

# 第10章 使用python编写网络爬虫

计算机系: 王学军

邮箱: wangxuejun@stdu.edu.cn

## CONTENT

- 1 什么是网络爬虫
- 02 网页结构
- 03 python爬虫相关库简介



什么是网络爬虫

#### 网络爬虫简介

#### 什么是网络爬虫

网络爬虫(Web Spider)是一种按照一定的规则,自动地抓取万维网信息的程序或者脚本。它可以通口程序口口来口取指定网口中的指定信息,如百度口吧的帖子信息,新口网站的新口、文章等等。获取到的数据多用于大数据分析口景,因此口写网口爬虫是从事大数据分析行口的必口技能之一。

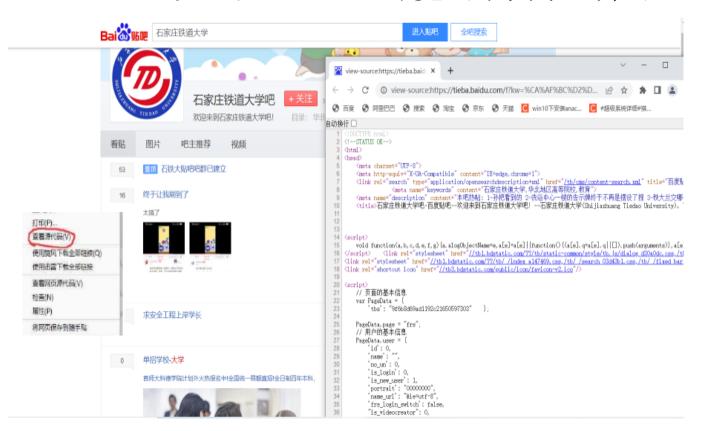


#### 网页结构简介

#### 网页结构

在□写网□爬虫之前,首先要□网□□构有一定的了解。大多数网□都是使用HTML(超文本标记语言)进行编写 ,通□□取网□源代□,我□就可以看到□个□面的HTML信息。

下面 以Chrome浏览器为例,介绍查看网页源代码的方法:



| 如口所示,打开一个网   |   |
|--------------|---|
| □,右□□□ 空白□,  |   |
| 在右口菜口 中有一个 🗆 |   |
| 看源代口 口口,通口点  | į |
| □□看源代□,我□就   |   |
| 可以 看到口个口面的   |   |
| HTML代码。      |   |

#### 网页结构简介

#### 网页结构

```
(!- 結果-php/道架模版 --)
      (/div)
  (/div)
  (div class="search_main_wrap")
      (div class="search form")
             (a rel="noopener" title="到贴吧首页" href="/" class="search_logo" id="search_logo_large">(/a)
              《a rel="noopener" id="search_logo_small" class="" title="到贴吧首页" href="/"></a>
              (form name="f1" class="clearfix j_search_form" action="/f"
                   id="tb header search form")
                                    Kinput class="search_ipt search_inp_border j_search_input tb_header_search_input"
                        name="kw1" value="石家庄铁道大学" type="text" autocomplete="off" size="42"
                        tabinder="1" id="wd1" maxlength="100" n-webkit-grammar="builtin;search"
                        r-webkit-speech="true"/>
                 《input autocomplete="off" type="hidden" name="kw" value="石家庄铁道大学" id="wd2"/>
                 (span class="search btn wrap search btn enter ba wrap")
                     <a rel="noopener" class="search_btn_search_btn_enter_ba_j_enter_ba" href="#"</p>
                        onclick="return false:"
                        onnousedown="this.className+=' search_btn_down'"
                       onnouseout="this.className=this.className.replace('search_btn_down','')'*)造入贴吧(/a>
                 (span class="search btn wrap")
                     «a rel="noopener" class="search_btn j_search_post" href="#" onclick="return false;"
                        onnousedown="this.className+=' search btn down'
                        onnouseout="this, classlame=this, classlame, replace('search btn down','')")全吧搜索(/a)
                 \div id="pagelet_search/pagelet/search_ad">\//div>
                            (input type="radio" class="nowth" name="tb" id="nowth")(label
                    for="nowth">吧内搜索(/label)
                 《input type="radio" class="searchtb" name="tb" id="searchtb") (label for="searchtb") 搜贴(/label)
                 《input type="radio" class="authortb" name="tb" id="authortb">(label for="authortb")投入(label)
                 (input type="radio" class="jointb" checked="checked" name="tb" id="jointb")(label
                     for="iointb">i#呼(/label)
                 (input type="radio" class="searchtag" name="tb" id="searchtag"
                        style="display:none:">Clabel for="searchtag"
                                                  style="display:none;">搜标签(/label)
              ⟨⟨p⟩
          (/div)
      (/div)
  (/div)
(div)
Idiv)
```

| □ □ 器会在新打开的 □ □ □ 中□ 示                    |
|---|
| □ 网 □ 的源 代 □,此 □ 我 □ 就 会 □ □ , 比          |
| 如我口想要口取口个口面所有帖子的口目,                       |
| 都可以在网口源代口中找到。而网口爬虫                        |
| <b>的主要工作原理</b> ,就是在网口源代码中把我               |
| □想要的内容抽取州来。                               |
| HTML语言中是通过不同的标签来编                         |
| 写网口的,不同的口口口口着网口中不同                        |
| 的元素,有些口口之口可以嵌套,有些口口                       |
| 通□ class 属性来指定自己的□□, 有些□                  |
| □通□ i d属性来唯一□示自己,常用的有:                    |
| 〈div〉标签,用来标定一块区域;〈p〉标签,                   |
| 用于口示一段文字; <h1><h2><h3>等标签,</h3></h2></h1> |
| 用于口示一个口口; <a>标签,用于放置一</a>                 |
| 个口接。                                      |

在进行爬虫实例前,还要了解爬虫中常用的一些库。

#### requests库

requests库是一个简洁且简单的处理HTTP请求的第三方库。其最大的优点就是程序编写过程更接近正常URL 访问过程。其支持非常丰富的链接访问功能,包括:

- 国口域名和URL获取
- HTTP长连接和连接缓存
- HTTP会话和Cookie保持
- 浏览器使用风格的SSL验证
- 基本的摘要认证
- 有效的键值对Cookie记录
- 自动解压缩
- 自动内容解码
- 文件分块上传
- HTTP(S)代理功能
- 连接超时处理
- 流数据下载等

有关requests库更多介绍可以访问:

https://docs.python-requests.org/zh CN/latest/user/quickstart.html



快速上手¶

迫不及待了吗?本页内容为如何入门 Requests 提供了很好的指引。其假设你已经安装了 Requests。如果还没有,去安装一节看看吧。

首先,确认一下:

- Requests 已安装
- Requests 是最新的

让我们从一些简单的示例开始吧。

#### 发送请求

使用 Requests 发送网络请求非常简单。

一开始要导入 Requests 模块:

#### development release. >>> import requests

然后,尝试获取某个网页。本例子中,我们来获取 Github 的公共时间线:

>>> r = requests.get('https://api.github.com/events')

现在,我们有一个名为 r 的 Response 对象。我们可以从这个对象中获取所有我们想要的信息。

Requests 简便的 API 意味着所有 HTTP 请求类型都是显而易见的。例如,你可以这样发送一个 HTTP POST 请求:

#### Stay Informed

Requests is an elegant and

currently looking at the documentation of the

simple HTTP library for Python, built for human beings. You are

Receive updates on new releases and upcoming projects.

Join Mailing List.



### requests库中的网页请求函数

| 描述                          |
|-----------------------------|
| □□于HTTP的GET方式,获取网页最常用的方法,可  |
| 以增加timeout=n参数,设定每次请求超时时间为n |
| 秒                           |
| □□于HTTP的 POST 方式,其中字典用于传递客户 |
| 数据                          |
| 口口于HTTP的DELETE方式            |
| 口口于HTTP的HEAD 方式             |
| □□于HTTP的options方式           |
| □□于HTTP的PUT方式,其中字典用于传递客户数据  |
|                             |

#### lxml库、selenium库、re库

● Beautiful Soup 和 lxml是两个常用的爬虫模块,常被用来对抓取的网页进行解析,以便于进一步的抓取

● selenium可以模拟真实浏览器,自动化测试工具,支持多种浏览器,爬虫中主要用来解决JavaScript渲染问题

● re库:正则表达式用来简洁表达一组字符串的表达式。进行字符串匹配

### 使用requests库获取网页源代码

在编写网页爬虫时,需要制定一个url作为爬取的起始点,首先,进入石家庄铁道大学的百度贴吧,为了后面实现翻页功能,先点击下一页,复制地址栏中的url:

 $https://tieba.\ baidu.\ com/f?kw=\%CA\%AF\%BC\%D2\%D7\%AF\%CC\%FA\%B5\%C0\%B4\%F3\%D1\%A7\&fr=a1a0\&tp1=5\&dyTabStr=MCw2LDMsNCwyLDEsNSw4LDcsOQ\%3D\%3D$ 

首先创建一个变量名为url,并把上述url复制给这个变量名。 然后创建一个变量名为html,将获取的网络源代码保存在这个变量中,通过输出html.text就可以获取网页源代码。

ur1 =

'https://tieba.baidu.com/f?kw=%CA%AF%BC%D2%D7%AF%CC%FA%B5%C0%B4%F3%D1%A7&fr=a1a0&tp1=5&dyTabStr=MCw2LDMsNCwyLDEsNSw4LDcsOQ%3D%3D'

html = requests.get(url)

print(html.text)

#### 使用正则表达式实现翻页功能

结合实例来演示正则表达式的作用以及使用方法。

首先,我们发现通过网页中点击下一页就可以发现,&pn的数值为当前页面数减去1再乘以50,比如第5页url中 &pn=200,除了&pn的值,其它完全不变。当我们在地址栏中修改&pn的值为0时,按下回车,就会发现跳转到石家庄铁道大学贴吧的第一页。

因此,可以通过修改&pn的值来实现翻页功能,即获取每一页的网页源代码。

https://tieba.baidu.com/f?kw=%E7%9F%B3%E5%AE%B6%E5%BA%84%E9%93%81%E9%81%93%E5%A4%A7%E5%AD%A6&ie=utf-8&pn=50

https://tieba.baidu.com/f?kw=%E7%9F%B3%E5%AE%B6%E5%BA%84%E9%93%81%E9%81%93%E 5%A4%A7%E5%AD%A6&ie=utf-8&pn=100

https://tieba.baidu.com/f?kw=%E7%9F%B3%E5%AE%B6%E5%BA%84%E9%93%81%E9%81%93%E5%A4%A7%E5%AD%A6&ie=utf-8&pn=150

#### 使用正则表达式实现翻页功能

```
for i in range(10):
    new_url = re.sub('&pn=\d+', '&pn=%d' % (i*50), url)
    print(new_url)
    html = requests.get(new_url)
```

re.sub()用于替换字符串中的匹配项。

第一个参数为正则表达式,&pn=\d+表示获取文本中'pn='字段后面的多个数字部分,'\d'表示一个数字字符,加号表示连续出现多次;

第二个参数表示将文本中'&pn='字段后面数字的值替换成i\*50;

第三个参数表示把url变量中的文本作为处理文本。

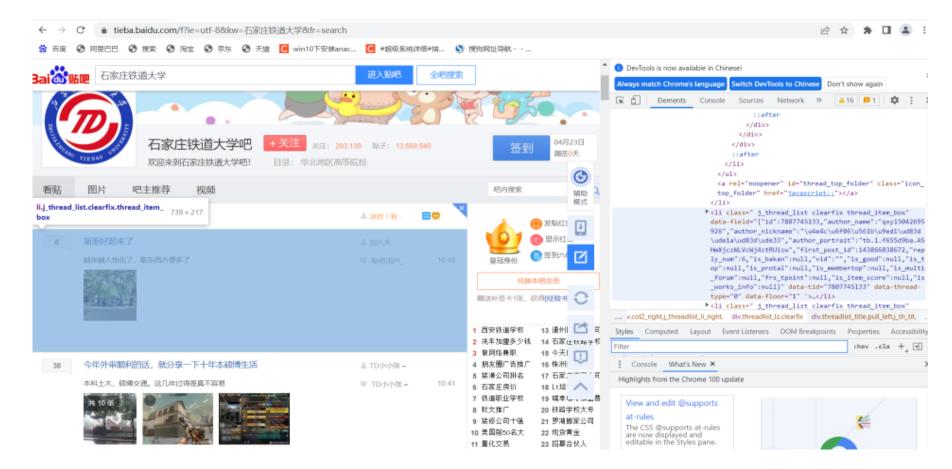
通过输出new\_url,我们就可以看到贴吧中第1页到第10页的url了,可以通过设置range的范围来获取更多页数的url。

获取到每一页的url后,我们就可以再次使用requests.get()方法来获取网页源代码。

#### 使用Xpath进行页面定位

Xpath是一种针对xml文本的快速标记语言,就像现实生活中描述家庭地址,精准高效通过Xpath可以快速在网页源代码中找到想要的所有内容。

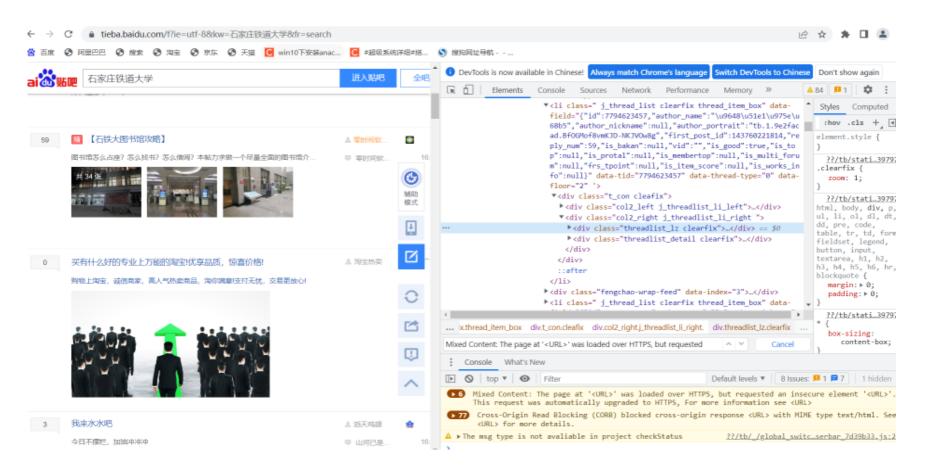
比如获取每一页帖子的标题,首先使用检查的方法分析网页源代码,在页面空白处点击右键,选择检查。如图:



打开开发者工具,通过点 开每一层标签以及鼠标在 代码上移动,左侧对应的 部分会用蓝底显示,找到 帖子标题所在位置。

#### 使用Xpath进行页面定位

通过分析可以看到,每个帖子的题目内容都在<a>标签中,而<a>标签的上层为一个class属性为 "threadlist\_title pull\_left j\_th\_tit"的<div>标签中,因此,只要找到所有class属性等于"threadlist\_title pull\_left j\_th\_tit"的<div>标签下的<a>标签的文字内容即可。所以定义一个xpath变量,并赋值。 注意threadlist\_title pull\_left j\_th\_tit 末尾有一个空格。



#### 使用Xpath进行页面定位

```
xpath = '//*[@class="threadlist_title pull_left j_th_tit "]/a/text()'
pages = etree.HTML(html.content)
title = pages.xpath(xpath)
```

//\*表示xpath表达式的开始,[@class="threadlist\_title pull\_left j\_th\_tit"]表示求class属性等于 "threadlist\_title pull\_left j\_th\_tit"的标签,/a表示该标签下的<a>标签,/text()表示获取<a>标签的文本信息。

然后将获取的到的网页源代码转化成etree类型,并使用xpath定位。

由于一个页面有多个标题,符合要求的<div>标签也有多个,因此pages.xpath()方法返回值为一个列表保存在title变量中,通过循环输出title列表中的内容,可以获取指定页码的贴吧中所有帖子的题目。

完整代码参考实验任务书。



# 第10章 使用python编写网络爬虫

Thank
you! **Q** & A