实验二

网页索引与检索

目录

- **倒排索引**
  - 网页预处理及正文提取
  - 中文分词
  - 停用词处理
  - 网页索引
- 检索系统
- 实验提交内容

<!-- [published at 2016-04-27 04:15:29] -->
<!-- LLTJ_MT:name ="综合" -->

<html>
<head>
  <meta charset="utf-8"/>
<meta name="sudameta" content="urlpath:s/; allCIDs:56145,257,51895,51924,56264,258,38790">
<title>中国船王后人获日2亿赔偿 家族爆发巨款争夺战|中国船王_新浪新闻</title>
<meta name="keywords" content="中国船王" />
<meta name="tags" content="中国船王" />
<meta name="description" content="参考消息网4月27日报道英媒称，2014年一项具有划时代意义的战争赔偿判决，迫使商船三井公司向一
执行，就因为一个最平常的原因：围绕如何分割这笔赔偿产生的家族争斗。据英国《金融时报》网" />
<link rel="mask-icon" sizes="any" href="http://www.sina.com.cn/favicon.svg" color="red">
<meta property="og:type" content="news" />
<meta property="og:title" content="中国船王后人获日2亿赔偿 家族爆发巨款争夺战" />
<meta property="og:description" content="中国船王后人获日2亿赔偿 家族爆发巨款争夺战" />
<meta property="og:url" content="http://news.sina.com.cn/s/wh/2016-04-27/doc-ifxrpvcy4522948.shtml" />
<meta property="og:image" content="http://n.sinaimg.cn/news/transform/20160427/vSpW-fxrpvcy4522918.jpg" />
<meta name="weibo: article:create_at" content="2016-04-27 04:15:28" />

<meta property="article:published_time" content="2016-04-27T04:15:28+08:00" />

2016年04月27日04:15 综合

552
553
554
555 陈顺通后代（图：英国《金融时报》网站）
556
557
558 参考消息网4月27日报道 英媒称，2014年一项具有划时代意义的战争赔偿判决，迫使商船三井公司向一名中国船王的后人支付2亿余元人民币。但这项判决至今却仍未能执行，就因为一个最平常的原因：围绕如何分割这笔赔偿产生的家族争斗。
559 据英国《金融时报》网站4月26日报道，二战期间，陈顺通的船被日本帝国海军征用后全部损失。这场官司在东京、上海两地的法院打了70年，最后以中国船王陈顺通的后代获得40亿日元（约2.34亿元人民币）赔偿告终，没想到家族纠纷却随之而来。
560 报道称，诉讼胜利并没有带来欢呼，反而在越来越多的陈氏后人及债权人中间播下了争执的种子。要求分割赔偿的起诉者包括一名债权人，此人声称陈顺通的孙子欠自己钱（此案已被驳回），另一人则声称自己是陈顺通的非婚生孙子。
561 同时，留在中国内地的陈氏家族分支威胁要质疑陈顺通几十年前留下的遗嘱。根据这份遗嘱，本案所有赔款均交由陈顺通长子的定居香港的后人。内地陈氏后人认为，这违反了女性后代平等分配财产的内地法律。
562 陈顺通曾孙陈中威说："我们尊重法律程序。我们之前是原告，现在成了被告。我们赢了这件案子后，我们依法走程序，但这些人出现了。"
563 报道称，20年前在上海，陈顺通的遗嘱曾成功顶住质疑，但内地的陈氏家族分支正争取依据中国《继承法》重启本案。
564 内地陈氏后人代理律师、上海市联合律师事务所的江宪在谈及应适用什么法律时表示："我认为这是一个非常有意思的法律问题。"是应该适用1949年以前的中华民国法律，还是现行的中华人民共和国法律？如果依照后者，家产应平等分配给男女继承人。
565 陈顺通的小儿子陈乾康从小在上海长大，他说："因为中国强大了，我父亲租给日本人的船才最终获得赔偿。我相信现在更强大的法律体系也意味着，赔偿可以被平等分配。"
566 报道称，二战结束后，中国等国家在与日本实现邦交正常化时，放弃了赔偿以换取日方的援助。在亚洲各国法院，遭遇入侵日军人身虐待的个人寻求赔偿时，几乎没有成功的先例。陈家的案子是个例外，因为在技术层面上，这是围绕20世纪30年代租船合同的条款发生的一起商业纠纷。
567 一些帮助其他中国公民寻求日本企业赔偿的活动人士和律师表示，陈家后人的赔偿争夺战与中威轮船公司最初的索赔一样不寻常。他们表示，其他索赔案件要求的金额要少得多，而且那些年长的原告们也没有这么多后代。
568 维权人士童增在20世纪90年代初首次提议由个人争取索赔，当时在中国引起轰动，他还曾在陈家诉商船三井案中提供建议。他表示，陈家的这场纠纷"令人遗憾"。
569 童增说："我很高兴能在对日诉讼案件中起到帮助作用。但我无法帮助他们解决赔偿分割问题。"

网页预处理及正

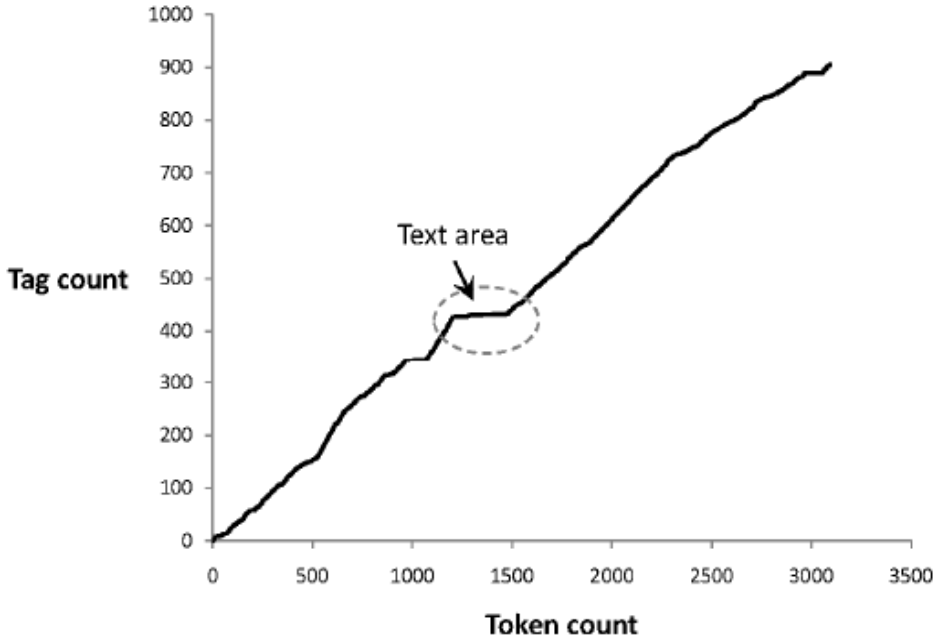
- 网页去噪
  - 什么是噪声？
  - 相关链接与评论是
- 噪声去除的方法
  - 鸵鸟算法，不去管
  - 根据特定网站规律，通用）
  - 文本密度的方法（
  - 模板识别（高级）
  - 自行设计算法去除

网页预处理及正文提取

- 找出规律，去除噪声

网页预处理及正文提取

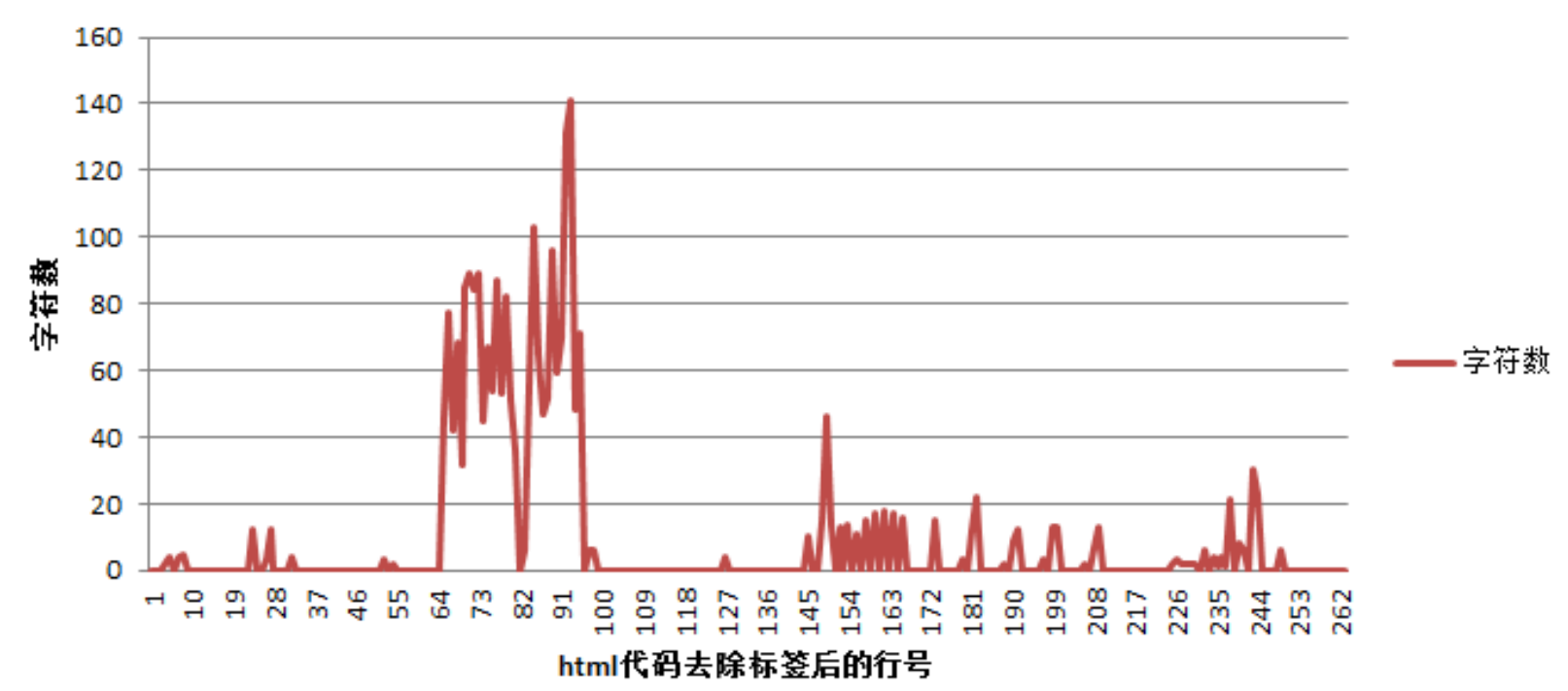
- 文本密度方法



图中的“平原”区域一般是主要正文区

网页预处理及正文提取

- 文本密度方法



网页预处理及正文提取

- Jsoup
- url:
  - 示例

- 策略：
  - 获取<body>标签及其内容
  - 移除其中超链接<a>
  - 移除标签，保留标签文本
  - 余下内容作为正文

```
InputStream input = null;
try{
    URL url = new URL("http://news.scut.edu.cn/s/22/t/3/72/1f/info29215.htm");
    HttpURLConnection con = (HttpURLConnection)url.openConnection();
    con.setConnectTimeout(5000);
    con.setReadTimeout(10000);
    con.connect();
    int code = con.getResponseCode();
    if(code==200){
        input = con.getInputStream();
        Document doc = Jsoup.parse(input,"utf-8","");
        Element body = doc.select("body").first();
        body.select("a").remove();
        System.out.println(body.text());
    }else{
        System.out.println("?");
    }
} catch (Exception e){
    e.printStackTrace();
} finally{
    if(input!=null){
        try {
            input.close();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

网页预处理及正文提取

- Boilerpipe
  - 通过训练获得一个分类器来提取出我们需要的信息（如正文、发布时间等）

- 参考：
  - boilerpipe

```
InputStream input = null;
try{
    URL url = new URL("http://news.scut.edu.cn/s/22/t/3/72/1f/info29215.htm");
    HttpURLConnection con = (HttpURLConnection)url.openConnection();
    con.setConnectTimeout(5000);
    con.setReadTimeout(10000);
    con.connect();
    int code = con.getResponseCode();
    if(code==200){
        input = con.getInputStream();
        BoilerpipeExtractor extractor = CommonExtractors.ARTICLE_EXTRACTOR;
        TextDocument textDoc = new BoilerpipeSAXInput(
                new InputSource(input)).getTextDocument();
        extractor.process(textDoc);
        System.out.println(textDoc.getTitle());
        System.out.println(textDoc.getContent());
    }else{
        System.out.println("?");
    }
} catch (Exception e){
    e.printStackTrace();
} finally{
    if(input!=null){
        try {
            input.close();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

中文分词处理

- 两种方式：自动分词与词典分词
- 自动分词：简单，速度快，但精度不够
- 词典分词：复杂，学习规则众多，速度慢，但精度可以控制

中文分词处理

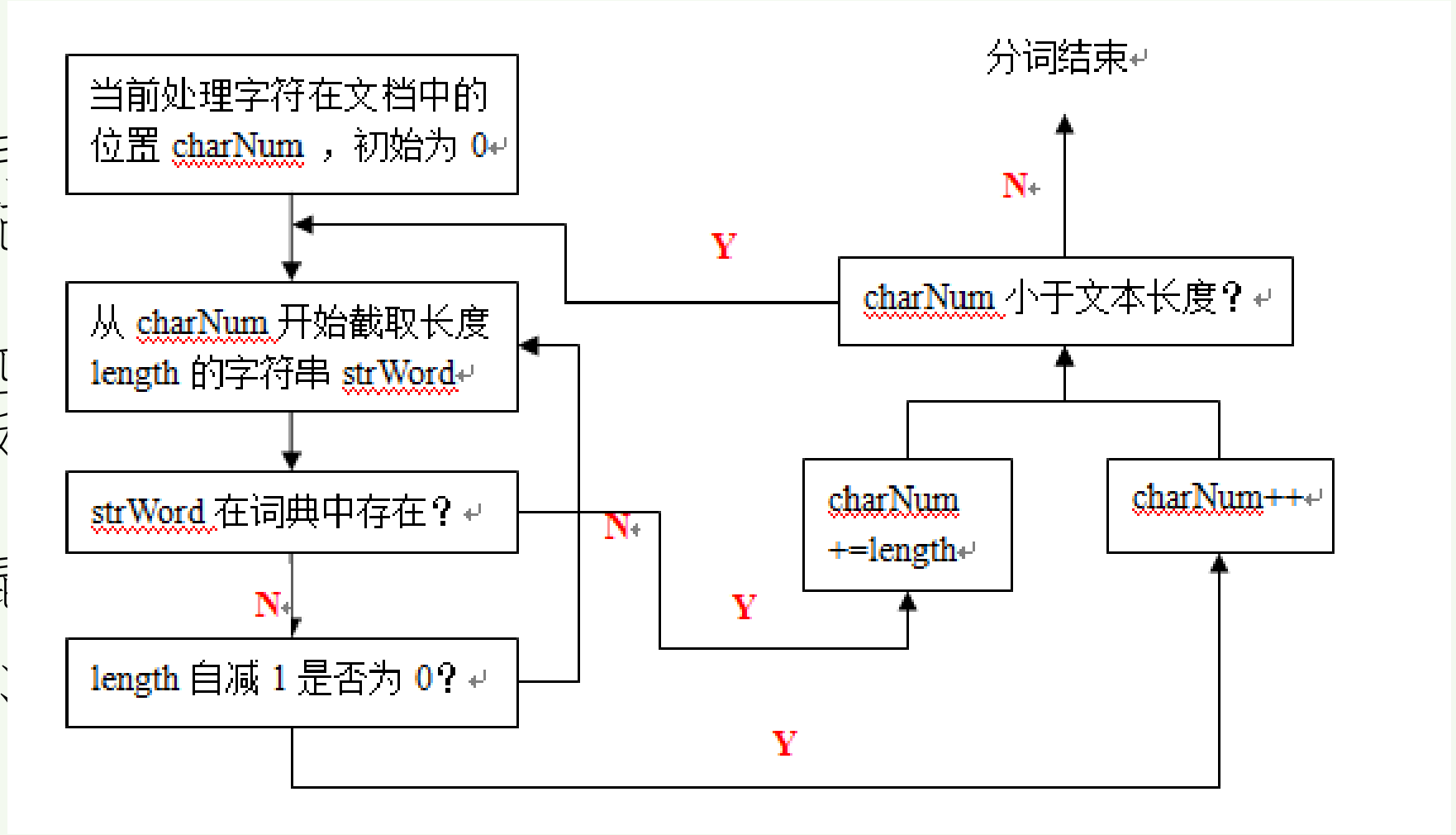
- 自动分词（例子：Lucene + CJKAnalyzer）

```
import org.apache.lucene.analysis.cjk.CJKAnalyzer; //需要下载这个lib
…
IndexWriter  writer = new IndexWriter(indexPath, new CJKAnalyzer(), true);
//很简单，直接new一 个CJKAnalyzer对象就可以了
writer.addDocument(doc);
….
```

- 如：黎明是好学生
- 结果：黎明/明是/是好/好学/学生

中文分词处理

- 词典分词
  - 主要是利用字符串匹配有：
  - 最大正向匹配，最大反向匹配
  - 上述算法混
- 自行设计算



中文分词处理（词典分词）

- JAVA开源分词工具
  - 中科院开发的ICTCLAS
  - IKAnalyzer

- python开源分词工具
  - jieba
  - 北大开源的pkuseg

```
Public statci void main( string[] args)
{
    ICTCLAS instance = ICTCLAS.getInstance();
    String sentence = “网络信息检索是计算机学院的一门课程” ;
    system.out.println( instance.paragraphProcess(sentence));
}
```

- 效果：网络/信息/检索/是/计算机 学院/的/一/门/课程

```
title_seg=list(jieba.cut(title))
content_seg=list(jieba.cut(content))
```

停用词处理

- 需要“停止词典”的支持，设计一个简单的过滤器即可
- 对于类似“的士”、“也门”这些特殊的词，需要“特例词表”的支持，这样就不会被停止词典所过滤。
- 这类词表可以去网上下载。
- 常用的有哈工大停用词表

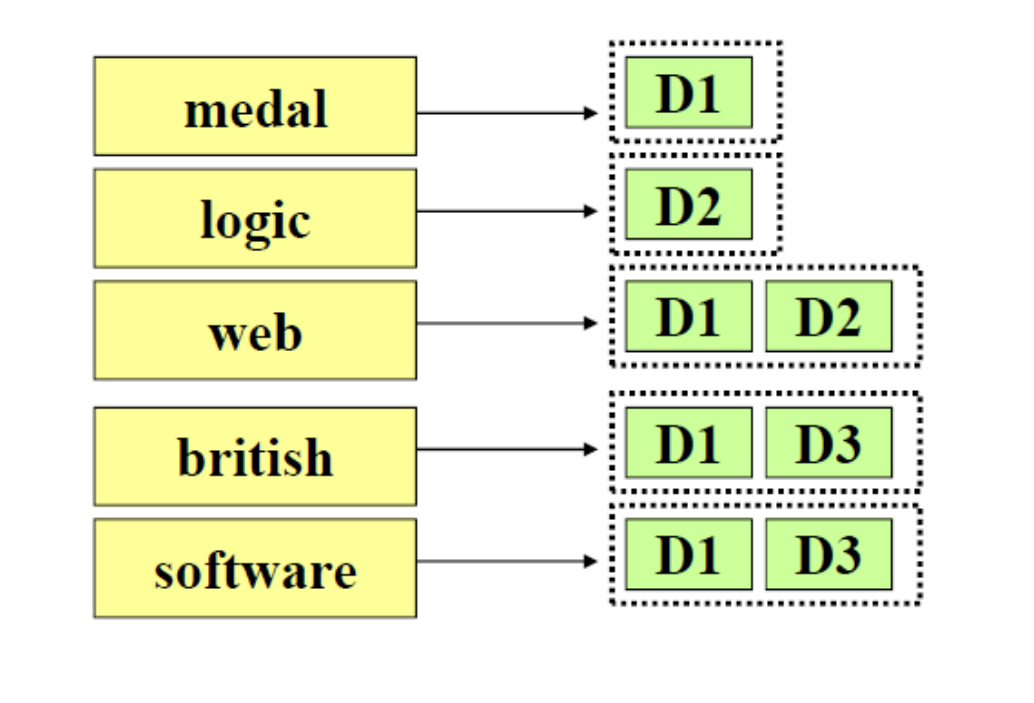
网页索引

- 为什么建立索引？
  避免大量的字符串匹配，同时配合检索策略。

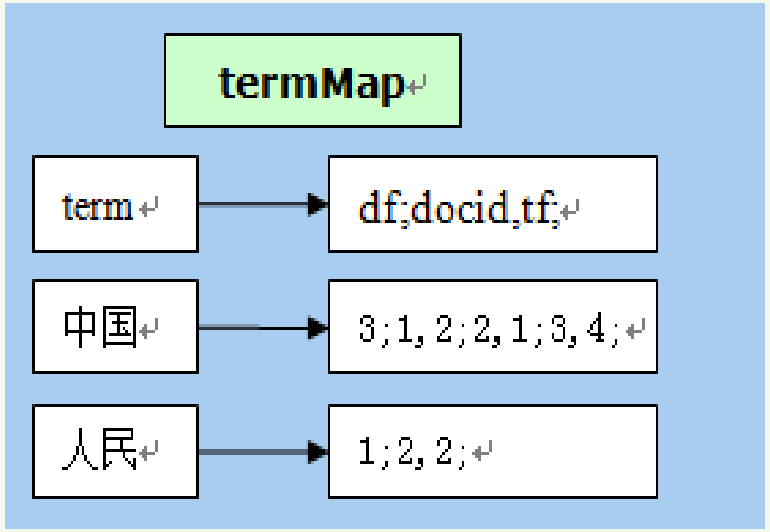
- 索引的格式和可用性，影响检索的速度和质量

网页索引

- 倒排索引



- 自行设计索引存储结构（推荐）
- 如：



- 即（key, value）键值对，key是索引词，value是（文档频率；文档号，词频；文档号，词频；……）

网页索引

- 将文档用词袋模型表示，每个文档表示为词和词频；
- 根据所有文档建立索引
- 索引存储结构可为：
- 词，文档频率，DocID1：词频1，DocID2：词频2，DocID3：词频3。。。。

华南理工大学	20	0:0.175	1:0.7	25:0.0014563106796116503	35:0.032608695652		
招生	4	0:0.175	35:0.006521739130434782	53:0.0012482662968099861	149:0.001		
工作	6	0:0.175	35:0.013043478260869565	53:0.0020804438280166435	59:0.0054		
办公室	5	0:0.175	35:0.006521739130434782	53:0.010402219140083217	59:0.00363636		
学校	15	7:0.35	8:0.35	9:0.35	10:0.35	11:0.35	12:0.35	25:0.0043689320388349
标识	1	10:0.35					
夕阳	2	82:0.0875	100:0.09999999999999999				
日晷	1	100:0.09999999999999999					

网页索引

- Lucene 5.5 建立网页索引
  - Analyzer — 分词器
    - Analyzer a = new SimpleAnalyzer();
  - Directory — 索引存放目录
    - Directory d = FSDirectory.open(path);
  - IndexWriterConfig — 设置索引时参数
    - IndexWriterConfig config =
      - new IndexWriterConfig(a);
- IndexWriter — 负责创建索引
  - IndexWriter writer =
    - new IndexWriter(d,a);
- Document — 索引和检索的单元
  - Document d = new Document();
- Field — Document中的一个域
  - Field f =
    - new StringField(“url”,url_value,Store.Yes)

网页索引

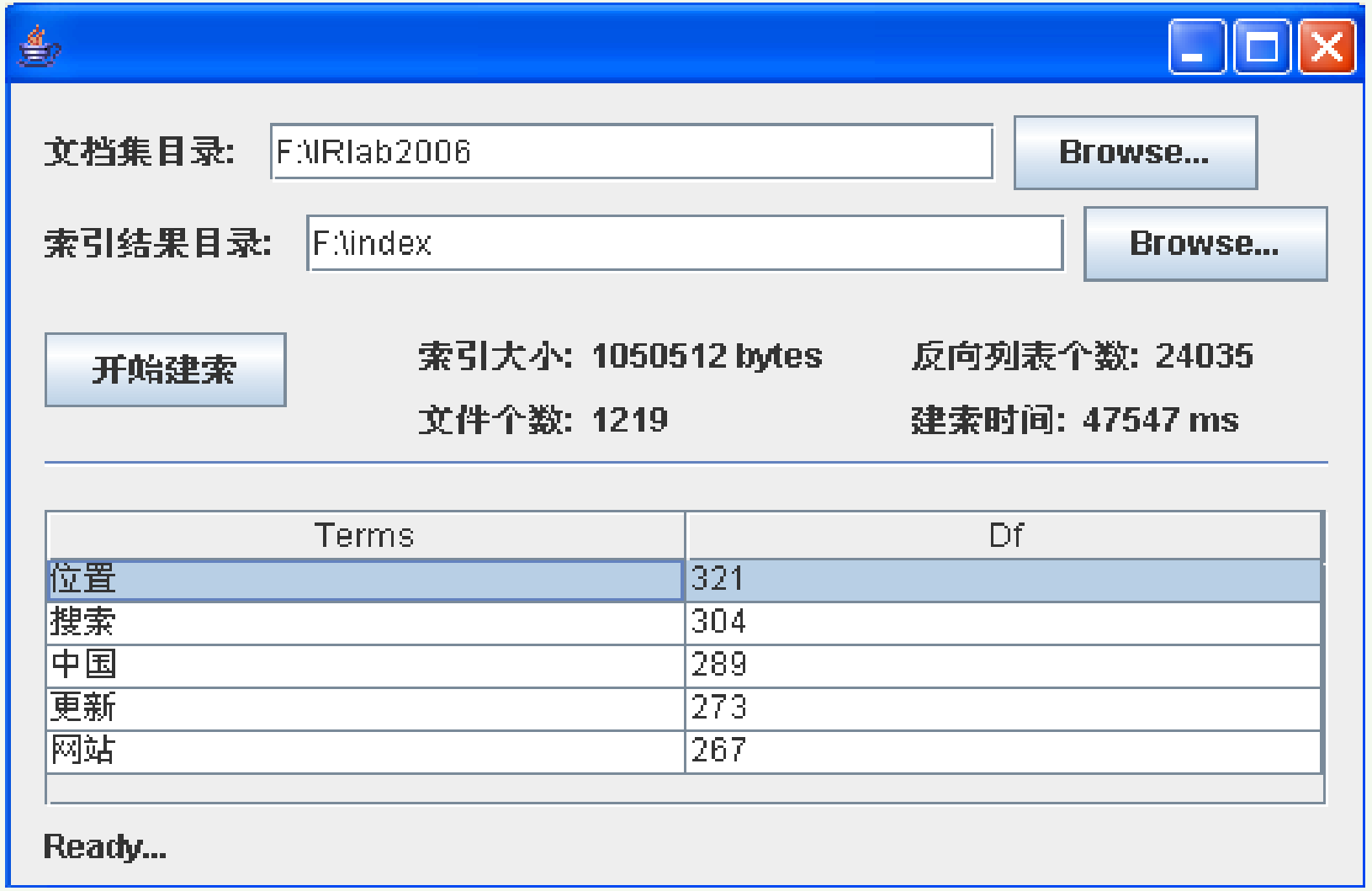
例:
url: http://www.scut.edu.cn/
content: 华南 理工 大学 首页

- WhitespaceAnalyzer
  - 根据“空白”字符作为分隔符分词
- RAMDirectory
  - 索引存于内存
  - FSDirectory.open(Path);
- Field
  - StringField 整串索引、不分词
  - TextField 分词索引
  - LongField、DoubleField、IntField等

```
Analyzer analyzer = new WhitespaceAnalyzer();
IndexWriterConfig config = new IndexWriterConfig(analyzer);
Directory dir = new RAMDirectory();
IndexWriter writer = null;
try {
    writer = new IndexWriter(dir,config);
    Document doc = new Document();
    doc.add(new StringField("url","http://www.scut.edu.cn/",Store.YES));
    doc.add(new TextField("content","华南 理工 大学 首页",Store.YES));
    writer.addDocument(doc);
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
} finally{
    if(writer!=null){
        try {
            writer.close();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

网页索引——期望的程序截图

- 体现以下信息：
  - 创建索引时间
  - 索引大小
  - 词汇表长度



目录

- 倒排索引
- **检索系统**
  - 检索模型
  - 查询表示
  - 网页检索
- 实验提交内容

检索模型

- 课本上第二章介绍的三大模型：
  - 布尔模型
  - 向量模型（TF-IDF法）
  - 概率模型

- 推荐优先考虑实现向量模型（这也是考试重点）

网页检索

- QueryParser

  分词，去停用词，计算每个索引词的tf-idf值；
- IndexSearcher

  从索引中找到索引词对应的文档，计算文档的tf-idf权重；
- 相似度量

  内积或者余弦计算相似度，选择top10返回。

python网页检索

```python
def __init__(self,query):
    index=self.build_index()
    result=self.search(index,query)
    for k in result:
        doc=k[0]
        score=k[1]
        print(str(doc)+'\t'+str(score))
```

```python
def search(self,index,query):
    query_seg=jieba.cut(query)
    query_seg=self.drop_out(query_seg)
    query_tf=self.get_tf(query_seg)
    query_dict={}
    for word in query_tf:
        doc_tf=index[word]
        df=float(doc_tf[0])
        idf=math.log(self.total/df)
        for i in range(1,len(doc_tf)):
            doc=int(doc_tf[i].split(':')[0])
            tf=float(doc_tf[i].split(':')[-1])
            if doc in query_dict:
                query_dict[doc]=query_dict[doc]+tf*idf*query_tf[word]*idf
            else:
                query_dict[doc]=tf*idf*query_tf[word]*idf
    query_dict=sorted(query_dict.items(), key=lambda d: d[1],reverse=True)
    return query_dict
```

```python
def build_index(self):
    index={}
    files_list={}
    files = os.listdir(self.PATH)
    #词和词频列表表示所有文档
    for file in files:
        if 'txt' not in file:
            continue
        self.total+=1
        doc=int(file.split('.')[0])
        words_dict=self.analy(file)
        files_list[doc]=words_dict
    #根据词构建倒排索引
    for word in self.df_words:
        word_value=[]
        word_value.append(str(self.df_words[word]))
        for i in range(len(files_list)):
            words_dict=files_list[i]
            if word in words_dict:
                word_value.append(str(i)+':'+str(words_dict[word]))
        index[word]=word_value
        self.indexfile.write(word+'\t')
        self.indexfile.write('\t'.join(word_value))
        self.indexfile.write('\n')
    return index
```

Lucene网页检索

QueryParser

解析查询字符串，获得Query对象

QueryParser parser = new(String s, Analyzer a);

查询域不止一个时，可以使用MultiFieldQueryParser

IndexSearcher

执行查询操作，获取命中文档

IndexSearcher searcher = new IndexSearcher(IndexReader reader);

TopDocs docs = searcher.search(Query q, int maxDocs);

```
String s = "华南 大学";
String FIELD = "content";
Analyzer analyzer = new WhitespaceAnalyzer();
QueryParser parser = new QueryParser(FIELD, analyzer);
Directory dir = null;
IndexReader reader = null;
Query query = null;
try {
    query = parser.parse(s);
    dir = FSDirectory.open(Paths.get("/data/index"));
    reader = DirectoryReader.open(dir);
    IndexSearcher searcher = new IndexSearcher(reader);
    TopDocs topDocs = searcher.search(query, 10);
    ScoreDoc[] hits = topDocs.scoreDocs;
    System.out.println("Found" + hits.length + " hits.");
    for(int i=0;i<hits.length;i++){
        int docId = hits[i].doc;
        Document d = searcher.doc(docId);
        System.out.println((i + 1) + ". " + hits[i].score +"\t"+ d.get("url")
                + "\t" + d.get("title"));
    }
} catch (IOException e1) {
    // TODO Auto-generated catch block
    e1.printStackTrace();
} catch (ParseException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
} finally{
    try {
        reader.close();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

网页检索——提交内容

- 对附录文件（IR2019查询词.txt），给出每个查询结果排序以及相似度得分。
  - 格式：

```
<序号>
<URL 相似度得分>
```

```
TD03:校园新闻
http://sites.scut.edu.cn/s/56/t/121/main.jspy                    0.437167
http://www2.scut.edu.cn/s/56/t/121/main.htm                      0.437167
http://www2.scut.edu.cn/s/27/t/3/06/8b/info132747.htm            0.299308
http://news.scut.edu.cn/s/22/t/3/70/0f/info28687.htm             0.271394
http://www2.scut.edu.cn/s/135/t/141/a1/3a/info106810.htm         0.233740
http://news.scut.edu.cn/s/22/t/7/71/fd/info29181.htm             0.233740
http://news.scut.edu.cn/s/22/t/3/51/e7/info20967.htm             0.231391
http://news.scut.edu.cn/s/22/t/3/68/cf/info26831.htm             0.226162
http://news.scut.edu.cn/s/22/t/3/p/59/c/18/list.htm              0.218584
http://mapp.scut.edu.cn/Welcome/Index.html                       0.209064
```

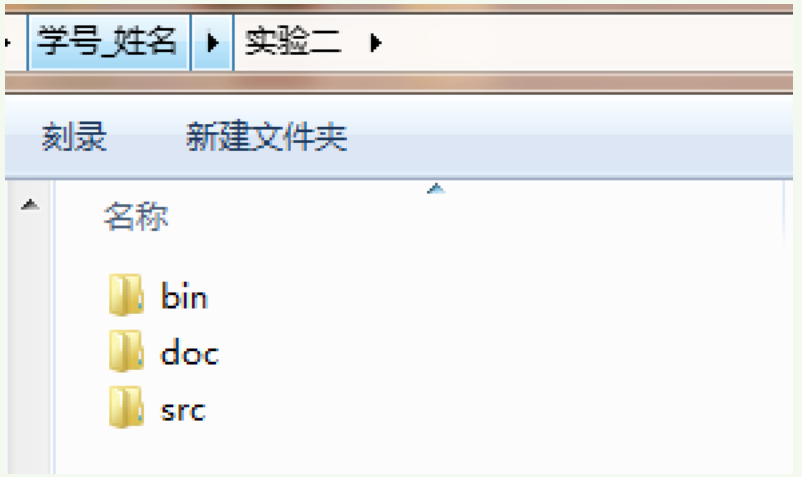
网页检索——查询优化（可选）

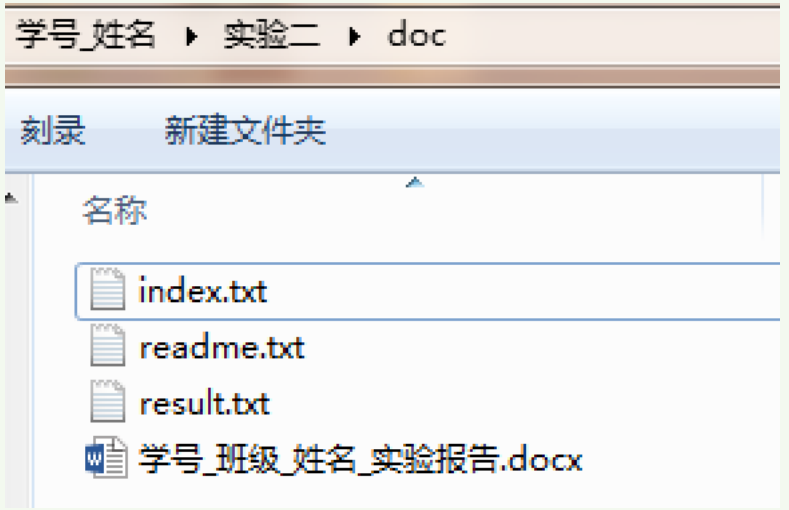
- 采用各种查询处理技术对查询进行优化处理。

- 用评测指标Precision@10和MAP计算系统的检索性能指标，并对所采用的不同技术的应用效果进行比较分析。

实验提交内容

- **程序**
  - 包括源程序和注释、可执行文件以及程序使用说明
- **索引及查询结果文件**
  - 索引相关信息（index.txt）
  - 查询结果汇总（result.txt）
- **实验报告**
  - 说明实验设计的思路，并对实验过程进行分析和总结





THANK YOU!

樊建业
联系方式：1647023764@qq.com
广东省计算机网络重点实验室