计算机网络爬虫作业说明文档

1、概述

本次计算机网络爬虫作业，我选择了**飞常准**作为爬取的目标，飞常准提供了较为全面的国内外航班列表和航班的数据，但是缺点是飞常准有较强的反爬虫机制。

为了绕过网站的反爬虫机制，我没有选择助教推荐的`Scrapy`来编写代码，而是选择了灵活度更高的python自带的原生的`requests`包和`xpath`工具作为我的工具来爬取数据。

我选择了讯代理搭建ip代理池来绕过网站的反爬虫机制。

2、爬虫使用说明

2.1、爬虫项目的使用方法

- 解压缩项目文件夹后，压缩包下共有四个文件。
    - variflight.py 爬虫的主体文件，爬取网页和获取需要的数据，并以json格式将数据存入以当天日期命名的json文件中，例如`2019-12-10-flight.json`。
    - xundaili.py 讯代理文件，负责生成一个请求头，该请求头可以将我们对网站的请求先转发到讯代理的服务器，然后再由讯代理帮我们随机选择ip再去转发请求，这样可以避免本机爬取网站的时候被封禁ip。
    - requirements.py 需要用到的python库。
    - 2020-12-09-flight.json 爬取的数据示范。
- 首先运行

    ```
    pip install requirements.py
    ```

    安装所需要的python库。

    接着直接运行`variflight.py`文件即可。

    终端会打印一些东西，可以忽略，爬取的数据会存储在`{当天日期}-flight.json`文件中。

2.2、绕过网站的反爬虫机制

飞常准网站的反爬虫机制主要有三点。

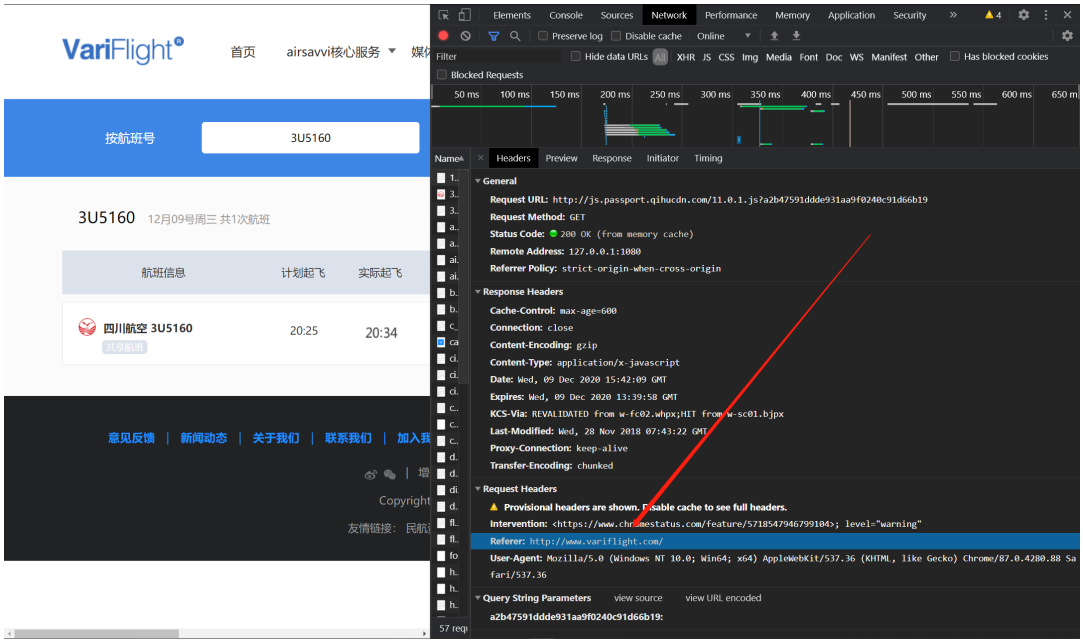
（1）频繁访问会封禁ip。

（2）如果想访问网址A页面下的网址B，只能从网址A跳转过去。如果直接在浏览器输入网址B，会返回406错误，当然，用requests也同样会直接返回406错误。

（3）我们想要的数据，例如起飞时间，到达情况等，在网站上都是以图片的形式展示，如果直接爬取图片的话还需要用图像识别一下才能获得我们需要的数据，这大大加剧了我们获得数据的难度。

为此，我在爬取网站的时候用了下述几种方法来绕过上述反爬机制。

- 用讯代理来搭建自己的ip池，每次访问随机选择ip，这样就不会被封禁ip了。
- 在仔细研究了为什么不能直接访问网址B，而只能是从网站A跳转过去到网址B时。最后终于找到了原因，在网站A访问网址B时，会在**HTTP请求头上加上一个Referer字段**。



这个Referer字段里面放的就是网址A的地址。所以我在访问时在HTTP访问的headers中加入这一字段，就能成功访问网址B了。

```
'''
variflight.py中下面两行代码都是为了解决这一问题。
'''
#代码41行
headers['Referer'] = 'http://www.variflight.com/sitemap.html?AE71649A58c77='
#代码72行
headers['Referer'] = flight_url
```

- 在仔细观察网站的结构，仔细阅读网站的源码后，以及查阅网络上其他爬取该网站的经验后，我发现了其实他的数据虽然被转换成了图片，但是他是有一个数据的获取接口，但是由于每架次航班的数据获取接口都不一样，主要由本次航班的航班号、出发机场的代码、目的机场的代码、起飞时间决定，并且这些数据可以由网站上另外一个链接得到。所以我们只要找到这个链接，提取数据，构造好伪造的数据接口链接，去访问，就能获得我们想要的数据了。

```
'''
这是构造数据接口url的主要代码
'''
        if flag:
            selector = etree.HTML(r.text)
            next_json_list = selector.xpath('//iframe/@src')
            if len(next_json_list) < 1 :
                return None
            else:
                #获得的url: https://flightadsb.variflight.com/flight-
playback/3U2014/HRB/HGH/1607345700
                url = next_json_list[0]
                num = len('https://flightadsb.variflight.com/flight-playback/')
                #截取后半部分：3U2014/HRB/HGH/1607345700
                data_url = url[num:]
                values = data_url.split('/')
                fnum = values[0]        #航班号
                forg = values[1]        #出发地机场代码
                fdst = values[2]        #目的机场代码
                ftime = values[3]       #时间戳
                #构造数据接口地址
```

```
                json_url = 'https://adsbapi.variflight.com/adsb/index/flight?
lang=zh_CN&fnum={fnum}&time={time}&forg={forg}&fdst=
{fdst}'.format(fnum=fnum,time=ftime,forg=forg,fdst=fdst)
                return json_url
        else:
            return None
```

2.3、爬虫项目的几点说明

- 飞常准网站的反爬虫机制做的很好，网站的**爬虫协议**中也明确说明了不许爬取该网站的任何数据。本次爬取网站的数据并没有任何商用目的，只是单纯为了完成计算机网络的大作业。
- 在尝试搭建ip代理池时，我首先是爬取了快代理网站的免费的ip作为代理ip，但是后续实际效果并不好，ip的质量很差，访问也很慢。后来还是决定使用付费服务，最后我选择了讯代理提供的动态转发套餐。20元10万次访问，使用期限是半年。在写代码调试期间和爬取示范数据差不多已经使用了快1万次。还剩下很多，助教测试的时候应该还可以在爬取五万条数据左右。
- 每次运行代码会默认爬取前2000个航班的数据，可以在variflight.py中main函数下修改大小。

```
flight_nums = 2000  #默认爬取数据大小，可以修改大小，最大为 len(flight_list) = 6000
```

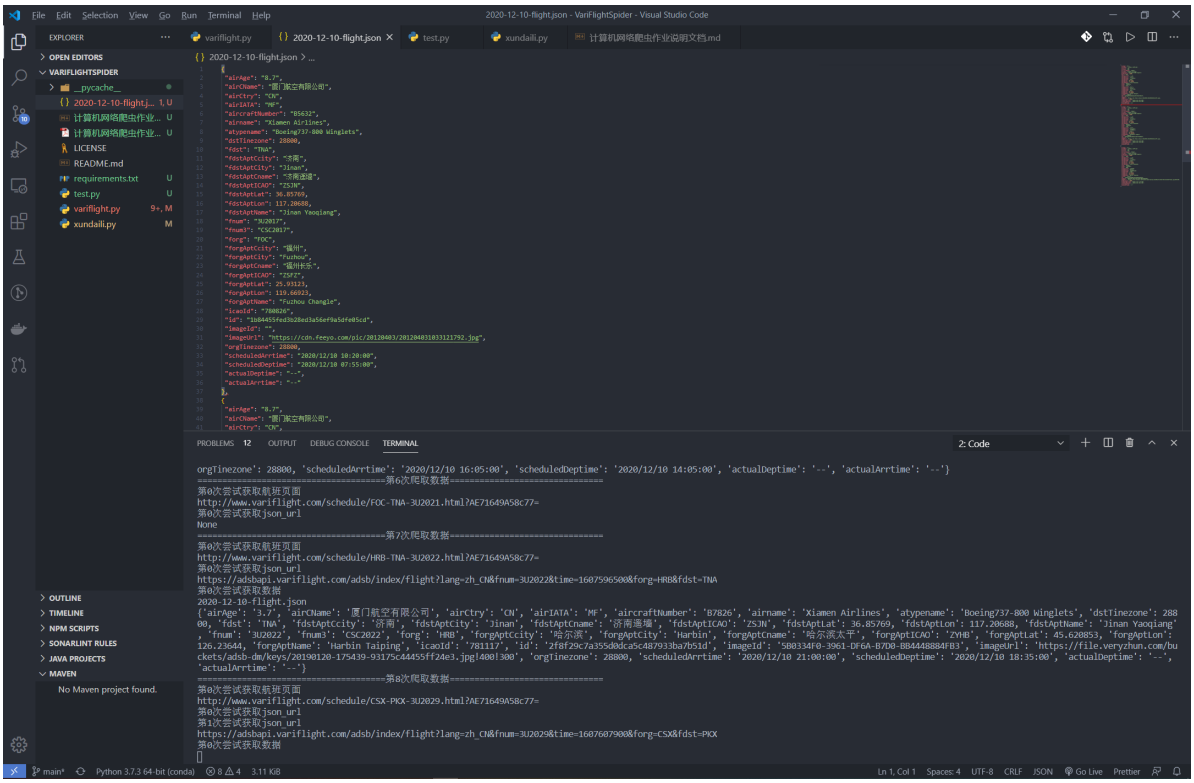
3、数据介绍

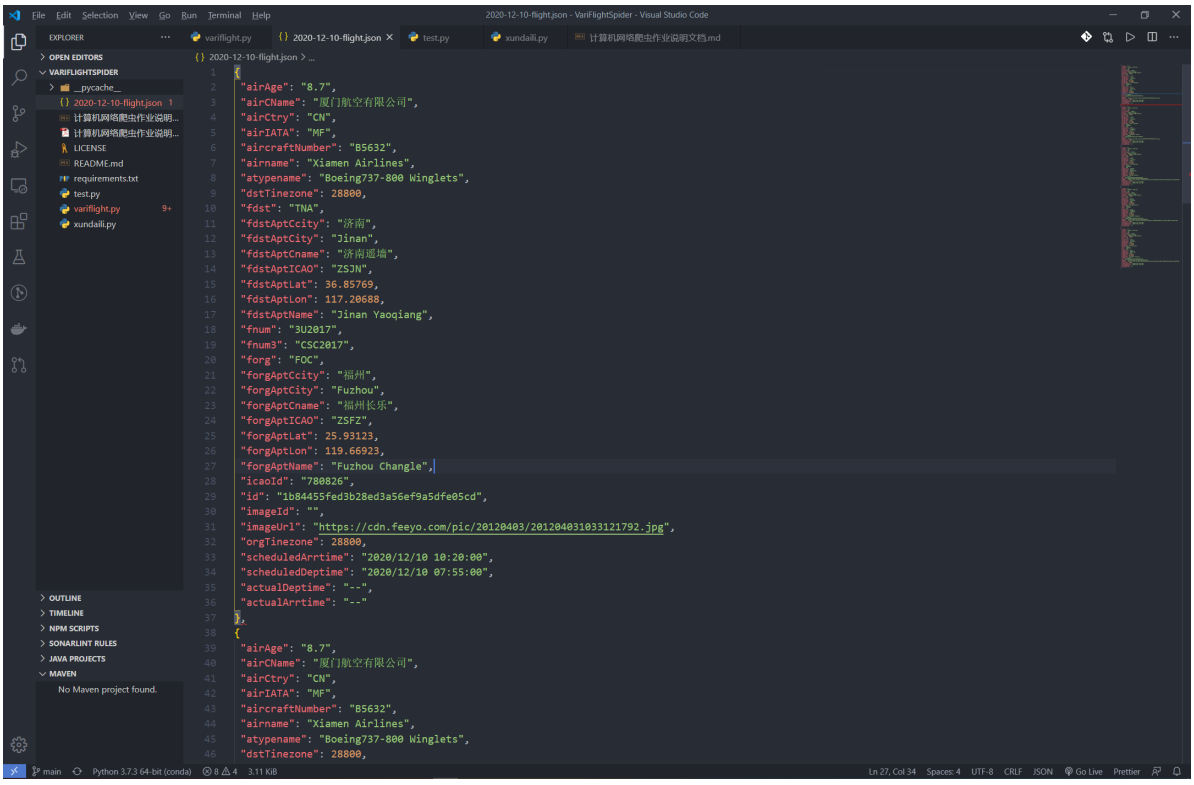
示范数据存在`2019-12-10-flight.json`中。我们截取其中一条讲述。

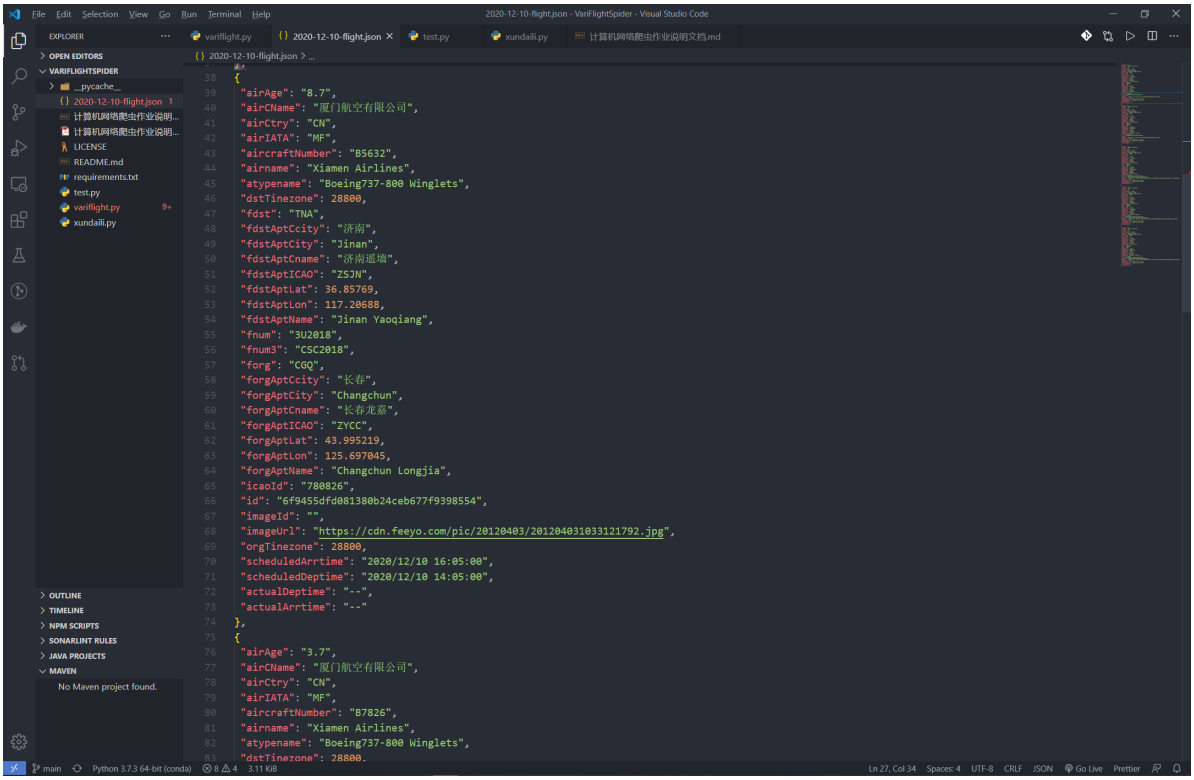
```
{
 "actualArrtime": 2020/12/10 20:29:00,          //航班实际到达时间
 "actualDeptime": 2020/12/10 20:29:00,          //航班实际起飞时间（--表示还未起飞）
 "airAge": "5.3",                               //航班所用飞机机龄
 "airCName": "厦门航空有限公司",                 //航空公司中文名
 "airCtry": "CN",                               //航空公司所属国家
 "airIATA": "MF",                               //航空公司代码
 "aircraftNumber": "B6487",                     //飞机代码
 "airname": "Xiamen Airlines",                  //航空公司英文名
 "atype": "B738",                               //飞机机型
 "atypename": "Boeing737-800 Winglets",         //飞机名称
 "dstTinezone": 28800,                          //目的地时区
 "estimatedArrtime": 1607516880,                //预计到达时间
 "fdst": "CSX",                                 //目的地代码
 "fdstAptCcity": "长沙",                        //目的地中文名
 "fdstAptCity": "Changsha",                     //目的地英文名
 "fdstAptCname": "长沙黄花",                    //目的机场中文名
 "fdstAptICAO": "ZGHA",                         //目的机场英文名
 "fdstAptLat": 28.193336,                       //目的机场经度
 "fdstAptLon": 113.21459,                       //目的机场纬度
 "fdstAptName": "Changsha Huanghua",            //目的机场英文名
 "fnum": "3U2030",                              //航班号
 "fnum3": "CSC2030",                            //航空公司航班编号
 "forg": "PKX",                                 //出发地城市代码
 "forgAptCcity": "北京",                        //出发地中文名
 "forgAptCity": "Beijing",                      //出发地英文名
 "forgAptCname": "北京大兴",                    //出发机场中文名
 "forgAptICAO": "ZBAD",                         //出发机场ICAO代码
 "forgAptLat": 39.509167,                       //出发机场经度
 "forgAptLon": 116.410556,                      //出发机场纬度
 "forgAptName": "Beijing Daxing",               //出发机场英文名
 "ftype": "B738",                               //飞机机型
 "icaoId": "780E47",                            //ICAO代码
 "id": "41abf41b0c1797c1df556739ff27484e",      //id
```

```
  "imageId": "941C69E8-CE8F-9EB5-FF72-BD4DCAA1F656", //飞机图片id
  "imageUrl": "https://file.veryzhun.com/buckets/wxapp/keys/20180823-164616-
90105b7e745877cfd.jpg!400!300",                    //飞机图片地址
  "orgTinezone": 28800,                            //出发地时区
  "scheduledArrtime": 2020/12/10 20:35:00,         //规划到达时间
  "scheduledDeptime": 2020/12/10 18:00:00          //规划起飞时间
 },
```

4、爬取结果截图






File Edit Selection View Go Run Terminal Help
2020-12-10-flight.json - VariFlightSpider - Visual Studio Code
EXPLORER
OPEN EDITORS
VARIFLIGHTSPIDER
__pycache__
2020-12-10-flight.json 1
计算机网络爬虫作业说明...
计算机网络爬虫作业说明...
LICENSE
README.md
requirements.txt
test.py
variflight.py 9+
xundaili.py
variflight.py
2020-12-10-flight.json
test.py
xundaili.py
计算机网络爬虫作业说明文档.md
OUTLINE
TIMELINE
NPM SCRIPTS
SONARLINT RULES
JAVA PROJECTS
MAVEN
No Maven project found.
{
"airAge": "8.7",
"airCName": "厦门航空有限公司",
"airCtry": "CN",
"airIATA": "MF",
"aircraftNumber": "B5632",
"airname": "Xiamen Airlines",
"atypename": "Boeing737-800 Winglets",
"dstTinezone": 28800,
"fdst": "TNA",
"fdstAptCcity": "济南",
"fdstAptCity": "Jinan",
"fdstAptCname": "济南遥墙",
"fdstAptICAO": "ZSJN",
"fdstAptLat": 36.85769,
"fdstAptLon": 117.20688,
"fdstAptName": "Jinan Yaoqiang",
"fnum": "3U2018",
"fnum3": "CSC2018",
"forg": "CGQ",
"forgAptCcity": "长春",
"forgAptCity": "Changchun",
"forgAptCname": "长春龙嘉",
"forgAptICAO": "ZYCC",
"forgAptLat": 43.995219,
"forgAptLon": 125.697045,
"forgAptName": "Changchun Longjia",
"icaoId": "780826",
"id": "6f9455dfd081380b24ceb677f9398554",
"imageId": "",
"imageUrl": "https://cdn.feeyo.com/pic/20120403/20120403103312l792.jpg",
"orgTinezone": 28800,
"scheduledArrtime": "2020/12/10 16:05:00",
"scheduledDeptime": "2020/12/10 14:05:00",
"actualDeptime": "--",
"actualArrtime": "--"
},
{
"airAge": "3.7",
"airCName": "厦门航空有限公司",
"airCtry": "CN",
"airIATA": "MF",
"aircraftNumber": "B7826",
"airname": "Xiamen Airlines",
"atypename": "Boeing737-800 Winglets",
"dstTinezone": 28800,
main
Python 3.7.3 64-bit (conda)
8 4
3.11 KiB
Ln 27, Col 34
Spaces: 4
UTF-8
CRLF
JSON
Go Live
Prettier