**IP被限制的解决方法：**

1. 放慢抓取速度，减小对于目标网站造成的压力。但是这样会减少单位时间类的抓取量。
2. 第二种方法是通过设置代理IP等手段，突破反爬虫机制继续高频率抓取。但是这样需要多个稳定的代理IP。

3. 最有效的方法，还是动态切换本地IP（首先，买一个可以动态切换IP的阿里云服务器。然后，用十几行python就能实现。）<https://zhuanlan.zhihu.com/p/25285987>

**爬虫代理ip池的建立方式**

**{1.免费的代理网站有反爬虫策略**

**2.免费的代理ip慢，很不稳定}**

参考：<https://zhuanlan.zhihu.com/p/24130550>

在爬取网站信息的过程中，有些网站为了防止爬虫，可能会限制每个ip的访问速度或访问次数。对于限制访问速度的情况，我们可以通过time.sleep进行短暂休眠后再次爬取。对于限制ip访问次数的时候我们需要通过代理ip轮换去访问目标网址。所以建立并维护好一个有效的代理ip池也是爬虫的一个准备工作。

网上提供免费代理ip的网址很多，下面我们以西刺网站为例来建立一个有效的代理ip池。

项目流程：



***第一步：构造请求代理ip网站链接***

def get\_url(url): # 国内高匿代理的链接

url\_list = []

for i in range(1,100):

url\_new = url + str(i)

url\_list.append(url\_new)

return url\_list

get\_url ：生成要爬取目标网址的链接

***第二步：获取网页内容***

def get\_content(url): # 获取网页内容

user\_agent = 'Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.22 Safari/537.36 SE 2.X MetaSr 1.0'

headers = {'User-Agent': user\_agent}

req = urllib.request.Request(url=url, headers=headers)

res = urllib.request.urlopen(req)

content = res.read()

return content.decode('utf-8')

get\_content：接受的参数是传入的目标网站链接

***第三步：提取网页中ip地址和端口号信息***

def get\_info(content): # 提取网页信息 / ip 端口

datas\_ip = etree.HTML(content).xpath('//table[contains(@id,"ip\_list")]/tr/td[2]/text()')

datas\_port = etree.HTML(content).xpath('//table[contains(@id,"ip\_list")]/tr/td[3]/text()')

with open("data.txt", "w") as fd:

for i in range(0,len(datas\_ip)):

out = u""

out += u"" + datas\_ip[i]

out += u":" + datas\_port[i]

fd.write(out + u"\n") # 所有ip和端口号写入data文件

get\_info：接收从get\_content函数传来的网页内容，并使用etree解析出ip和端口号，将端口号和ip写入data.

***第四步：验证代理ip的有效性***

def verif\_ip(ip,port): # 验证ip有效性

user\_agent ='Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.22 Safari/537.36 SE 2.X MetaSr 1.0'

headers = {'User-Agent':user\_agent}

proxy = {'http':'http://%s:%s'%(ip,port)}

print(proxy)

proxy\_handler = urllib.request.ProxyHandler(proxy)

opener = urllib.request.build\_opener(proxy\_handler)

urllib.request.install\_opener(opener)

test\_url = "https://www.baidu.com/"

req = urllib.request.Request(url=test\_url,headers=headers)

time.sleep(6)

try:

res = urllib.request.urlopen(req)

time.sleep(3)

content = res.read()

if content:

print('that is ok')

with open("data2.txt", "a") as fd: # 有效ip保存到data2文件夹

fd.write(ip + u":" + port)

fd.write("\n")

else:

print('its not ok')

except urllib.request.URLError as e:

print(e.reason)

verif\_ip：使用ProxyHandler建立代理，使用代理ip访问某网址，查看是否得到响应。如数据有效，则保存到data2.txt文件

***最后：调用各个函数***

if \_\_name\_\_ == '\_\_main\_\_':

url = 'http://www.xicidaili.com/nn/'

url\_list = get\_url(url)

for i in url\_list:

print(i)

content = get\_content(i)

time.sleep(3)

get\_info(content)

with open("dali.txt", "r") as fd:

datas = fd.readlines()

for data in datas:

print(data.split(u":")[0])

# print('%d : %d'%(out[0],out[1]))

verif\_ip(data.split(u":")[0],data.split(u":")[1])

得到爬取结果

-----------------------------------------------------------分割线-------------------------------------------------------------

本程序运行环境: Python 3.5.2