

10 | Python爬虫：如何自动化下载王祖贤海报？

2019-01-04 陈旻



讲述：陈旻

时长 08:39 大小 7.93M



上一讲中我给你讲了如何使用八爪鱼采集数据，对于数据采集刚刚入门的人来说，像八爪鱼这种可视化的采集是一种非常好的方式。它最大的优点就是上手速度快，当然也存在一些问题，比如运行速度慢、可控性差等。

相比之下，爬虫可以很好地避免这些问题，今天我来分享下如何通过编写爬虫抓取数据。

爬虫的流程

相信你对“爬虫”这个词已经非常熟悉了，爬虫实际上是用浏览器访问的方式模拟了访问网站的过程，整个过程包括三个阶段：打开网页、提取数据和保存数据。

在 Python 中，这三个阶段都有对应的工具可以使用。

在“打开网页”这一步骤中，可以使用 Requests 访问页面，得到服务器返回给我们的数据，这里包括 HTML 页面以及 JSON 数据。

在“提取数据”这一步骤中，主要用到了两个工具。针对 HTML 页面，可以使用 XPath 进行元素定位，提取数据；针对 JSON 数据，可以使用 JSON 进行解析。


在最后一步“保存数据”中，我们可以使用 Pandas 保存数据，最后导出 CSV 文件。

下面我来分别介绍下这些工具的使用。

Requests 访问页面

Requests 是 Python HTTP 的客户端库，编写爬虫的时候都会用到，编写起来也很简单。它有两种访问方式：Get 和 Post。这两者最直观的区别就是：Get 把参数包含在 url 中，而 Post 通过 request body 来传递参数。


假设我们想访问豆瓣，那么用 Get 访问的话，代码可以写成下面这样的：

 复制代码

```
1 r = requests.get('http://www.douban.com')
```

代码里的“r”就是 Get 请求后的访问结果，然后我们可以使用 r.text 或 r.content 来获取 HTML 的正文。

如果我们想要使用 Post 进行表单传递，代码就可以这样写：

 复制代码

```
1 r = requests.post('http://xxx.com', data = {'key':'value'})
```

这里 data 就是传递的表单参数，data 的数据类型是个字典的结构，采用 key 和 value 的方式进行存储。

XPath 定位

XPath 是 XML 的路径语言，实际上是通过元素和属性进行导航，帮我们定位位置。它有几种常用的路径表达方式。

表达式	含义
node	选node节点的所有子节点
/	从根节点选取
//	选取所有的当前节点，不考虑他们的位置
.	当前节点
..	父节点
@	属性选择
	或，两个节点的合计
text()	当前路径下的文本内容

我来给你简单举一些例子：


- 1. xpath('node') 选取了 node 节点的所有子节点；
- 2. xpath(' /div') 从根节点上选取 div 节点；
- 3. xpath(' //div') 选取所有的 div 节点；
- 4. xpath(' ./div') 选取当前节点下的 div 节点；
- 5. xpath(' ...') 回到上一个节点；
- 6. xpath(' //@id') 选取所有的 id 属性；

7. `xpath(' //book[@id]')` 选取所有拥有名为 id 的属性的 book 元素；
8. `xpath(' //book[@id= "abc"]')` 选取所有 book 元素，且这些 book 元素拥有 id="abc" 的属性；
9. `xpath(' //book/title | //book/price')` 选取 book 元素的所有 title 和 price 元素。

上面我只是列举了 XPath 的部分应用，XPath 的选择功能非常强大，它可以提供超过 100 个内建函数，来做匹配。我们想要定位的节点，几乎都可以使用 XPath 来选择。

使用 XPath 定位，你会用到 Python 的一个解析库 lxml。这个库的解析效率非常高，使用起来也很简便，只需要调用 HTML 解析命令即可，然后再对 HTML 进行 XPath 函数的调用。

比如我们想要定位到 HTML 中的所有列表项目，可以采用下面这段代码。

 复制代码

```
1 from lxml import etree
2 html = etree.HTML(html)
3 result = html.xpath('//li')
```

JSON 对象

JSON 是一种轻量级的交互方式，在 Python 中有 JSON 库，可以让我们将 Python 对象和 JSON 对象进行转换。为什么要转换呢？原因也很简单。将 JSON 对象转换成为 Python 对象，我们对数据进行解析就更方便了。

方法	含义
<code>json.dumps()</code>	将Python对象转换成JSON对象
<code>json.loads()</code>	将JSON对象转换成Python对象

这是一段将 JSON 格式转换成 Python 对象的代码，你可以自己运行下这个程序的结果。

```
1 import json
2 jsonData = '{"a":1,"b":2,"c":3,"d":4,"e":5}';
3 input = json.loads(jsonData)
4 print input
```

接下来，我们就要进行实战了，我会从两个角度给你讲解如何使用 Python 爬取海报，一个是通过 JSON 数据爬取，一个是通过 XPath 定位爬取。

如何使用 JSON 数据自动下载王祖贤的海报

我在上面讲了 Python 爬虫的基本原理和实现的工具，下面我们来实战一下。如果想要从豆瓣图片中下载王祖贤的海报，你应该先把我们日常的操作步骤整理下来：

1. 打开网页；
2. 输入关键词“王祖贤”；
3. 在搜索结果页中选择“图片”；
4. 下载图片页中的所有海报。

这里你需要注意的是，如果爬取的页面是动态页面，就需要关注 XHR 数据。因为动态页面的原理就是通过原生的 XHR 数据对象发出 HTTP 请求，得到服务器返回的数据后，再进行处理。XHR 会用于在后台与服务器交换数据。

你需要使用浏览器的插件查看 XHR 数据，比如在 Chrome 浏览器中使用开发者工具。

在豆瓣搜索中，我们对“王祖贤”进行了模拟，发现 XHR 数据中有一个请求是这样的：

[https://www.douban.com/j/search_photo?
q=%E7%8E%8B%E7%A5%96%E8%B4%A4&limit=20&start=0](https://www.douban.com/j/search_photo?q=%E7%8E%8B%E7%A5%96%E8%B4%A4&limit=20&start=0)

url 中的乱码正是中文的 url 编码，打开后，我们看到了很清爽的 JSON 格式对象，展示的形式是这样的：

```
1 {"images":
2     [{"src": ..., "author": ..., "url":..., "id": ..., "title": ..., "width":..., "height":...},
```

```
3     ...
4     {"src": ..., "author": ..., "url":..., "id": ..., "title": ..., "width":..., "height":...}],
5     "total":22471,"limit":20,"more":true}
```



从这个 JSON 对象中，我们能看到，王祖贤的图片一共有 22471 张，其中一次只返回了 20 张，还有更多的数据可以请求。数据被放到了 images 对象里，它是个数组的结构，每个数组的元素是个字典的类型，分别告诉了 src、author、url、id、title、width 和 height 字段，这些字段代表的含义分别是原图片的地址、作者、发布地址、图片 ID、标题、图片宽度、图片高度等信息。


有了这个 JSON 信息，你很容易就可以把图片下载下来。当然你还需要寻找 XHR 请求的 url 规律。

如何查看呢，我们再来重新看下这个网址本身。

https://www.douban.com/j/search_photo?q=王祖贤&limit=20&start=0

你会发现，网址中有三个参数：q、limit 和 start。start 实际上是请求的起始 ID，这里我们注意到它对图片的顺序标识是从 0 开始计算的。所以如果你想要从第 21 个图片进行下载，你可以将 start 设置为 20。

王祖贤的图片一共有 22471 张，你可以写个 for 循环来跑完所有的请求，具体的代码如下：

 复制代码

```
1 # coding:utf-8
2 import requests
3 import json
4 query = '王祖贤'
5 ''' 下载图片 '''
6 def download(src, id):
7     dir = './' + str(id) + '.jpg'
8     try:
9         pic = requests.get(src, timeout=10)
10        fp = open(dir, 'wb')
11        fp.write(pic.content)
12        fp.close()
13    except requests.exceptions.ConnectionError:
14        print('图片无法下载')
```



```

15
16 ''' for 循环 请求全部的 url '''
17 for i in range(0, 22471, 20):
18     url = 'https://www.douban.com/j/search_photo?q='+query+'&limit=20&start='+str(i)
19     html = requests.get(url).text    # 得到返回结果
20     response = json.loads(html,encoding='utf-8') # 将 JSON 格式转换成 Python 对象
21     for image in response['images']:
22         print(image['src']) # 查看当前下载的图片网址
23         download(image['src'], image['id']) # 下载一张图片

```

如何使用 XPath 自动下载王祖贤的电影海报封面

如果你遇到 JSON 的数据格式，那么恭喜你，数据结构很清爽，通过 Python 的 JSON 库就可以解析。

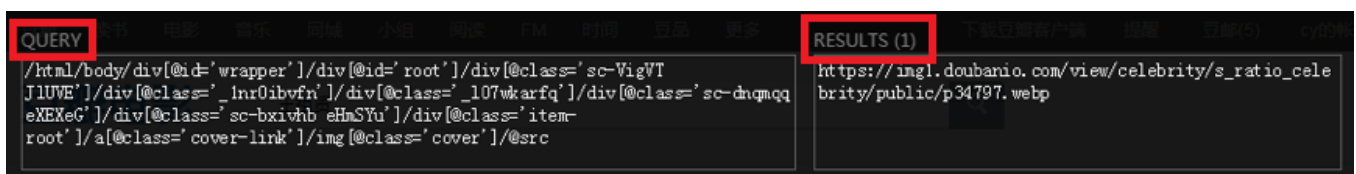
但有时候，网页会用 JS 请求数据，那么只有 JS 都加载完之后，我们才能获取完整的 HTML 文件。XPath 可以不受加载的限制，帮我们定位想要的元素。

比如，我们想要从豆瓣电影上下载王祖贤的电影封面，需要先梳理下人工的操作流程：

1. [打开网页 movie.douban.com](http://movie.douban.com)；
2. 输入关键词“王祖贤”；
3. 下载图片页中的所有电影封面。

这里你需要用 XPath 定位图片的网址，以及电影的名称。


一个快速定位 XPath 的方法就是采用浏览器的 XPath Helper 插件，使用 Ctrl+Shift+X 快捷键的时候，用鼠标选中你想要定位的元素，就会得到类似下面的结果。



XPath Helper 插件中有两个参数，一个是 Query，另一个是 Results。Query 其实就是让你来输入 XPath 语法，然后在 Results 里看到匹配的元素的结果。


我们看到，这里选中的是一个元素，我们要匹配上所有的电影海报，就需要缩减 XPath 表达式。你可以在 Query 中进行 XPath 表达式的缩减，尝试去掉 XPath 表达式中的一些内容，在 Results 中会自动出现匹配的结果。

经过缩减之后，你可以得到电影海报的 XPath（假设为变量 src_xpath）：

 复制代码

```
1 //div[@class='item-root']/a[@class='cover-link']/img[@class='cover']/@src
```

以及电影名称的 XPath（假设为变量 title_xpath）：


 复制代码

```
1 //div[@class='item-root']/div[@class='detail']/div[@class='title']/a[@class='title-text'
```



但有时候当我们直接用 Requests 获取 HTML 的时候，发现想要的 XPath 并不存在。这是因为 HTML 还没有加载完，因此你需要一个工具，来进行网页加载的模拟，直到完成加载后再给你完整的 HTML。

在 Python 中，这个工具就是 Selenium 库，使用方法如下：

 复制代码

```
1 from selenium import webdriver
2 driver = webdriver.Chrome()
3 driver.get(request_url)
```


Selenium 是 Web 应用的测试工具，可以直接运行在浏览器中，它的原理是模拟用户在进行操作，支持当前多种主流的浏览器。

这里我们模拟 Chrome 浏览器的页面访问。

你需要先引用 Selenium 中的 WebDriver 库。WebDriver 实际上就是 Selenium 2，是一种用于 Web 应用程序的自动测试工具，提供了一套友好的 API，方便我们进行操作。

然后通过 WebDriver 创建一个 Chrome 浏览器的 drive，再通过 drive 获取访问页面的完整 HTML。

当你获取到完整的 HTML 时，就可以对 HTML 中的 XPath 进行提取，在这里我们需要找到图片地址 srcs 和电影名称 titles。这里通过 XPath 语法匹配到了多个元素，因为是多个元素，所以我们需要用 for 循环来对每个元素进行提取。

 复制代码

```
1 srcs = html.xpath(src_xpath)
2 titles = html.xpath(title_path)
3 for src, title in zip(srcs, titles):
4     download(src, title.text)
```

然后使用上面我编写好的 download 函数进行图片下载。

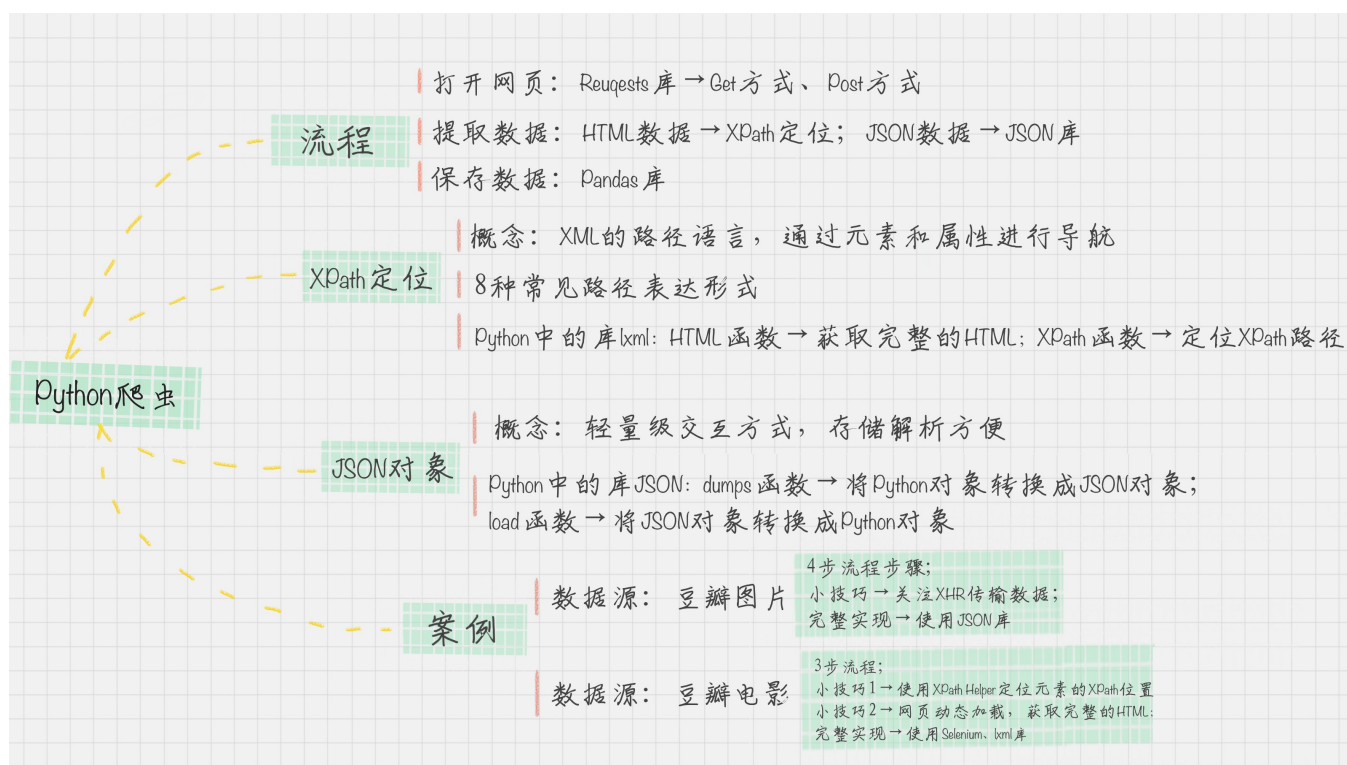
总结

好了，这样就大功告成了，程序可以源源不断地采集你想要的内容。这节课，我想让你掌握的是：

1. Python 爬虫的流程；
2. 了解 XPath 定位，JSON 对象解析；
3. 如何使用 lxml 库，进行 XPath 的提取；
4. 如何在 Python 中使用 Selenium 库来帮助你模拟浏览器，获取完整的 HTML。

其中，Python + Selenium + 第三方浏览器可以让我们处理多种复杂场景，包括网页动态加载、JS 响应、Post 表单等。因为 Selenium 模拟的就是一个真实的用户的操作行为，就不用担心 cookie 追踪和隐藏字段的干扰了。

当然，Python 还给我们提供了数据处理工具，比如 lxml 库和 JSON 库，这样就可以提取想要的内容了。



最后, 你不妨来实践一下, 你最喜欢哪个明星? 如果想要自动下载这个明星的图片, 该如何操作呢? 欢迎和我在评论区进行探讨。

你也可以把这篇文章分享给你的朋友或者同事, 一起动手练习一下。



数据分析实战 45 讲

即学即用的数据分析入门课

陈旸

清华大学计算机博士



新版升级: 点击「 请朋友读」, 10位好友免费读, 邀请订阅更有**现金**奖励。

上一篇 09 | 数据采集：如何用八爪鱼采集微博上的“D&G”评论

下一篇 11 | 数据科学家80%时间都花费在了这些清洗任务上？

精选留言 (51)

写留言



rOMEo罗密...

2019-01-04

14

老师请问一下：如果是需要用户登陆后才能爬取的数据该怎么用python来实现呢？

展开

作者回复: 你可以使用python+selenium的方式完成账户的自动登录，因为selenium是个自动化测试的框架，使用selenium的webdriver就可以模拟浏览器的行为。找到输入用户名密码的地方，输入相应的值，然后模拟点击即可完成登录（没有验证码的情况下）

另外你也可以使用cookie来登录网站，方法是你登录网站时，先保存网站的cookie，然后用下次访问的时候，加载之前保存的cookie，放到request headers中，这样就不需要再登录网站了



caidy

2019-01-05

7

你需要使用浏览器的插件查看XHR数据，比如在Chrome的开发者工具
在豆瓣搜索中，我们对“王祖贤”进行了模拟，发现XHR数据中有一个请求是这样的：
https://www.douban.com/j/search_photo?q=王祖贤&limit=20&start=0

这个是如何查出来的，我使用chrome的开发者工具查看，但是查不到这部分，麻烦老...

展开



伪君子

2019-01-04

5

那些用ChromeDriver的出现报错的可能是没有安装ChromeDriver，或者是没给出ChromeDriver的路径，具体可以看看下面这篇文章。

<https://mp.weixin.qq.com/s/UL0bcLr3KOb-qpI9oegaIQ>

展开

作者回复: 对的，主要是配置ChromeDriver的问题。有相同问题的人，可以看下这个留言



伪君子

2019-01-04

👍 5

老师您好，我根据您的代码修改了一下，主要是添加了一个图片的目录，然后是下载大图。这里的大图是因为 /photo/thumb/public/ 这样的链接下载的图片是缩略图，只有把 thumb 替换成 l 之后下载的图片才是相对来说的大图。replace 方法和 re 中的 sub 方法都能实现替换，我的疑问是哪个实现起来更高速一点呢？提前感谢老师，我写的代码在下面~...

展开 ∨



許敲敲

2019-01-04

👍 4

要下载所有James 哈登的图片

展开 ∨

作者回复: NBA明星也是不错的选择



LY

2019-01-04

👍 3

#环境：Mac Python3

#pip install selenium

#下载chromedriver，放到项目路径下

(<https://npm.taobao.org/mirrors/chromedriver/2.33/>)

coding:utf-8...

展开 ∨

作者回复: GoodJob



germany

2019-01-04

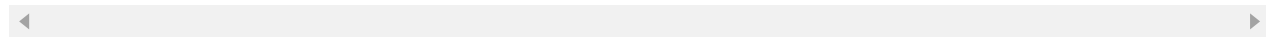
👍 3

老师：为什么我在豆瓣网查询图片的网址与你不一样？

<https://www.douban.com/search?cat=1025&q=王祖贤&source=suggest>。是什么

原因？

作者回复: 咱们访问豆瓣查询图片的网址应该是一样的。只是我给出的是json的链接。方法是：
用Chrome浏览器的开发者工具，可以监测出来网页中是否有json数据的传输，所以我给出的链接是json数据传输的链接 https://www.douban.com/j/search_photo?q=%E7%8E%8B%E7%A5%96%E8%B4%A4&limit=20&start=0



ldw

2019-01-06

👍 2

可以用爬虫爬谷歌吗？会不会被当成恶意攻击？不会引来国际官司吧。

展开 ▾



CNxxxxxx

2019-01-06

👍 1

好玩好玩。就是我用chromedriver会调用浏览器访问页面，不知道大家会不会

```
# coding:utf-8
```

```
import requests
```

```
import json
```

```
from lxml import etree...
```

展开 ▾



ldw

2019-01-06

👍 1

网上最丰富的图片资源可能是谷歌的图片吧？他们不是号称把全网的内容都保存了镜像吗？

但是，用爬虫爬谷歌的话会不会被当成恶意攻击啊？会不会惹上国际官司啊？

请老师解惑。谢谢

展开 ▾



yeeeeeeeti

2019-01-04

👍 1

老师您这个一个页面显示20条url，只下载了一条呀。传入的src 是一个的类型是str，就访问第一个元素吗？



比国王

2019-01-04

👍 1

下载所有James 哈登 后撤步三分的图片

展开 ▾



Yezhiwei

2019-01-04

👍 1

用Scrapy爬取数据更方便哈，请问老师怎么做一个通用的爬虫呢？比如要爬取文章标题和内容，不同的网站Xpath结构不一样，如果源少的话可以分别配置，但如果要爬取几百上千的网站数据，分别配置Xpath挺麻烦的。请问这个问题有什么解决方案吗？谢谢

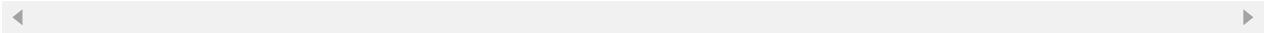
展开 ▾

作者回复: 网站的抓取和网页的HTML结构有很大关系，所以一般都是用XPath解析，如果你用第三方工具，比如八爪鱼，也是要个性化的把每个网站流程模拟出来，这样工具会自动定位XPath

网站的抓取和网页的HTML结构有很大关系，所以一般都是用XPath解析，如果你用第三方工具，比如八爪鱼，也是要个性化的把每个网站流程模拟出来，这样工具会自动定位XPath

如果想要做一个通用的解决方案，自动识别文章的标题和内容。就需要先把HTML下载下来，然后将HTML解析为DOM树，再对每个节点做评估（文章标题还是内容的可能性）

这样做的好处是通用性强，缺点就是可能会出错。



爱喝酸奶的...

2019-02-26

👍

有个问题selenium，是用来自动化测试的，他回打开浏览器.....我做爬虫是不想让代码打开浏览器，只想要他爬取的动作~要怎么办呢？



□ □ ...

2019-02-24

👍

老师您能不能帮我看下哪里有错，我从pycharm换到了jupyter都下在不了图片海报。第一个for循环用print 可以输出内容，第二个for 循环没有任何输出。

coding:utf-8

```
import requests...
```

展开 ▾





废蛋-02-22



Selenium ide火狐插件可以自动生成模拟的脚本

展开 ∨



执笔，封心

2019-02-19



发现xpath这块不会了

展开 ∨



littlePerf...

2019-02-18



老师我用 XPath 的方法下载了图片但是打不开？

展开 ∨



王彬成

2019-02-12



在讲解“如何使用 XPath 自动下载王祖贤的电影海报封面”的内容时，希望能给出完整的代码块。因为感觉很凌乱，不知如何下手。

看了留言中的代码块，下载下来的海报文件也无法打开

展开 ∨



王彬成

2019-02-12



mac用户启动 ChromeDriver 方案：

下载chromedriver的2.46版本，因其对应的Chrome版本为v71-73，

代码层写入即可启动：

```
from selenium import webdriver
```

```
driver = webdriver.Chrome('/Users/' + username + '/Downloads/chromedriver') //括号...
```

展开 ∨