基于 Elasticsearch 的网盘搜索系统设计与开发*

陈凯旋　林宁

南宁学院信息工程学院，南宁，530200

摘　要　Elasticsearch 既是一款分布式的搜索引擎同时也可以作为数据搜索引擎，开箱即用，非常简单。该系统开发选择以 Elasticsearch 作为搜索引擎去检索用户所需的网盘资源链接。本系统共分为四个功能模块：搜索功能模块、数据爬取模块、用户模块和后台管理模块。搜索模块根据用户需要去检索资源并返还结果，数据爬取模块负责解析包含网盘链接资源的网页，提取符合条件的数据，用户模块用于处理注册用户的个人信息，后台管理模块用于系统服务和用户信息管理，资源数据和日志查看。系统完成后的测试结果，检索速度快，匹配率高，后台数据获取也正常运行，满足使用者的正常需求。

关键字　Elasticsearch，数据爬取，检索资源

Design and Development of Web Disk Search System Based on Elasticsearch

Chen Kaixuan　Lin Ning

School of Information Engineering
Nanning University
Nanning 530200,China;

Abstract—Elasticsearch is a distributed search engine that also works as a data search engine, right out of the box, very simple. The system uses Elasticsearch as a search engine to retrieve the web disk resource links required by users. The system is divided into four function modules: search function module, data crawl module, user module and background management module. Search module according to user needs to retrieve resources and returned as a result, the data crawl module is responsible for parsing and include links to a network backup resource web pages, extract accords with a condition data, the user module used for processing of registered users personal information, background management module for system service and user information management, resource data and log viewer. After the system is completed, the test results show that the retrieval speed is fast, the matching rate is high, and the background data acquisition also runs normally, meeting the normal needs of users.

Keyword—Elasticsearch, Data crawl, Retrieve the resources

1　引　言

随着我国在计算机及网络技术知识方面不断的进步，现代化通讯工具应用的遍及，现代化通讯工具应用的遍及，计算机和其它智能设备在人们的日常生活中扮演着重要的角色并发挥着极其重要的作用[1]。而这些设备在使用的同时会产生大量的应用数据和必须存储的资料文件，根据 IDC 预测，全球数据量将由 2016 年的 16.1ZB 增长至 2025 年的 163.0ZB[2]。这些数据如何处理、传输和存储会是当今数据信息技术的重要组成。因此很多人会将一些学习资料，电影视频，安装包，图片上传到一些网盘上保存起来，也会从一些网盘上去下载我们所需要的文件资源。

获取网盘资源链接的途径有好友分享和博客分享，这些方式比较麻烦，传统网盘搜索系统需要先登录或者关注公众号才能使用搜索服务，这无疑增加用户的使用难度，另一方面就算登录后却只有检索功能，而无法对搜索内容进行收藏或者上传本地文件。所以为此开发了网盘搜索系统；将分散的网盘链接资源整理分类保存起来，用户可以通过接口去查询所需的资源链接，并提供本地资源上传功能；对于用户而言就可以很好的去访问所需的网盘资源，给用户节省很多时间[3]。

网盘搜索系统是根据使用者提供的关键字检索出所需的网盘链接资源。搜索引擎选用 Elasticsearch，是因为其支持分布式集群部署、全文检索、数据分析和

* **基金资助**：本文得到南宁学院 2019 年度教授培育工程项目（2019JSGC12）资助。
通讯作者：林宁，副教授，bgy_2009@163.com

对海量数据近实时处理的特点，而开箱即用配置简单更适用于本次设计开发。

本文针对传统网盘搜索系统，简化搜索方式和新增用户功能模块，实现了基于 Elasticsearch 作为搜索引擎结合 Springboot 框架和 HTTP 协议等技术构建出一个网盘搜索系统。

2 相关技术研究

2.1 Web 服务器

Web 服务器可以部署运行程序，处理使用者的操作请求和存储所需的数据。

本文采用 Tomcat 作为 Web 服务器，Tomcat 是一款 Web 应用服务器，且具有处理 HTML 的功能，将打包好的项目部署到该服务器上只需简单的配置就可以启动项目，非常方便。

2.2 SpringBoot 框架

SpringBoot 不仅有 Spring 框架的优点与特性，而且该框架大大简化了编程中所需的配置环节，应用特定的方式来配置，自动配置和起步依赖是其最长用的，项目的快速搭建，使编码变的简单，部署和监控也很简单。应用 SpringBoot 框架作为项目基础结构可以将更多的精力投入到程序的开发中，不在忙于去文件的配置，相应的也减轻了一些不必要的工作量。

2.3 MySQL 数据库

MySQL 是很流行的一款关系型数据库系统。在语言上支持本次研究所使用的 Java 开发语言同时可以处理上千万的数据量，存储文件也很庞大，体积小和速度快可以存储系统数据和作为永久数据的备份。

2.4 Elasticsearch 技术

Elastic Search 是一款开源的、实时性极高的分布式文本搜索引擎，它使用 Java 语言开发实现的，其底层是开源的 Apache Lucene 库。Elasticsearch 基于 RESTful 协议提供了 HTTP Web 接口规范，并使用 JSON 数据格式实现了文档数据的分布式存储与全文检索。Elasticsearch 具有分布式集群、分片存储、集群容灾与数据恢复和负载均衡等特点[4]。

2.5 Logstash 数据采集

为了能将爬虫模块采集到的数据同步到 Elasticsearch，就需要用到数据定时同步，减轻管理者的工作负担。

Logstash 是数据采集处理引擎，与 Elasticsearh 能很好的配合使用。它支持来自不同数据源的数据采集工作，对数据进行提取和转换，然后存储[6]。

Logstash 原理具体如图 1 所示。

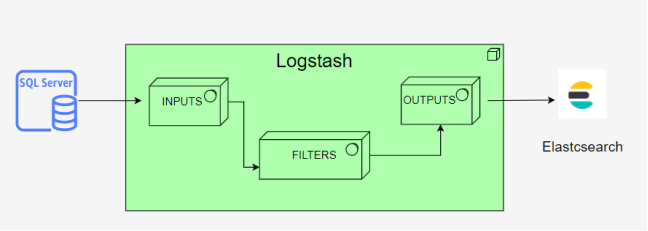

图 1 Logstash 原理具体图

3 网盘搜索系统设计

网盘搜索系统总体架构设计如图 2 所示。

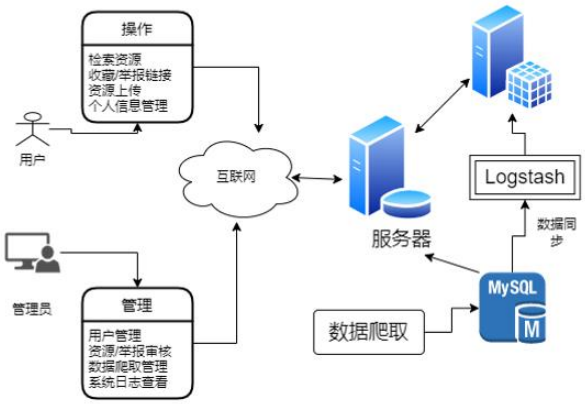

图 2 网盘搜索系统设计图

网盘搜索系统架构图说明:

(1) 用户操作系统可以使用检索功能去检索链接资源，登录后有权去对搜索出的网盘链接进行收藏或者举报，同时也可以上传本地资源到服务器，对自己的个人信息进行查看与编辑。

(2) 管理员主要负责管理该系统，对系统用户信息进行管理，审核使用者上传的资源文件和处理使用者举报的链接，对数据资源库的管理，查看与编辑数据资源，启动和停止数据爬取功能，对系统日志查看。

(3) 数据爬取模块负责从网上爬取网盘链接，将清洗好的数据存入 MySQL 数据库中。

(4) MySQL 数据库用于存储系统结构化的数据和作为链接资源的持久化存储层。

(5) 数据传输，Logstash 定时从 MySQL 中读取更新的数据同步到 Elasticsearch 服务中。

(6) Elasticsearch 作为搜索引擎和数据处理引擎，对使用者提供的关键词进行分词去检索，将各个分片的数据综合传给服务器，进而展示给使用者。

(7) 服务器提供服务，响应使用者的请求以及数据处理和 HTTP 解析。

4 网盘搜索系统实现

该系统由客户端和后台管理系统构成，客户端主要是提供使用者该系统的服务功能，后台管理则负责处理系统的功能业务。系统总体架构可以分为四个功能模块：搜索模块、用户功能模块、爬虫模块、后台管理模块。

4.1 搜索模块

使用者在不登录的情况下就可以进入搜索界面，输入搜索词点击搜索按钮进行资源检索。搜索服务提供了分类查找，可以根据资源的种类搭配搜索词综合查询，也可以不分类去检索资源，该系统默认是不分类检索。用户点击搜索后，Elasticsearch 服务收到用户传来的关键词和分页字段数据，开始构建搜索服务，选择模糊搜索方式，组合查询条件并将查询条件封装给查询对象。

构建搜索服务的实现代码如下：

```
SearchRequest searchRequest = new
 SearchRequest("bdy");
//构建搜索
SearchSourceBuilder sourceBuilder = new
 SearchSourceBuilder();
//分页检索
sourceBuilder.from(page);
sourceBuilder.size(account);
// 使用模糊搜索查询，设置 field 精度
MultiMatchQueryBuilder
 multiMatchQueryBuilder=QueryBuilders.multiMatc
 hQuery(keyword,"titleName").field("titleName",  8);
//组合字段条件查询，
BoolQueryBuilder boolQueryBuilder =
QueryBuilders.boolQuery();
if(!keyword.isEmpty()){
boolQueryBuilder.must(multiMatchQueryBuilder);
// 对关键字过滤查询
if(!category.isEmpty()){  boolQueryBuilder.filter(Qu
 eryBuilders.termQuery("category.keyword",categor
 y));}}else {
 boolQueryBuilder.must(QueryBuilders.termQuery("
 category.keyword", category));}
// 将查询条件封装给查询对象
sourceBuilder.query(boolQueryBuilder);
```

下面的程序段是使用 HighlightBuilder 方法解析搜索词，将搜索词和对搜索词分词的字符进行高亮展示，字体高亮的颜色可以自己定义。

```
//高亮处理过程
HighlightBuilder hbu= new
HighlightBuilder();

//配置高亮字段，这里设置为红色高亮
hbu.field("*");
hbu.requireFieldMatch(false);
hbu.preTags("<span style='color:red'>");
hbu.postTags("</span>");
sourceBuilder.highlighter(hbu);
```

4.2 用户功能模块

用户进入登录注册页面，可以在这里输入个人信息去注册账号，输入正确的账号密码登录到网盘搜索系统。登录后可以对搜索出的链接进行收藏，点击举报填写举报信息就可以举报该链接。进入个人中心可以查看自己的个人信息和收藏信息器，点击通知按钮还可以了解举报的状况。在资源管理中可以查看自己上传的资源信息列表，知道是否都审核通过，可以对自己上传的文件从服务器上下载到本地。同时也提供了本地文件上传功能，使用者选中上传文件类型，上传本地资源，等待管理员审核通过即可公开。

使用者本地文件上传实现代码如下：

```
List<String> mlist = new ArrayList<>();

//判断是否为空文件，防止异常
 if(!mfile.isEmpty()){String
 f_type=mfile.getOriginalFilename().substring(mfile.
 getOriginalFilename().lastIndexOf("."),
 mfile.getOriginalFilename().length());
System.out.println(realPath);String  uuidFileName  =
 UUID.randomUUID().toString().replace("-",  "")  +
 f_type;File files = new File(realPath,
uuidFileName);File newFile = new File(realPath);
if (!newFile.exists()){ newFile.mkdirs();}
// 上传
mfile.transferTo(files);
mlist.add(uuidFileName);
}else {
mlist.add("0");
}
//返回资源名
return mlist;
```

用户点击下载按钮，开始请求服务器该资源，随后资源会通过浏览器进行下载。下载功能代码如下所示：

```
// 读取配置目录和文件名->新的 file
File fl = new File(sysPath, fileName);
      FileInputStream ins= new FileInputStream(fl);
      // 设置响应头、告诉浏览器下载文件
String  attach="attachment;  fileName="  +
 fileName;response.setHeader("content-disposition",
 attach);ServletOutputStream
 ous=response.getOutputStream();
      int flen = 0;
      byte[] dataArr = new byte[2048];
      while ((flen = ins.read(dataArr )) != -1) {
      ous.write(dataArr , 0,f len);
      }
      ous.close();
      ins.close();//关闭
```

4.3 链接数据爬取

数据作为该系统的核心支撑，它的获取就显得非常重要。数据来源于博客网站博主的分享，例如 CSDN 博客网站，初始化一个种子存储在集合中用于爬虫启动初始化页面爬取。网页爬取技术来自 HttpURLConnection 包，为了持续爬取数据，许哟啊设置配置请求头文件，使用代理 IP 池，随机抽取可用 IP，最后在搭配采用随机休眠线程的方式，可以最大化去获取所需的数据。网盘链接资源来源于很多网站，为了快速和高效的利用 CPU 采用多线程方式去爬取，线程统一由基于 ThreadPoolTaskExecutor 的线程池去管理。

线程池配置如下：

```
ThreadPoolTaskExecutor executor = new ThreadPoolTaskExecutor();
executor.setMaxPoolSize(maxPoolSize);
executor.setCorePoolSize(corePoolSize);
executor.setQueueCapacity(queueCapacity);
executor.setKeepAliveSeconds(keepAliveSeconds);
executor.setRejectedExecutionHandler(new ThreadPoolExecutor.CallerRunsPolicy());
```

CSDN 等博客网站爬取到的网页内容，会先判断该网页有无包含的关键字，存在的话会先进行初步提取，并将该网页的拓展链接放入队列中，为了避免多线程下的并发冲突，需要为公共对象和方法加索，此过程会采用 ReentrantLock 为其加锁，ReentrantLock 具有可重入、可中断、可限时、公平锁等特点。下面代码片段使用该技术控制爬虫服务暂停功能。

```
try {
    //csdnurl.size :待爬取的 csdn url
//isCancle:是否取消暂停
    while (csdnurl.size()>0 && !isCancle) {
    if (isPause) {//线程暂停
    csdnCon.await();}
    TimeUnit.SECONDS.sleep(10);
    spiderCSDN(csdnurl.get(0), false);}} catch (InterruptedException e) { e.printStackTrace();
}finally {csdnLock.unlock();
}
```

队列中的链接先进先出保证不会多次爬取操作。提取出的资源名和链接如果两者有一种不一样就定义为全新的一条数据，存进集合中，在数据进入队列中会先判断该集合中是否存在，从而保证相同数据只爬取一次。种子链接也是相同操作去存储。难点在于如何锁定该网页包含所需链接，提取所需数据，提取的数据又如何清洗，并判断种类存入数据库中进行算法设计。

数据爬取的简略说明过程：

(1) 确定爬取的链接地址。

```
URL obj = new URL(url);
```

(2) 解析爬取到的网页字符

```
Document
document=Jsoup.parse(response.toString());
```

(3) 开始模糊提取链接和标题

```
//提取包含 pan.baidu 的内容
    Pattern.compile("(.*)(pan.baidu(.*))");

//判断有无提取码
m.group(2).contains("提取") ||m.group(2).contains("密码")

//获取提取码
m.group(2).split("(提取(码)?|密码)([:：])?");

//换种方式再次匹配,进一步提取数据
Patternpattern=Pattern.compile("(.*)(pan.baidu(.*)(提取码(.*)|密码(.*))?(\\s|\\n|)?)");

//提取内容初步去噪
url=pawd[0].replaceAll(":","/").replaceAll("(提取(码)?|密码)([:：])?","");
strings[1] = "https://"+url.replace(".coms",".com/s/")
    .replace("）","")
    .replace("（","").trim()
    .replaceAll(" ","");
```

4.4 后台管理模块

后台管理功能模块是管理员与系统交互的实现方式。管理员登录获取权限进入系统，可以在首页看到该系统注册用户数，链接数据总量，待处理举报信息通知，与按周、月、年统计的热搜词。

用户管理功能可以管理用户的个人信息，例如查看注册的用户信息列表，了解该账号的状态，新增、更改和删除个人信息，更改账户的权限，设置为系统管理员或者将将管理员设置为普通用户。该功能模块还可以审核用户上传的资源文件，查看资源名称和内容是否违规，判定审核通过或者审核未通过。用户则可以在消息通知栏查看资源审核的结果。

数据管理包含系统的链接数据和操作爬虫模块是否启动的功能。选择该功能项进入数据列表页，可以查看所有的链接数据，也可以搜索查询。管理员有权限在这里手动新增网盘链接信息，也可以修改与删除已存在的网盘链接。管理员进入爬虫模块页有四项按钮供管理员操作：分别是启动、暂停、恢复、结束。点击启动按钮，开始爬取和解析数据。点击暂停按钮，可以暂停线程池中正在运行的爬虫任务等待恢复，如果长时间未恢复则停止爬虫任务，释放资源。恢复按钮被点击后，原先暂停的爬虫任务继续从暂停时继续工作。点击结束按钮，停止全部爬虫任务，释放系统资源。

举报管理功能可以接收用户举报的信息，由管理员做出判断，举报理由符合，删除存在问题的链接，

举报理由证据不足，驳回该举报内容。用户可以在消息栏查看举报结果通知。

日志管理功能，记录管理员对系统的操作，例如登录到该系统，修改用户信息，启动数据爬取等操作。

4.5 Web 服务器搭建

yum 源没有 Nginx 软件，则要使用源码编译安装的方法来搭建 Nginx 服务。编译安装步骤如下：

```
[root@web01~]#vim/etc/zabbix/zabbix_agentd.conf
 98 Server=172.20.1.71
```

(1) 启动服务

```
[root@web01 ~]# systemctl start zabbix-agent
```

(2) 查看服务运行状态。

```
[root@web01 ~]# netstat -lntup|grep 10050
```

5 结束语

本系统基于 Elastisearch 实现，进行网盘搜索系统的实现，使用者可以通过 Web 浏览器进行访问使用，为简化用户使用，采取不登录即可以检索链接资源，登录可享受资源收藏与上传等功能服务。Elastisearch 与 SpringBoot 结合开发的模式，系统服务可以分布在多台服务器，提高搜索效率，减少检索时间。爬虫模块采取了很多反爬虫方式，携带请求头，代理 IP 和线程休眠，来保证系统平稳运行。爬取的数据先保存子啊 Mysql 数据库中，在定时更新 Elastisearch 服务中，一方面保证数据更新，另一方面保证数据安全。

参 考 文 献

[1] 唐庆谊.计算机网络技术发展模式分析[J].大众标准化,2020(24):156-157.

[2] 孙诗军,段元梅.基于 Java 的网盘系统的设计与实现[J].无线互联科技,2022,19(01):60-61.

[3] 唐权,韩文智.基于 SpringMVC 框架文件上传技术应用研究[J].信息通信,2018(11):188-189.

[4] 梁文楷,涂红玲,陈佳欢.一种基于 ElasticSearch 全文检索技术的研究[J].中国科技信息,2021(18):82-84+87.

[5] 柳帆.基于 ElasticSearch 的科技资源检索系统的研究与实现[J].现代计算机,2021,27(26):93-100.

[6] 贾继洋,徐涛,潘文文,李旭宏.海量日志采集可视化平台设计[J].福建电脑,2021,37(12):117-120.DOI:10.16707/j.cnki.fjpc.2021.12.027.

[7] Elasticsearch B.V.; Researchers Submit Patent Application, "Document-Level Attribute-Based Access Control", for Approval (USPTO 20200184090)[J]. Computers, Networks & Communications,2020.

[8] Elasticsearch B.V.; "Shard Splitting" in Patent Application Approval Process (USPTO 20200133550)[J]. Computer Technology Journal,2020.

[9] Multi-index Approach to Search Chinese, Japanese, and Korean Text with Elasticsearch 6.6[J]. INTERNATIONAL CONFERENCE ON FUTURE INFORMATION & COMMUNICATION ENGINEERING,2019,11(1).

[10] 李尚林,陈宫,雷勇.基于 Java 的网络爬虫系统研究与设计[J].新型工业化,2021,11(04):74-77+80.DOI:10.19335/j.cnki.2095-6649.2021.4.029.

[11] 罗恒洋,张林.Java 中的正则表达式应用探讨[J].电脑知识与技术,2019,15(32):95-98.DOI:10.14004/j.cnki.ckt.2019.3807.