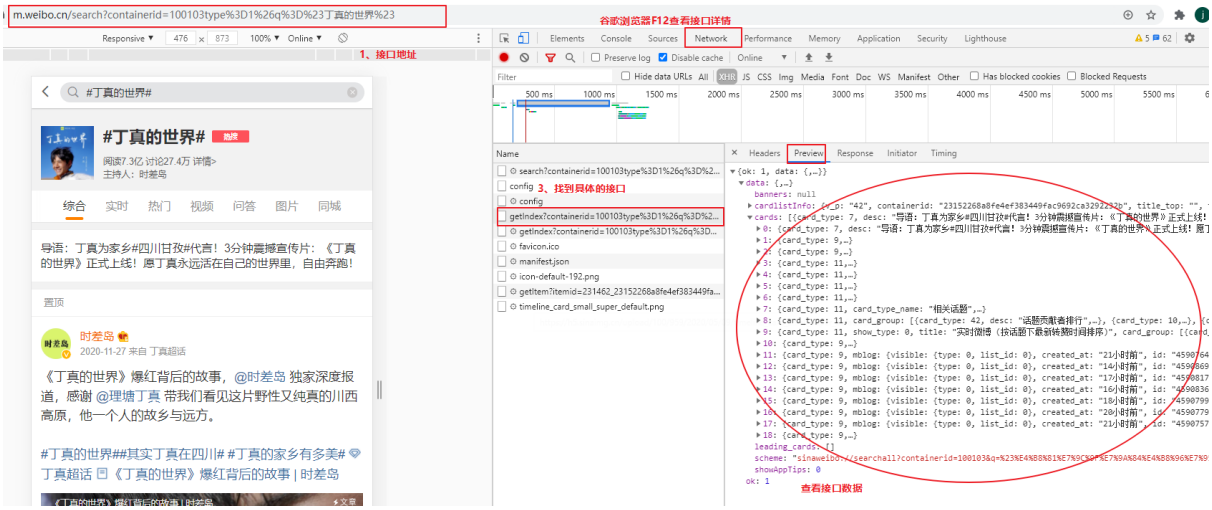
微博评论数据抓取

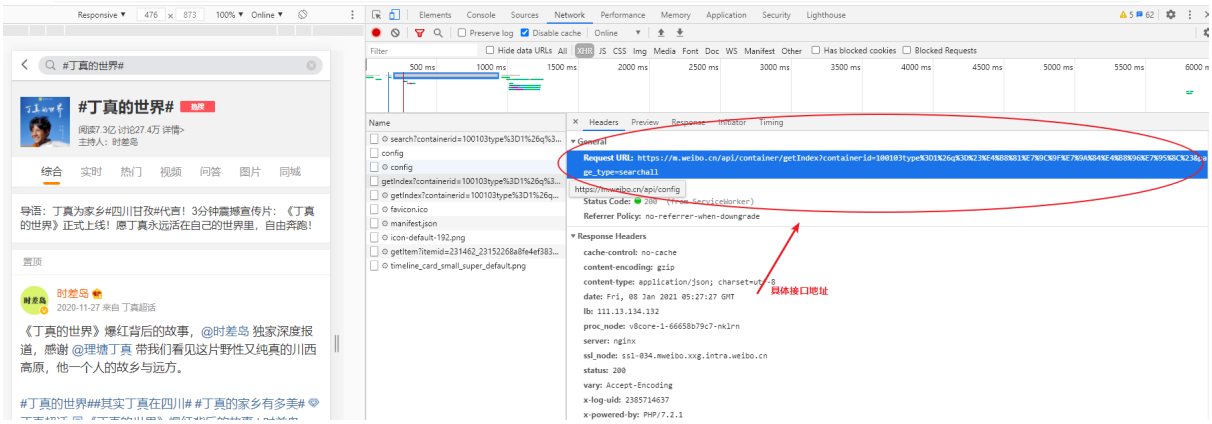
- **1、数据来源网站（手机端微博）**
  - https://m.weibo.cn/
  - 

- **2、数据内容**
  - **2.1 搜索关键字"#丁真#"**
    -

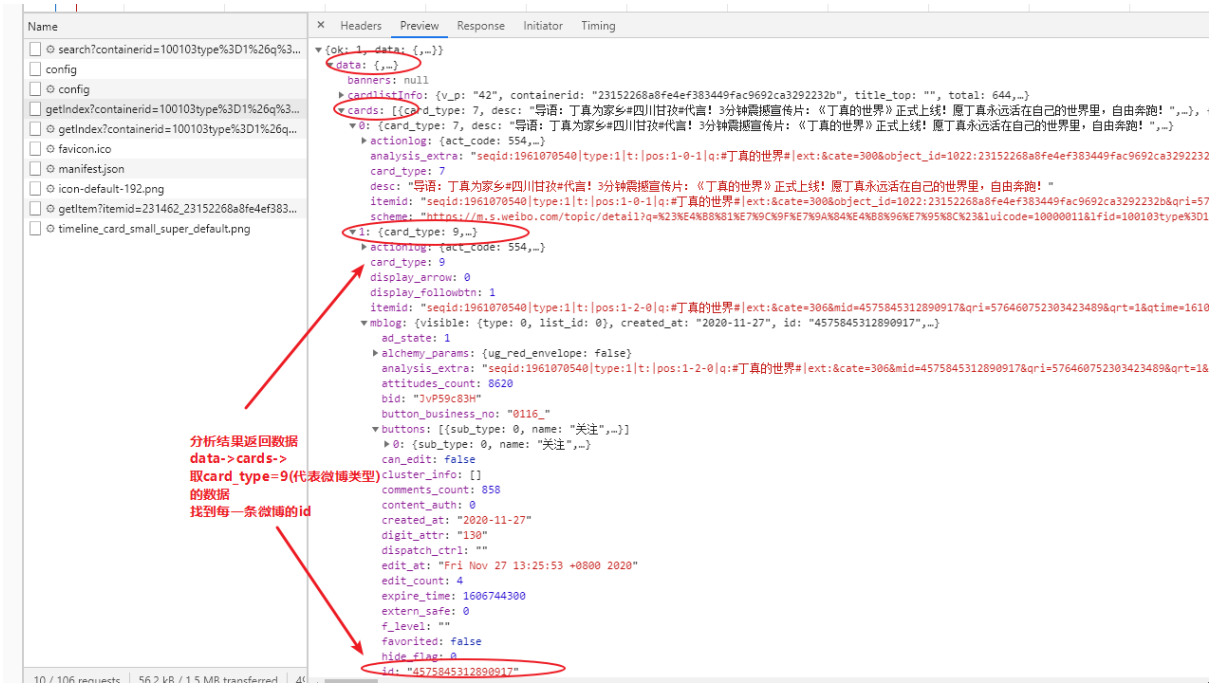
- **2.2 拿到浏览接口地址1**
  - "#丁真#" 的搜索结果，接口地址
    - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall%23%E4%B8%81%E7%9C%9F%23
  - **同样方法拿到其余四个地址**
    - #丁真的世界#
      - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%E7%9A%84%E4%B8%96%E7%95%8C%23&page_type=searchall
    - "#丁真说不要再p了#
      - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall
    - #四川为了丁真有多努力#
    - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E5%9B%9B%E5%B7%9D%E4%B8%BA%E4%BA%86%E4%B8%81%E7%9C%9F%E6%9C%89%E5%A4%9A%E5%8A%AA%E5%8A%9B%23&page_type=searchall
    - "#丁真所在国企负责人回应拒绝选秀#"
      - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall
- **2.3 以第一个接口为例（#丁真的世界#）**
  -



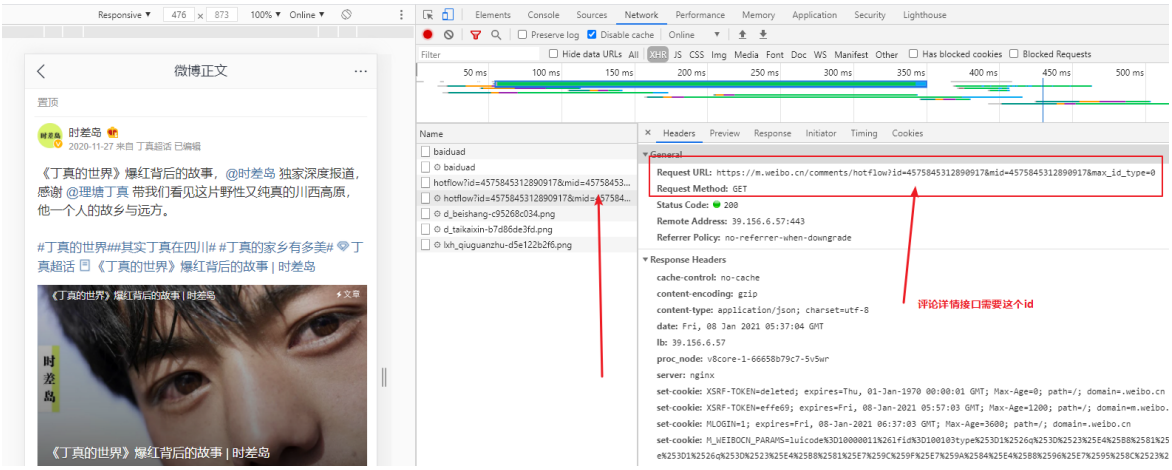
- **2.4 拿到具体的接口调用地址**
  - https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%E7%9A%84%E4%B8%96%E7%95%8C%23&page_type=searchall
  -

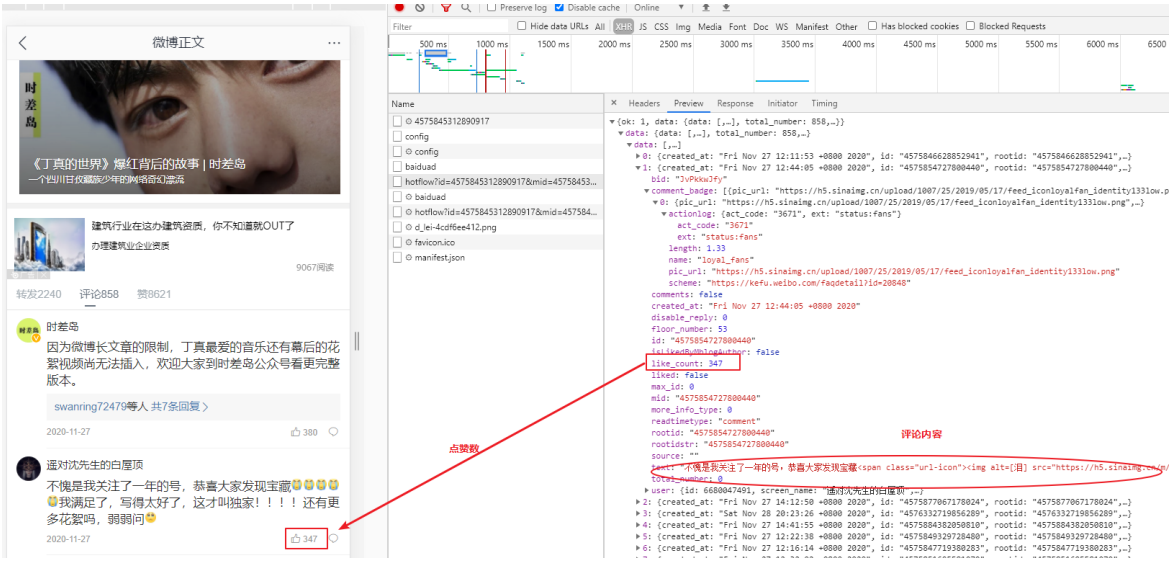


- **2.5 分析相关的返回结果参数**
  -



  - 为什么要这个id 呢，因为查看具体的微博评论需要传入这个id
    - 微博评论接口地址：https://m.weibo.cn/comments/hotflow?id=4575845312890917&mid=4575845312890917&max_id_type=0
    -



- 观察评论接口返回数据，找到点赞数和评论内容的参数
  -


  - 其他一些需要的参数也能找到哟

- **3、分析除了数据来源，剩下的就是通过计算机程序去自动抓取数据了**
  - 程序可以模拟接口调用，这里用Python实现的
    - **3.1 设置header**

```
headers = {
'Cookie': '_T_WM=53629218447; XSRF-TOKEN=db4d17; WEIBOCN_FROM=1110006030; MLOGIN=0; M_WEIBOCN_PARAMS=fid%3D100103type%253D1%2526q%253D%2525E4%2525B8%252581%2525E7%25259C%25259F%26uicode%3D10000011',
    'Referer': 'https://m.weibo.cn/detail/4312409864846621',    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36 Edg/87.0.664.66',    'X-Requested-With': 'XMLHttpRequest'}
```

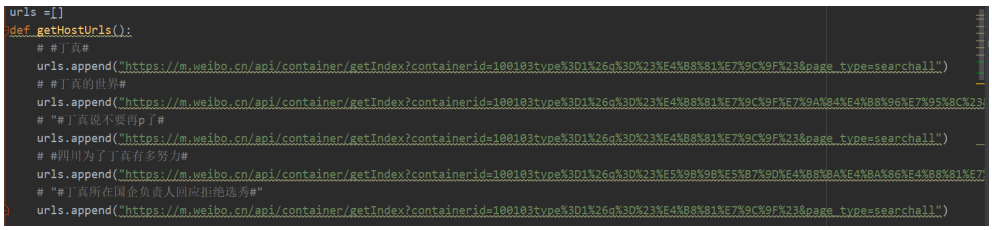
  -


```
headers = {
    'Cookie': '_T_WM=53629218447; XSRF-TOKEN=db4d17; WEIBOCN_FROM=1110006030; MLOGIN=0; M_WEIBOCN_PARAMS=fid%3D100103type%253D1%2526q%253D%2525
    'Referer': 'https://m.weibo.cn/detail/4312409864846621',
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36 Edg/87.0.
    'X-Requested-With': 'XMLHttpRequest'
}
```

    - **3.2 定义爬虫的地址（这里是固定的五个和丁真相关的话题）**

```
urls =[]
def getHostUrls():
# #丁真#    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")
# #丁真的世界#    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%E7%9A%84%E4%B8%96%E7%95%8C%23&page_type=searchall")
# "#丁真说不要再p了#    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")
# #四川为了丁真有多努力#    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E5%9B%9B%E5%B7%9D%E4%B8%BA%E4%BA%86%E4%B8%81%E7%9C%9F%E6%9C%89%E5%A4%9A%E5%8A%AA%E5%8A%9B%23&page_type=searchall")
```

"#丁真所在国企负责人回应拒绝选秀#" urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")

- 

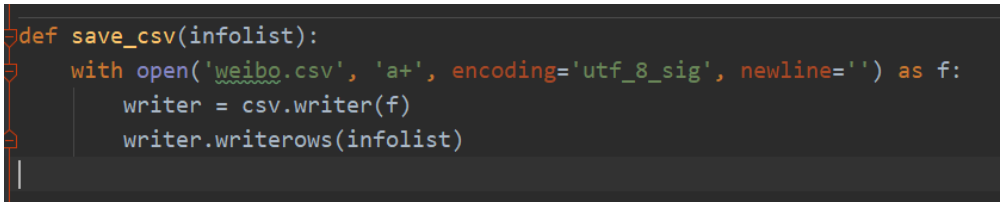
```
urls =[]
def getHostUrls():
    # #丁真#
    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")
    # #丁真的世界#
    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%E7%9A%84%E4%B8%96%E7%95%8C%23
    # "#丁真说不要再p了#
    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")
    # #四川为了丁真有多努力#
    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E5%9B%9B%E5%B7%9D%E4%B8%BA%E4%BA%86%E4%B8%81%E7
    # "#丁真所在国企负责人回应拒绝选秀#"
    urls.append("https://m.weibo.cn/api/container/getIndex?containerid=100103type%3D1%26q%3D%23%E4%B8%81%E7%9C%9F%23&page_type=searchall")
```

- **3.3 详情的爬虫代码 (参考的百度的解析response的代码，自己不太想写了)**

def spider(page_num,hostUrl):
# main_url为要爬取的主页地址 if page_num:
    main_url = hostUrl + '&page=' + str(page_num)
# 微博的分页机制是每页10条微博 try:
    r = requests.get(url=main_url, headers=headers)
    r.raise_for_status()
except Exception as e:
print("爬取失败", e)
return 0 result_json = json.loads(r.content.decode('utf-8'))
  info_list = []
for card in result_json['data']['cards']:
    info_list_sub = []
if card.get("mblog"):
        info_list_sub.append(card['mblog']['attitudes_count']) # 获赞数 info_list_sub.append(card['mblog']['comments_count']) # 评论数 info_list_sub.append(card['mblog']['reposts_count']) # 转发数 if page_num == 1:
            info_list_sub.append(card['mblog']['created_at']) # 发博时间 elif '2018' not in card['mblog']['created_at']:
            info_list_sub.append(card['mblog']['created_at'])
else:
print("2019年微博爬取完毕")
break info_list_sub.append(card['mblog']['weibo_position']) # 是否原创 if card['mblog'].get('raw_text'):
            info_list_sub.append(card['mblog']['raw_text']) # 微博内容 else:
            info_list_sub.append(card['mblog']['text'])
# if card['mblog']['source'] == '': # info_list_sub.append(None) # else: # info_list_sub.append(card['mblog']['source']) # time.sleep(random.randint(4, 6)) # 每爬取一条微博暂停4到6秒，防反爬 info_list.append(info_list_sub)
else:
continue return info_list

- **3.4 最终保存到csv文件中**

def save_csv(infolist):
with open('weibo.csv', 'a+', encoding='utf_8_sig', newline='') as f:
    writer = csv.writer(f)
    writer.writerows(infolist)

- 

```
def save_csv(infolist):
    with open('weibo.csv', 'a+', encoding='utf_8_sig', newline='') as f:
        writer = csv.writer(f)
        writer.writerows(infolist)
```

- **3.5 定义运行的main方法**

def main(num):
for hostUrl in urls:
for i in range(1, num+1):
        information = spider(i,hostUrl)
        save_csv(information)
print("第%s页爬取完毕" % i)

-

```
def main(num):
    for hostUrl in urls:
        for i in range(1, num+1):
            information = spider(i,hostUrl)
            save_csv(information)
            print("第%s页爬取完毕" % i)
```

- **3.6 启动代码**

print("### 开始爬取微博 ")
# 1、封装地址到urls中getHostUrls()
# 2、遍历封装好的urls，循环查询接口，获取评论数if __name__ == '__main__':
main(10)

```
print("### 开始爬取微博 ")
# 1、封装地址到urls中
getHostUrls()
# 2、遍历封装好的urls，循环查询接口，获取评论数
if __name__ == '__main__':
    main(10)
```

- **3.7 补充**
  - 运行代码的时候，需要在.py 的同级建立一个weibo.csv文件
  - 微博有反爬机制，可以设置线程休眠
    - 代码中是注释的版本
    - **time.sleep(random.randint(4, 6)) # 每爬取一条微博暂停4到6秒，防反爬**

- **4、git源码地址**
  - https://github.com/xjdm/pythonWorkspace/blob/master/spider.py