李墨珩 2017050025

程序设计训练第三周大作业说明文档

September 15, 2018

程序设计训练第三周大作业

新闻搜索引擎设计

综合运用了 Python3 Jieba BeautifulSoup4 正则表达式 Django HTML CSS Javascript Mysql Bootstrap4 搭建了新闻搜索引擎。

一、数据收集

分为三部分：获取URL、获取全文以及关键字处理。

第一部分：

从新华网的不同板块出发，首先采集新闻链接 url 、新闻标题 title 和新闻摘要 summary ，存入 Mysql 数据库。以自建 id 为关键字（以采集次序自增），在向数据库中加入数据时对 url 查重，确保 url 唯一。

并对在采集过程中可能出现的错误进行了预处理，避免运行时报错，降低效率。

```
db_config ={
    'host': '127.0.0.1',
    'port': 3306,
    'user': 'root',
    'password': 'passMY.18',
    'db': 'pytest',
    'charset': 'utf8'
}
connection = pymysql.connect(**db_config)
oldurls=['http://www.xinhuanet.com/world/index.htm','http://www.xinhuanet.com/overseas/index.htm
'http://www.xinhuanet.com/local/index.htm','http://www.xinhuanet.com/house/index.htm',
'http://www.xinhuanet.com/politics/rs.htm','http://www.xinhuanet.com/tech/index.htm',
'http://www.xinhuanet.com/politics/leaders/index.htm','http://www.xinhuanet.com/politics/xhll.ht
'http://www.xinhuanet.com/food/index.htm','http://www.xinhuanet.com/auto/index.htm',
'http://www.xinhuanet.com/info/index.htm','http://www.xinhuanet.com/health/',
'http://www.xinhuanet.com/travel/','http://www.news.cn/money/index.htm','http://www.xinhuanet.co
'http://www.news.cn/comments/index.htm','http://www.news.cn/fashion/index.htm',
'http://www.xinhuanet.com/legal/ffu.htm','http://www.news.cn/fortune/caiyan.htm','http://www.new
'http://www.news.cn/abroad/index.htm','http://www.news.cn/city/index.htm','http://www.news.cn/zh
'http://www.news.cn/info/spsy/index.htm','http://www.news.cn/xhsd/index.htm',]
urls=['http://www.news.cn/politics/xgc.htm','http://www.news.cn/finance/','http://chanye.news.cn
headers = {'User-Agent':'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Ge
try:
    with connection.cursor() as cursor:
        sql = 'insert into newslinks(title, url, summary) values(%s, %s, %s)'
        for url in urls:
            page = request.Request(url, headers=headers)
            page_info = request.urlopen(page).read().decode('utf-8')
            soup = BeautifulSoup(page_info, 'html.parser')
            alllis = soup.find_all(name='li',attrs={'class':'clearfix'})
            for li in alllis:
                if li.find(name='a') is not None:
                    title = li.find(name='a').string
                    if li.find(name='a').attrs['href'] is None:
                        url='None'
                    else:
                        url = li.find(name='a').attrs['href']
                    if li.find(name='p',attrs={'class':'summary'}) is None:
                        summary = 'None'
                    else:
                        summary = li.find(name='p',attrs={'class':'summary'}).string
                    cursor.execute(sql, (title, url, summary))
                    connection.commit()
            print('finished a website!')
finally:
    connection.close()
print('findlinks.py all finished!')
```

第二部分：

从数据库中收集的 url 出发，利用 BeautifulSoup4 和 正则表达式 逐条寻找每篇新闻的发布时间 date 和新闻全文 text ，并对在采集过程中可能出现的错误进行了预处理，避免运行时报错，降低效率。

```
    cursor.execute(sqlr)
    results=cursor.fetchall()
    for row in results:
        url = row[3]
        page = request.Request(url, headers=headers)
        try:
            page_info = request.urlopen(page).read().decode('utf-8')
        except urllib.error.HTTPError as e:
            print('HTTPError',row[0])
            continue
        except UnicodeDecodeError as e:
            print('UnicodeDecodeError')
            continue

        soup = BeautifulSoup(page_info, 'html.parser')

        text=''
        div = soup.find(name='div',attrs={'id':'p-detail'})
        if div is not None:
            #print(div)
            for p in div.find_all(name='p'):
                if p.string is not None:
                    text+=p.string
                    text+='\n'
            #print(text)


        date=0
        span=soup.find(name='span',attrs={'class':'h-time'})
        if span is not None and span.string is not None:
            datat=span.string
            date=int(datat[1:5])*10000+int(datat[6:8])*100+int(datat[9:11])
        else:
            date=99999999
        #print(date)

        sqlu = 'UPDATE newslinks SET date = %s WHERE (id = %s)'
        cursor.execute(sqlu, (date,row[0]))
        connection.commit()

        sqlu = 'UPDATE newslinks SET text = %s WHERE (id = %s)'
        cursor.execute(sqlu, (text,row[0]))
        connection.commit()

        print('finished one row!'+str(row[0]))
        time.sleep(0.7)
```

第三部分：

对数据库中每条新闻的摘要和全文做分词处理（搜索引擎模式），整理并统计词频，并将重要的关键字写入数据库。整理关键字时，为确保搜索效率，做出了如下规定：

1:关键字长度大于1；

2:关键字词频大于3；

3:以词频为主要权重，倒序排列，最多取10个关键字写入数据库。

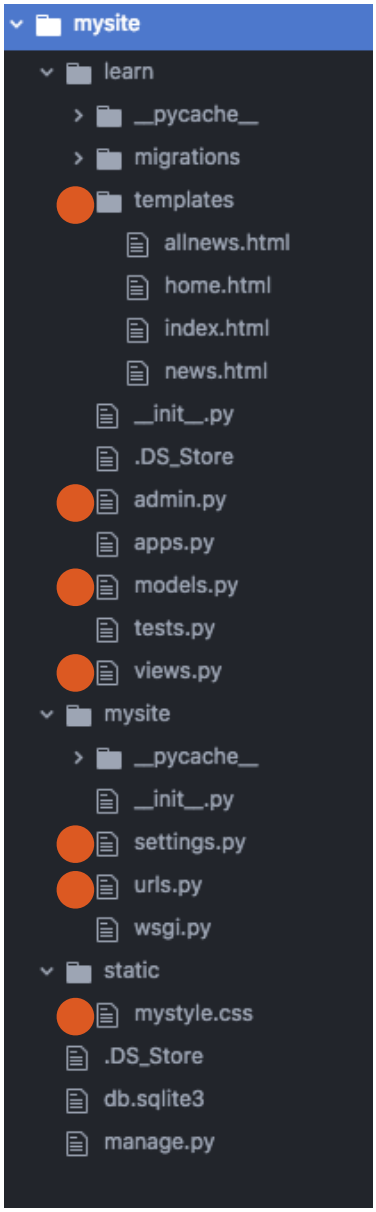
对得到的关键字建立倒排列表，便于后续索引。对在采集过程中可能出现的错误进行了预处理，避免运行时报错，降低效率。

```
connection = pymysql.connect(**db_config)
cursor = connection.cursor()
sqlr = "select * from newslinks where id>2000 "
cursor.execute(sqlr)
results=cursor.fetchall()

for row in results:
    try:
        ans=''
        wordcnt=0
        wordlist=[]
        worddict={}
        text=''
        if row[4] is not None:
            text+= row[4]
        text+= row[5]
        if text!='':
            wordlist=jieba.cut_for_search(text)
            for item in wordlist:
                if len(str(item))>1:
                    if item not in worddict:
                        worddict[item]=1
                    else:
                        worddict[item]+=1
            orderList=list(worddict.values())
            orderList.sort(reverse=True)
            for i in range(len(orderList)):
                for key in worddict:
                    if worddict[key]==orderList[i] and worddict[key]>3 and wordcnt<11:
                        ans+=(key+str(worddict[key])+',')
                        wordcnt+=1
                        worddict[key]=0

            sqlu = 'UPDATE newslinks SET keywords = %s WHERE (id = %s)'
            cursor.execute(sqlu, (ans,row[0]))
            connection.commit()
            print('finished one!'+str(row[0]))
        else:
            print('text is empty!'+str(row[0])+'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!')
            continue
    except Exception as e:
        raise e
        print('something is wrong with'+str(row[0]+'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'))
        continue

print('all finished!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!')
```

二、网站搭建

首先建立 project 命名为 mysite ，在建立 app 命名为 learn 。

文件目录如图：



主要操作的是红色标注的几个文件：

templates只要存储我使用的网页模版，四个模版由上至下别是：整合所有新闻并展示的页面、搜索首页、搜索结果显示页面、新闻详情页面。

admin.py 是我后台整理数据的操作文件。

models.py 是使用 MVC 架构时与数据库接轨的一个操作文件。

views.py 是使用 MVC 架构时处理model传来的数据，并将数据发送到html模版中的操作文件。

settings.py 是管理项目设置的文件。

urls.py 是处理不同url与view.py不同函数对接关系的操作文件。

mystyle.css 是CSS文件，为了设计并美化html模版而用。

下面分页面进行说明：

搜索首页：

构建了一个POST表单，里面包含三个搜索栏，一个提交按钮。

关键字搜索必需项，支持多关键词搜索，两个时间搜索为可选项。点击提交后，会自动判断有无时间限定。

多个关键词输入后，我们会将关键词分开，对每个关键词设立权值，单独搜索，最后整合搜索结果，若有时间限定，我们再过滤掉不合法的时间，按权值大小展示结果，得到最优的答案。

页面中“或者你想看看我们都收集了什么？”可以链接到“所有新闻页面”的第一页。

“近些天发生了这些事情：”会收集最近日期的热点新闻，并展示出来。

搜索结果展示页面：

得到关键字之后，在 views.py 中执行相应函数，在数据库中进行检索，得到数据后，将数据传递给 index.html 模版，进行展示。

检索耗时：计算了函数开始检索到检索完毕的耗时；

分页显示：（后端分页模式）在传递新闻条目时，并不是全部传递到模版。每页只显示10条结果，每次回传递当前页页码和所对应的十条结果，当点击下一页的时候，采用GET方法获取想要显示的页码，在对应好数据，传递给模版。

关键字高亮：采用前端 Javascript 方法，对带有特定id的板块进行关键字正则匹配，对其添加额外的标签。对应代码如下：

```
<script>
  function SearchHighlight(idVal, keyword) {
    var pucl = document.getElementById(idVal);
    if ("" == keyword) return;
    var temp = pucl.innerHTML;
    var htmlReg = new RegExp("\<.*?\>", "i");
    var arrA = new Array();
    for (var i = 0; true; i++) {
      var m = htmlReg.exec(temp);
      if (m) {
        arrA[i] = m;
      } else {
        break;
      }
      temp = temp.replace(m, "{[(" + i + ")]}");
    }
    words = unescape(keyword.replace(/\+/g, ' ')).split(/\s+/);
    for (w = 0; w < words.length; w++) {
      var r = new RegExp("(" + words[w].replace(/[(){}.+*?^$|\\\[\]]/g, "\\$&") + ")", "ig");
      temp = temp.replace(r, "<b style='color:Red;'>$1</b>");
    }
    for (var i = 0; i < arrA.length; i++) {
      temp = temp.replace("{[(" + i + ")]}", arrA[i]);
    }
    pucl.innerHTML = temp;
  }
  SearchHighlight("news", "{{orignalinput}}");
</script>
```

所有新闻页面：

这是我们收集到的所有新闻:

我们对得到的 6947 条新闻进行了分页展示，共有 695 页，当前您所浏览的是第 1 页。点击此处带您回到搜索首页。

新华国际时评：引领中俄关系保持高水平发展

20180911

在两国元首的战略引领下，中俄全面战略协作伙伴关系步入高水平、大发展的新时代。习近平主席和普京总统都曾用“典范”形容当前的中俄关系。

财经观察：龙狮共舞共促繁荣

20180911

在中非合作升温的大背景下，“中国龙”与“非洲狮”共同起舞，踩在经济全球化的时代鼓点上，不仅会给２６亿中非人民带来深远福祉，也将增进中非以及世界经济的发展繁荣。

施压巴解组织和国际刑事法院 美欲"一石三鸟"恐难如愿

20180911

特朗普政府意在迫使巴方接受美国调停的巴以和平方案，阻挠国际刑事法院对美国及其盟国开展战争罪行调查，并在美中期选举前巩固对外强硬捍卫国家利益的形象以争取选民支持，不过这些目的恐难实现。

国际观察：当“北欧福利主义”遭遇极右浪潮

20180910

作为“北欧福利主义”代表的“瑞典模式”遭遇极右浪潮，未来新政府组阁和社会福利制度的改革走向扑朔迷离，也给难民问题带来的“欧洲困境”增添新案例。

外媒：美国反对调查美涉战争罪行为

和搜索结果展示页面异曲同工，不在重复叙述。

新闻详情页面：

利用GET方法得到新闻id之后，在 views.py 中执行相应函数，在数据库中进行检索，得到数据后，将数据传递给 news.html 模版，进行展示。

相关新闻: 获取到当前新闻的第一个和第二个关键字，以该关键字再次进行检索，将得到的结果展示出来，只展示最相关的前四个。

具体urls对应关系：

```
urlpatterns = [
    path('', learn_views.showhomepage,name='home'),
    path('admin', admin.site.urls,name='admin'),
    path('news/<int:id>',learn_views.showNewsDetail,name='newsdetail'),
    path('search',learn_views.index,name='index'),
    path('search/<int:wantpagecnt>',learn_views.showwantindex,name='pagechangeto'),
    path('allnews/<int:wantpagecnt>',learn_views.showwantallnews,name='allnewspagechangeto'),
]
```

三：具体实现

请移步代码，那里有比较详细的注释说明。