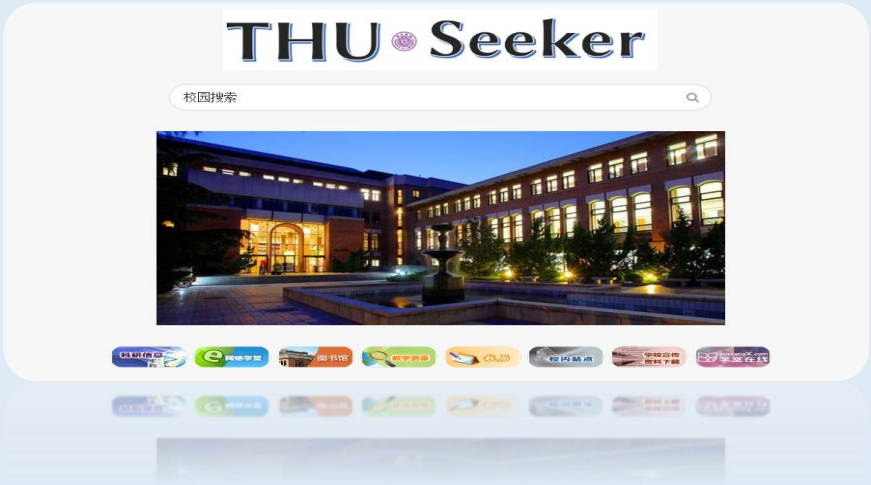
校园搜索实验报告

搜索引擎技术基础课程设计

Tsinghua　Seeker

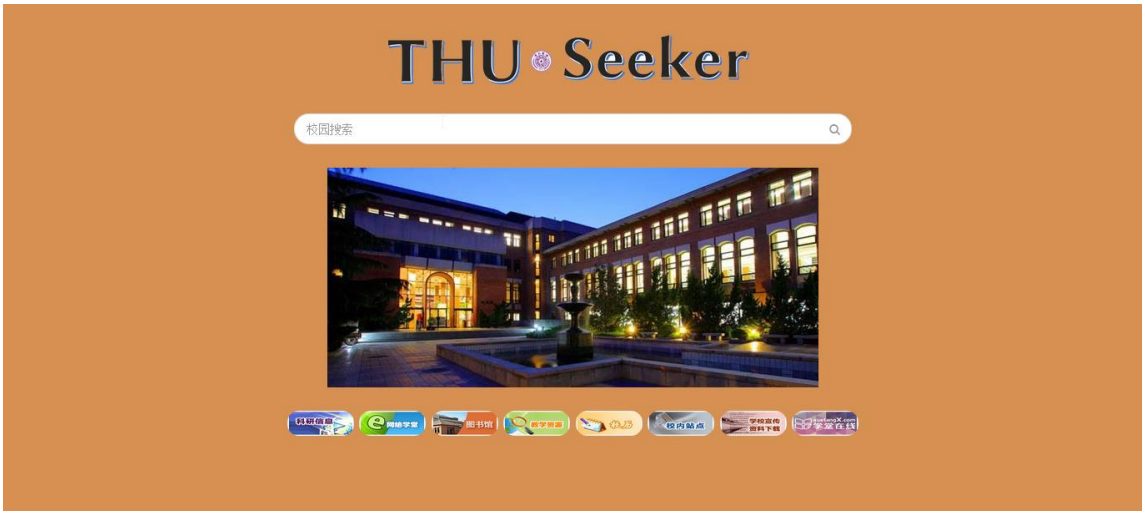
计 22　吴鹏和　2012011274

计 24　杜　鹃　2012011354



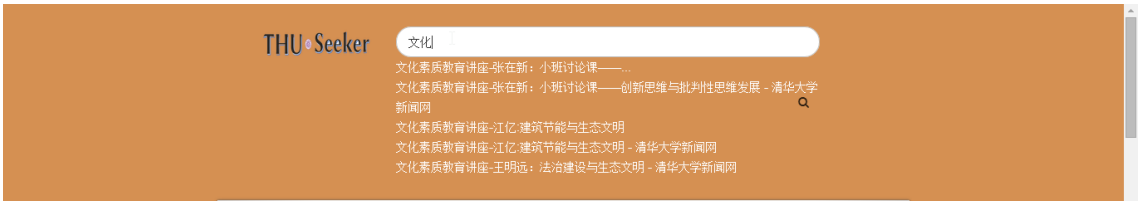
一、 功能介绍

1. 整体界面



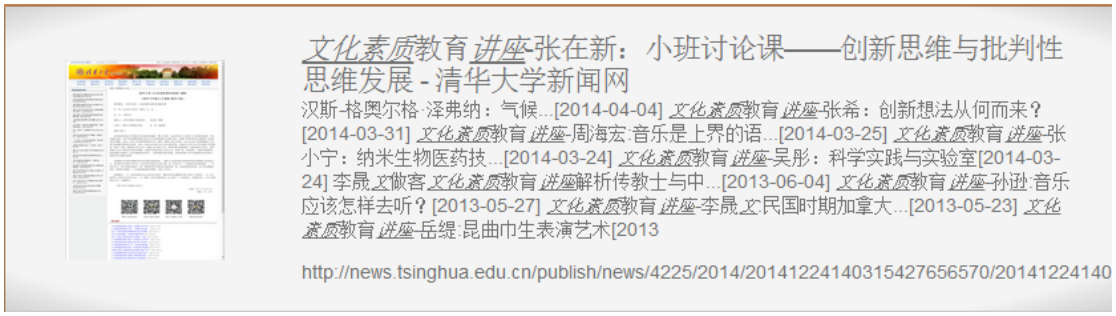
2. 检索建议/自动补全

用户输入查询词时可根据索引内容自动给出建议并提示补全。



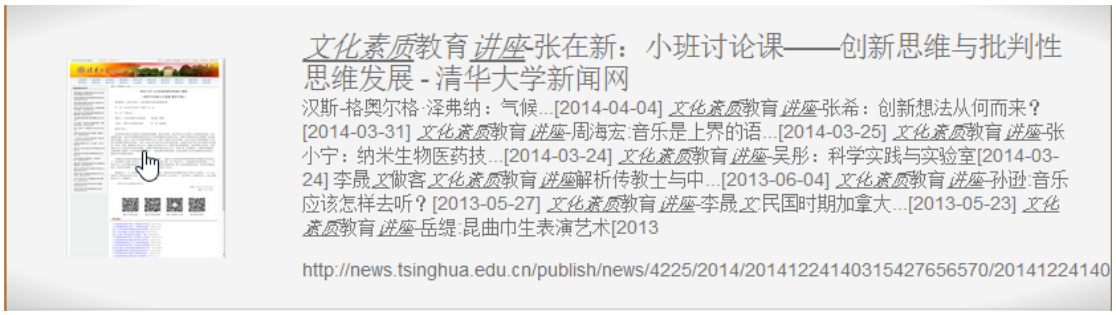
3. 高亮&摘要

对于结果中含全部或部分查询关键词的内容高亮显示（这里采用斜体+下划线）。



4. 网页快照

可在模态框中预览页面截图，并进行上下翻页及链接点击，便捷友好的交互体验。



二、 项目概述

1. 环境说明

系统： Windows 8.1

语言： Java

IDE： Eclipse JavaEE LUNA

项目类型： Dynamic Web Project

运行环境： jdk1.7.0_67

2. 开源工具综述

- Heritrix

  网络爬虫工具，用于抓取原始数据。

- Solr 5.1.0

  Solr 是一个高性能，基于 Lucene 的企业级全文搜索服务器。同时对其进行了扩展，提供了比 Lucene 更为丰富的查询语言，同时实现了可配置、可扩展并对查询性能进行了优化，被 Macy's, EBay, Zappo's 等许多一线大型网站所使用，用于构建索引、查询、高亮、检索建议等。

  官网：http://lucene.apache.org/solr/。

  入门：https://cwiki.apache.org/confluence/display/solr/Getting+Started。

- smartcn-5.1.0

  中文分词器。

- jsoup-1.8.2

  网页解析工具，用于解析 html、jsp 等各类网页。

- pdfbox-app-1.8.9

  pdf 解析工具，用于解析 pdf 文件。

- poi 3.12

  Office 文档解析工具，用于解析包括 doc/docx、xls/xlsx、ppt/pptx 在内的各类 Office 文件。

- org.json

  json 封装与解析工具，用于数据处理与传递。

- url2bmp

  根据 url 生成页面截图，支持命令行调用，用于网页快照。

  主页：http://www.pixel-technology.com/freeware/url2bmp/english/。

  参数：http://www.pixel-technology.com/freeware/url2bmp/english/cl.html。

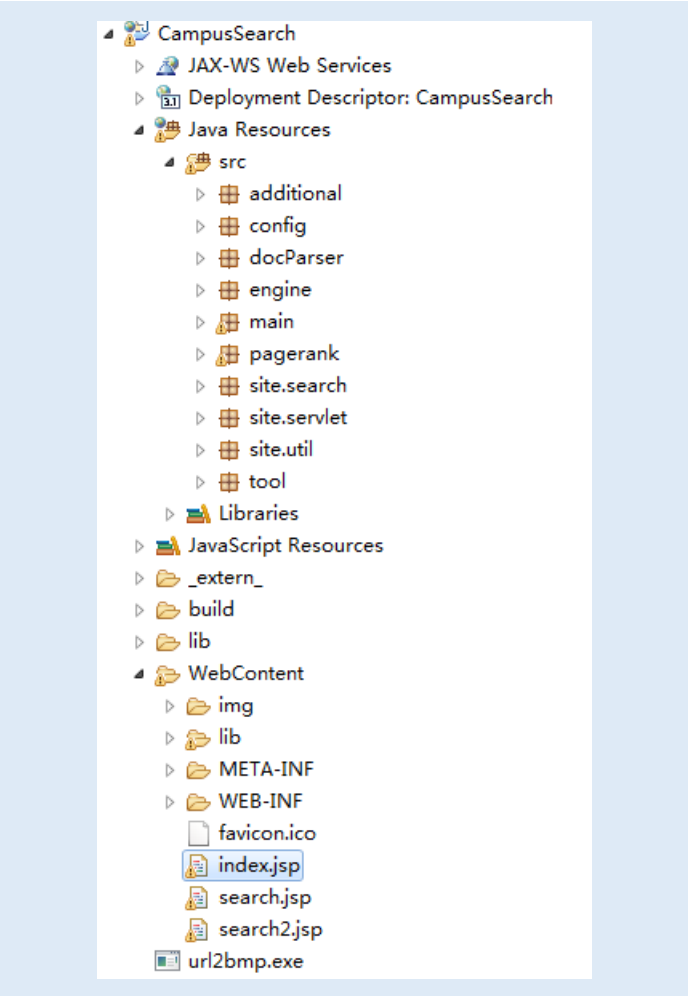
- tomcat 8.0

  网站服务器。

- semantic ui， jQuery 等前台框架与工具。

  用于页面布局与展示。

3. 项目架构



下面是对各个文件夹的作用说明：

-CampusSearch

-_extern_ 外部引用文件

-lib 开源工具包

-doc javadoc

-src 源码

-additional 附加功能

-*.java

-config 配置管理

-*.java

-docParser 网页和各类文档解析

-*.java

-engine 索引与查询

-*.java

-main 整体流程管理与程序入口

-Main.java

-pagerank 链接结构分析

-*.java

-site 网站服务

-search

-servlet

-util

-tool 辅助工具

*.java

-WebContent 前台页面

-*.jsp

详见 javadoc：CampusSearch/doc/index.html。

三、 实现细节

1. 整体流程

1) 使用 Heritriix 抓取原始数据。
2) 对原始网页进行链接解析，提取锚文本和链接拓扑。
3) 根据链接拓扑计算 PageRank。
4) 对原始网页和文档进行文本解析，提取各域信息，并生成对应 json 文件。
5) 根据 url 将各锚文本添加至对应页面 json 文件。
6) 根据 PageRank 得分为各 json 文件赋 boost 值。
7) 启动 solr，根据各 json 文件建立索引。
8) 启动 tomcat。
9) 用户查询与交互。
10) 关闭 tomcat， 关闭 solr。
11) 生成各页面截图，可与上述各步骤并行执行。

2. 资源抓取控制

- 使用 Heritrix 抓取

  抓取清华校内绝大部分网页资源以及大部分在线万维网文本资源（含 M.S. office 文档， pdf 文档等，约 20－ 30 万个文件）。

  在实验中我们以 news.tsinghua.edu.cn 为种子， 设置文件格式/IP 地址过滤器等， 抓取了清华大学新闻网上的资源，大小约为 1.7G。

- 使用正则表达式对 URL 进行过滤

  需要注意的是， PPT 中老师给的正则表达式需要进行修改，应该把“txt/pdf/PDF/doc?/DOC?”从过滤 url 的正则表达式里去掉，因为这些内容也要抓取下来，进行相应处理再建立索引。

3. 网页预处理

- 网页内容编码：

  获取页面中含有”http-equiv=Content-Type”属性的 meta 标签，该标签”content”属性中”charset=”部分的值即为该页面编码。若未找到，则采用默认编码。

  页面示例：

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

  处理过程：

```
static String getCharset (File htmlFile) throws IOException
{
    Document doc= Jsoup.parse(htmlFile, EngineConfig.getDefaultCharset());

    Elements eles = doc.select("meta[http-equiv=Content-Type]");

    if (eles.size() > 0)
    {
        String charsetMeta= eles.get(0).attr("content").toLowerCase();
        int index= charsetMeta.lastIndexOf("charset=") + new String("charset=").length();
        String charset= charsetMeta.substring(index);

        return charset.trim();
    }

    return EngineConfig.getDefaultCharset();
}

html= Jsoup.parse(htmlFile, getCharset(htmlFile));
```

- 无关内容过滤：

  使用 jsoup 的 text()函数，只返回可见内容而不包括 js 脚本、标签、注释等各类无关内容。

```
return html.body().text();
```

- 链接提取：

```
html.getElementsByTag("a"); //获取各链接元素

for(Element ele:anchors){     //链接元素遍历

href = ele.attr("href"); //获取链接地址
text = ele.text();       //获取锚文本
```

4. PageRank 计算

离线计算 **PageRank**，与在线更新结果整合，应用到搜索结果在工程中。我们主要利用之前的 pagerank 链接结构分析小作业的执行文件，因此本次大作业实现过程中主要是分析网页内容，读取和存储为要求的格式。

我们编写了 **File2node** 类，首先将所有的网页遍历一遍，对它们赋值为不同大小的整数，即用<url,int>保存每个网页对应的节点信息。然后利用 CreateGraph 类，再对所有的网页进行遍历，对于每个网页中包含的链接今天提取，保存为<int0: int1, int2, int 3,...> 的格式，其中整数即代表网页的 url 对应的 node 数。这样生成了两个文件，分别为 node.txt 和 graph.txt。最后通过命令行调用 pagerank.exe 进行 pagerank 的计算。

下图列出了 **PageRank** 最高、居中和最低的链接 **url** 地址和对应 **PageRank** 值：

http://news.tsinghua.edu.cn/publish/news/4205/2012/20120817171535504800802/20120817171535504800802_.html	3.23105e-006	3
http://news.tsinghua.edu.cn/publish/news/4205/2013/20130701142448846412859/20130701142448846412859_.html	3.19128e-006	2
http://news.tsinghua.edu.cn/publish/news/4225/2011/20110225230659843235087/20110225230659843235087_.html	3.15585e-006	1
http://news.tsinghua.edu.cn/publish/news/4207/2013/20130318155023528460970/20130318155023528460970_.html	2.18928e-005	116
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110225232438359156266/20110225232438359156266_.html	9.31606e-006	72
http://news.tsinghua.edu.cn/publish/news/4211/2012/20120925103828331652823/20120925103828331652823_.html	3.15546e-006	1
http://news.tsinghua.edu.cn/publish/news/6753/2011/20110225232259921998733/20110225232259921998733_.html	5.41118e-006	38
http://news.tsinghua.edu.cn/publish/news/6682/2011/20110225231556953586687/20110225231556953586687_.html	3.27401e-006	3
http://news.tsinghua.edu.cn/publish/news/4213/2011/20110225232138015541315/20110225232138015541315_.html	4.19483e-006	5
http://news.tsinghua.edu.cn/publish/news/mobile/4204/2011/20110225225343546398150/20110225225343546398150_.html	3.17187e-006	1
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110225231606265827242/20110225231606265827242_.html	4.53775e-006	4
http://news.tsinghua.edu.cn/publish/news/4208/2011/20110225231448734344997/20110225231448734344997_.html	7.36109e-006	29
http://news.tsinghua.edu.cn/publish/news/4207/2014/20140303144106669716426/20140303144106669716426_.html	8.95263e-006	70
http://news.tsinghua.edu.cn/publish/news/mobile/4195/2014/20140508120304890996299/20140508120304890996299_.html	3.17092e-006	1
http://news.tsinghua.edu.cn/publish/news/mobile/4216/2014/20141015164353601109187/20141015164353601109187_.html	0.000238563	1
http://news.tsinghua.edu.cn/publish/news/4207/2014/20140829115318646387024/20140829115318646387024_.html	1.25727e-005	102
http://news.tsinghua.edu.cn/publish/news/6618/2011/20110225231214062362082/20110225231214062362082_.html	3.47169e-006	2
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110720135104699894129/20110720135104699894129_.html	3.9125e-006	13
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110225232311359180334/20110225232311359180334_.html	2.49469e-005	164
http://news.tsinghua.edu.cn/publish/news/4205/2015/20150317143636639757692/20150317143636639757692_.html	3.15547e-006	1
http://news.tsinghua.edu.cn/publish/news/mobile/4225/2011/20110225230713375946384/20110225230713375946384_.html	3.17092e-006	1
http://news.tsinghua.edu.cn/publish/news/4205/2012/20120606105217412734735/20120606105217412734735_.html	4.71793e-006	33
http://news.tsinghua.edu.cn/publish/news/4209/2011/20110225232204859269641/20110225232204859269641_.html	8.04615e-006	13
http://news.tsinghua.edu.cn/publish/news/4207/2013/20131227112823451513746/20131227112823451513746_.html	4.57282e-006	25
http://news.tsinghua.edu.cn/publish/news/4204/2011/20110225225345000882054/20110225225345000882054_.html	3.67884e-006	8
http://news.tsinghua.edu.cn/publish/news/4207/2014/20141105143130927600223/20141105143130927600223_.html	4.3761e-006	19
http://news.tsinghua.edu.cn/publish/news/4210/2011/20110225231405187749361/20110225231405187749361_.html	6.03698e-005	159
http://news.tsinghua.edu.cn/publish/news/4210/2011/20110225231346906134832/20110225231346906134832_.html	4.45133e-006	16
http://news.tsinghua.edu.cn/publish/news/mobile/4210/2015/20150416165331619555556/20150416165331619555556_.html	0.000238563	1
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110225232327515153384/20110225232327515153384_.html	1.69531e-005	113
http://news.tsinghua.edu.cn/publish/news/4205/2011/20110225231526953601915/20110225231526953601915_.html	1.04116e-005	57
http://news.tsinghua.edu.cn/publish/news/4204/2011/20110225225334796601826/20110225225334796601826_.html	7.1683e-006	37
http://news.tsinghua.edu.cn/publish/news/mobile/4210/2011/20110418160725118508736/20110418160725118508736_.html	3.17187e-006	1

5. Pagerank 与搜索过程的结合

将 PageRank 得分乘以 PageRank 总文档数再乘以 10 作为该文档的 boost 值，从而影响检索过程。

6. Anchor 信息重定位

我们在本次作业中还进行了 anchor 信息重定位处理，此处理的目的是将同一个 url 对应的不同文本信息集中到一起。实现此过程的方式是在进行 pagerank 链接结构分析的同时，进行文本的提取和保存，详见 Creategraph.java 文件。

7. Anchor 信息与搜索过程的结合

每个文档增设一个 anchors 域。对每个链接，将其文本添加至其链接地址对应页面的 anchors 域中，anchors 域参与索引。

8. Pdf、office 文档解析

利用 pdfbox 工具和 poi 工具，详见 docParser/PdfParser.java 和 docParser/OfficeParser.java。

9. 中文分词

在 solr 中配置 smartcn 作为中文分词器。见 schema.xml 配置文件。

10. 分域权重

使用 solr 的 DisMax query parser，设置权重为：

title^3.0, anchors^3.0, content^1.0, url^3.0, _text_^1.0。

```
queryObj.put("defType", "dismax");
queryObj.put("qf", "title^3.0+anchors^3.0+content^1.0+url^3.0+_text_^1.0");
```

11. （扩展）检索建议/自动补全

使用 solr 的 suggester 组件提供建议结果：

组件配置见：solr 的 solrconfig.xml 配置文件。

调用方式见：engine.Searcher 中的 suggest 函数。

结果解析见：site.search.SiteSearcher 中的 suggest 函数。

与前台交互：site.servlet.SuggestServlet 中的 doGet 函数。

使用 semantic ui 的 search 组件实现即时补全：

见 WebContent/index.jsp。

12. 高亮&摘要

使用 solr 的 highlight 组件实现结果高亮：

组件配置见：solr 的 solrconfig.xml 配置文件。

调用方式见：engine.Searcher 中的 search 函数。

结果解析见：site.search.SiteSearcher 中的 search 函数。

前台展示见：WebContent/search.jsp。

13. （扩展）网页快照

- 使用 url2bmp 工具进行网页截图，详见：additional.Url2Bmp 各函数。

  为了使得搜索结果更快返回，我们采取离线截图的方式。即每个网页在线下生成对应的网页快照图片，当用户的一个 query 到达时，与搜索到的结果一同返回给前端页面。

- 使用 semantic ui 实现模态浏览及上下翻页，见：WebContent/search.jsp。

四. 实验感想

在这次实验中，我们通过实现建立一个完整的搜索引擎的过程，进一步掌握了搜索引擎的实现过程和原理，包括数据抓取、预处理、建立索引、计算结果、前端设计等步骤。

同时，我们在整个实验过程中，学会了使用很多开源工具。尤其是在开始选择搜索引擎工具时，我们经过网上查询资料，发现除了 lucene 之外，另一种名为 solr 的企业级搜索引擎工具，为了学习和尝试更多工具、得到更多实践经验，我们选择了 solr 工具。在学习如何配置、使用的过程中，我们收获很多。