucene 优点 [EB/OL]. http://blog.csdn.net/yingsuixindong/article/details/5580983.

[6] 向禹. 基于 SOA 架构的高校档案资源管理系统设计与实现 [D]. 长沙：中南大学，2013：61－67.

（本文责任编辑：马 卓）