PYTHON爬虫

计算机科学与技术（大数据技术与应用） 程越 201711133007

淘宝

一、思路要点：

1.完成翻页操作，尝试爬取下一页商品（经观察发现，第一页链接后接&s=44，第二页后接&s=88，猜测每页有44个商品，以此规律设计代码）

2.爬取当页商品的一些基本信息（用正则表达式查找）

3.本来设计了利用正则表达式找出该商品页面，再用BeautifulSoup解析选取几条评论输出.但后来发现商品附带信息里给的网址都是错误的，无法到达指定网页，之后尝试find\_all(‘a’),却发现包含商品链接的代码根本不在html里书写，所以就放弃了这个想法。原先设计代码如下。

hlt = re.findall(r'\"detail\_url\"\:\".\*?\"', html)

p=getHTMLText(hlt[i].split(':')[1])

soup = BeautifulSoup(p, 'html.parser')

comment=soup.select(‘.tm-rate-fulltxt’)[0].string

4.之后尝试在b站、新浪新闻、mooc网中尝试3，但均是同样情况。

import requests

from bs4 import BeautifulSoup

url="https://search.bilibili.com/all?keyword=游戏"

page=requests.get(url)

html=page.text

soup=BeautifulSoup(html,'html.parser')

c=soup.find\_all("a")

print（c)

结果：

[<a class="t" href="//www.bilibili.com" title="主站"><i class="bili-icon"></i>主站</a>, <a class="t" href="//h.bilibili.com" target="\_blank" title="画友">画友</a>, <a class="t" href="//www.bilibili.com/audio/home/?type=10" target="\_blank" title="来探索bilibili音乐的世界吧~">音频</a>, <a class="t" href="//game.bilibili.com" target="\_blank" title="游戏中心">游戏中心</a>, <a class="t" href="//live.bilibili.com" target="\_blank" title="直播">直播</a>, <a class="t" href="//show.bilibili.com/platform/home.html?msource=pc\_web" target="\_blank" title="会员购">会员购</a>, <a class="t" href="//bml.bilibili.com/2018/index.html#/bmlbj" target="\_blank" title="BML">BML<i class="bml-beijing"></i></a>, <a class="t" href="//www.bilibili.com/moe/2018/jp/home" target="\_blank" title="萌战">萌战</a>, <a class="t" href="//app.bilibili.com" id="header-mobile-app" target="\_blank" title="下载APP">下载APP</a>, <a class="u-link" href="//member.bilibili.com/v/video/submit.html" target="\_blank">投 稿</a>, <a class="search-logo" href="javascript:;"></a>, <a>清空搜索历史</a>]

并不含有包含视频地址的代码，而在源代码中搜索时可以清晰的看到，地址包含在a标签中。

只能放弃，不知道是否是自己欠缺知识

二、源码

import requests

from bs4 import BeautifulSoup

import re

def getHTMLText(url):

try:

page = requests.get(url, timeout=30)

page.raise\_for\_status()

page.encoding = page.apparent\_encoding

return page.text

except:

return "爬取失败"

def parsePage(ilt,html):

try:

plt=re.findall(r'\"view\_price\"\:\"[\d\.]\*\"',html)

tlt=re.findall(r'\"raw\_title\"\:\".\*?\"',html)

hlt = re.findall(r'\"view\_sales\"\:\".\*?\"', html)

for i in range(len(plt)):

price=eval(plt[i].split(':')[1])

title = eval(tlt[i].split(':')[1])

num = eval(hlt[i].split(':')[1])

ilt.append([price,title,num])

except:

print("")

def printGoodsList(ilt):

tplt="{:4}\t{:8}\t{:50}\t{:16}"

print(tplt.format("序号","价格","商品名称","付款人数"))

count=0

for g in ilt:

count+=1

print(tplt.format(count,g[0],g[1],g[2]))

def main():

goods='八音盒'

page=2

start\_url=“http://s.taobao.com/search?q="+goods

infoList=[]

for i in range(page):

try:

url=start\_url+’&s='+str(44\*i)

html=getHTMLText(url)

parsePage(infoList,html)

except:

continue

printGoodsList(infoList)

main()

三、最终结果

见淘宝结果.txt