issuey Lv1
2019年12月05日 阅读 2079

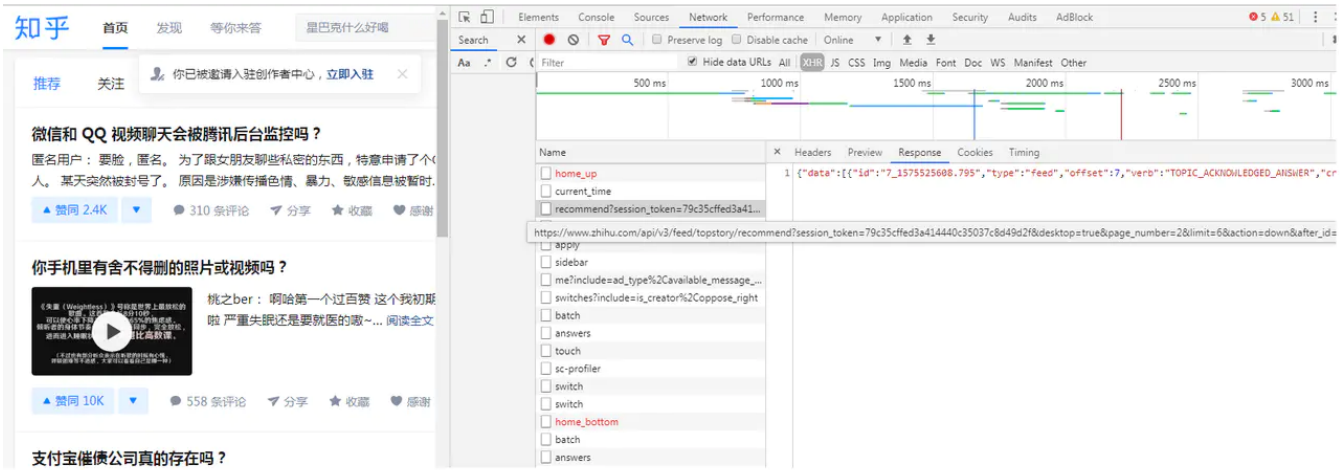
关注

小菜鸟的python爬虫之路：爬取知乎感兴趣的话题及详细回答

听说python爬虫经典实战项目有：豆瓣、知乎、 网易云音乐、 糗事百科，今天我们就来实战一下知乎。

一.分析知乎页面

我用的是chrome浏览器，打开知乎首页 www.zhihu.com/, 按F12进入开发者模式，选择 network 选项，刷新一下网页，发现有一个网络请求返回了一堆奇怪的json数据。



copy json数据到 www.json.cn/ 查看

这正是我们要的数据. 查看header信息部分

二.爬取知乎推荐话题

依赖库

```
import requests
import json
import sys
import random
```

请求头部分: cookie直接从header里的cookie复制

```
headers = {
    'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_3) AppleWebKit/537.36 (KHTM
    'referer': 'https: // www.zhihu.com /',
    'cookie': '_zap=5be1e1ef-530e-4b98-a2c5-a74abc65328c; __DAYU_PP=AnJrB73abrJaaAUnRnbqfff
}
```

分析url
第一页:

```
Request URL: https://www.zhihu.com/api/v3/feed/topstory/recommend?session_token=79c35cffed3
```

第二页

```
Request URL: https://www.zhihu.com/api/v3/feed/topstory/recommend?session_token=79c35cffed3
```

末尾的 after_id 代表分页属性，第一页 after_id=5,第二页 after_id=10，第三页 after_id=15...
个人ip大量爬取容易封ip，这里我们使用代理ip（www.goubanjia.com/ 使用的是免费的代理ip，有时效限制）。网络请求部分封装如下

```
poxy_list=[]

def initPoxy():
    poxy_list.append({"http":"223.111.131.100:8080"})
    poxy_list.append({"http": "218.60.8.99:3129"})
    poxy_list.append({"http": "39.137.107.98:80"})
    poxy_list.append({"http": "118.89.234.236:8787"})
    poxy_list.append({"http": "218.22.7.62:53281"})
    poxy_list.append({"http": "117.57.91.235:9999"})
    poxy_list.append({"http": "110.243.23.117:9999"})
```

```
    for num in range(1,3):#这里我们只请求前两页的推荐话题，可以自己改
        url_list.append(baseurl + str(num*5))
    for url in url_list:
        regPoxy(url,1)

#查询代理ip是否可用，不可用重新随机挑选一个ip，num =1 表示爬取推荐话题，num =2 表示爬取某一个话题全部回答
def regPoxy(url,num):
    response = requests.get(url, headers=headers, proxies=random.choice(poxy_list))
    if response.status_code == 200:
        if num == 1:
            getzhihutitle(response)
        else:
            getZhiHuItemDetail(response)
    else:
        regPoxy(url)
```

网络请求后返回json数据,接下来解析

```
#获取知乎推荐标题
def getzhihutitle(response):

    html = response.text
    dict_json = json.loads(html)
    dit_list = dict_json['data']
    for ditc in dit_list:
        ditTarget = ditc['target']
        # 标题
        try:
            dict_question = ditTarget['question']
            print(dict_question['title'])
        except:
            print(ditTarget['title'])

        # 回答者
        print('回答者: ' + ditTarget['author']['name'])
        # 回答者个人签名
        print('个人签名: ' + ditTarget['author']['headline'])

        if dict_question['type'] == 'question':
            # 问题具体页面
            print('问题详细url:  https://www.zhihu.com/question/' + str(dict_question['
```

```
        htmls = BeautifulSoup(ditTarget['content'], "html.parser")
        print(htmls.get_text())
        print('')
```

上面的数据是知乎推荐的话题，包括标题和一条回答。
接下来我们再爬取某一条话题里面的所有回答

三.爬取某一条话题的所有回答

分析：
随便点击一条话题详细回答页面，按F12查看network信息，刷新一下页面

发现

```
Request URL: https://www.zhihu.com/api/v4/questions/36789686/answers?include=data%5B*%5D.is
```

这个url的resopnse 返回的json正式我们要的数据

上面的url可分为两部分：
www.zhihu.com/api/v4/ques... + id + 后面一大堆 这里的id即为上面我们获取到的 dict_question['id'] 里面的id
我们分析返回的json 发现，更多回答在'next'属性里面

获取某一个话题的详细回答:

```
#单个问题抓取
def getZhiHuItemDetail(response):

    html = response.text
    dict_json = json.loads(html)
    dit_total = dict_json['paging']
    print('总共回答数：'+ str(dit_total['totals']))
```

```
            question_info = []  # 该问题所有数据
            global  count
            if count ==0 :
                print('问题： ' + dictBean['question']['title'])
                count = count + 1
            print('回答者：'+dictBean['author']['name'])
            print('个人签名：' + dictBean['author']['headline'])

            question_info.append('回答者：'+dictBean['author']['name'])
            question_info.append('个人签名：' + dictBean['author']['headline'])

            # 回答内容
            htmls = BeautifulSoup(dictBean['content'], "html.parser")
            print(htmls.get_text())
            question_info.append(htmls.get_text())
            print('')
            saveQuesInfo(question_info,dictBean['question']['title'],dit_total['totals'])

        is_end = dit_total['is_end']
        if is_end :
            print('.................结束.....................')
            # exit()
            startSpider()
        else:
            # getZhiHuItemDetail(dit_total['next'])
            regPoxy(dit_total['next'], 2)
```

保存某一详细回答到电脑本地

```
#保存某一个问题详情
def saveQuesInfo(question_info,title,totalNum):

    basePath=r"C:/Users/fp/Desktop/zhihu/"
    file = open(os.path.join(basePath)+"{}.txt".format(title), "a",encoding='utf-8')
    for ques in question_info:
        for i in range(len(ques)):
            file.write(ques[i])
        file.write('\n')
    file.write('\n')
    file.close()
```

全部代码如下:

```
import os
from bs4 import BeautifulSoup
import requests
import json
import sys
import random
sys.setrecursionlimit(10000)

#问题id
ques_id_list =[]

headers = {
    'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_3) AppleWebKit/537.36 (KHTM
    'referer': 'https: // www.zhihu.com /',
    'cookie': '_zap=5be1e1ef-530e-4b98-a2c5-a74abc65328c; __DAYU_PP=AnJrB73abrJaaAUnRnbqfff
}

poxy_list=[]

def initPoxy():
    poxy_list.append({"http":"223.111.131.100:8080"})
    poxy_list.append({"http": "218.60.8.99:3129"})
    poxy_list.append({"http": "39.137.107.98:80"})
    poxy_list.append({"http": "118.89.234.236:8787"})
    poxy_list.append({"http": "218.22.7.62:53281"})
    poxy_list.append({"http": "117.57.91.235:9999"})
    poxy_list.append({"http": "110.243.23.117:9999"})

def getZhiHu():
    baseurl = "https://www.zhihu.com/api/v3/feed/topstory/recommend?session_token=266ffeaeb
    url_list = []
    for num in range(1,3):
        url_list.append(baseurl + str(num*5))
    for url in url_list:
        regPoxy(url,1)

def regPoxy(url,num):
    response = requests.get(url, headers=headers, proxies=random.choice(poxy_list))
    if response.status_code == 200:
        if num == 1:
            getzhihutitle(response)
        else:
            getZhiHuItemDetail(response)
    else:
        regPoxy(url)
```

```
    html = response.text
    dict_json = json.loads(html)
    dit_list = dict_json['data']
    for ditc in dit_list:
        ditTarget = ditc['target']
        # 标题
        try:
            dict_question = ditTarget['question']
            print(dict_question['title'])
        except:
            print(ditTarget['title'])

        # 回答者
        print('回答者: ' + ditTarget['author']['name'])
        # 回答者个人签名
        print('个人签名: ' + ditTarget['author']['headline'])

        if dict_question['type'] == 'question':
            # 问题具体页面
            print('问题详细url:  https://www.zhihu.com/question/' + str(dict_question['id'])

        # ques_id_list.append(dict_question['id'])

        # 回答内容
        htmls = BeautifulSoup(ditTarget['content'], "html.parser")
        print(htmls.get_text())
        print('')

base_item_url_start = 'https://www.zhihu.com/api/v4/questions/'
base_item_url_end = '/answers?include=data%5B*%5D.is_normal%2Cadmin_closed_comment%2Creward

count = 0
#单个问题抓取
def getZhiHuItemDetail(response):

    html = response.text
    dict_json = json.loads(html)
    dit_total = dict_json['paging']
    print('总共回答数：'+ str(dit_total['totals']))

    dit_list = dict_json['data']
    for dictBean in dit_list:
        question_info = []  # 该问题所有数据
        global  count
        if count ==0 :
            print('问题： ' + dictBean['question']['title'])
```

```
            question_info.append('回答者：'+dictBean['author']['name'])
            question_info.append('个人签名：' + dictBean['author']['headline'])

            # 回答内容
            htmls = BeautifulSoup(dictBean['content'], "html.parser")
            print(htmls.get_text())
            question_info.append(htmls.get_text())
            print('')
            saveQuesInfo(question_info,dictBean['question']['title'],dit_total['totals'])

        is_end = dit_total['is_end']
        if is_end :
            print('.................结束.....................')
            # exit()
            startSpider()
        else:
            # getZhiHuItemDetail(dit_total['next'])
            regPoxy(dit_total['next'], 2)

#保存某一个问题详情
def saveQuesInfo(question_info,title,totalNum):

    basePath=r"C:/Users/fp/Desktop/zhihu/" #保存爬取的话题地址，自己更改
    file = open(os.path.join(basePath)+"{}.txt".format(title), "a",encoding='utf-8')
    for ques in question_info:
        for i in range(len(ques)):
            file.write(ques[i])
        file.write('\n')
    file.write('\n')
    file.close()


#爬取某一个问题
def startSpider():
    id = input('请输入想要查看的问题id：')
    if id != 'exit':
        url = base_item_url_start + id + base_item_url_end
        regPoxy(url,2)
    else:
        print('.................结束.....................')
        exit()

if __name__ == "__main__":
    initPoxy()
```

最后

如果对知乎推荐的话题不感兴趣，想爬取自己找的某一个话题到本地，可以直接网页打开该话题首页,copy一下话题id，然后输入，也可以爬取该话题全部回答。
如 话题 微信和 QQ 视频聊天会被腾讯后台监控吗？，打开网页 www.zhihu.com/question/36… 其话题id 为36789686，为question 后面的那串数字.

忧弧 · 11天前 · Python

Python 3.8.5 发布

5

头文件 · 1天前 · Python

面试必备：Python内存管理机制

4 1

豌豆花下猫 · 2天前 · Python

Python 为什么会有个奇怪的“...”对象？

2

前端小Ken · 2天前 · Python

爬虫（108）Python 3.8的超酷新功能（接近一万字，请耐心享用，而且建议收藏）

2

元宵大师 · 10天前 · Python

小散量化炒股记|不用追高！Python告诉你强势股回调介入的位置

6 9

MedusaSorcerer · 5天前 · Python

一个支持检索的日志记录器 --- eliot

2 1

MedusaSorcerer · 8天前 · Python

关于 Python-TK 小程序的 PAC 自动化问题

10