

# 李顺立\_2018210765\_L7

## 初识爬虫

- 导入模块
- 定义爬虫函数
- 获得前5页数据
- 总结

## 导入模块

```
In [1]: import requests           #打开链接
        from bs4 import BeautifulSoup #解码链接
        from collections import OrderedDict #特定字典?
        from urllib.parse import urlencode #输入内容转码
        import time                 #获取时间
        import pandas as pd
        import numpy as np
```

## 定义爬虫函数

```
In [2]: def sina_search(page=2):
        """Function to get web list pages for a given keyword and page number.

        Args:
            keywords: manual input.
            page: The page number, default 2.

        Returns:
            newsData: A dataframe of the contents in sina web about keywords.

        """

        import time as TM

        ##输出如期信息
        print(TM.strftime('现在是北京时间: %Y-%m-%d %A %H:%M:%S', TM.localtime(TM.time()))))

        ###手动输入需要查询的内容
        key_words = input('请输入新闻关键词 (空格分开, 回车结束): ')
        key_word = re.split(r'\s', key_words) #按照空格分开

        #初始化时间
        time_begin = TM.time()

        k = 0 #初始化数据框的index
        newsData = pd.DataFrame(columns = ['title', 'date', 'time', 'source', 'abstract', 'url', 'content'])

        ###第一层循环: 针对输入的词
        for compRawStr in key_word:
            time_keyword = TM.time()
            # Dealing with character encoding
            comp = compRawStr.encode('gbk') #解码名字
            d = {'q': comp}
            pname = urlencode(d) #名称编码

            count_info = 0 #记录爬取数据条数
            ###第二层循环: 针对每一个词的页数进行循环
            for i in range(1, page+1): #从第一页开始
                href = 'http://search.sina.com.cn/?%s&range=all&c=news&sort=time&col=&source=&from=&country=&size=&time=&a=&page=%s'%(pname, i) # comp -> first %s; page
                # -> 2nd %s; col=1_7 -> financial news in sina
                html = requests.get(href) #打开链接
                # Parsing html
                soup = BeautifulSoup(html.content, 'html.parser') #解码链接
                divs = soup.findAll('div', {"class": "box-result clearfix"}) #解码后找到相应内容

                ###第三个循环: 对每一页的内容进行循环
                for div in divs: #找到各个文件下的东西
                    head = div.findAll('h2')[0] #标题和链接
                    # News title
                    titleinfo = head.find('a') #进入子目录
                    title = titleinfo.get_text() #获得a的内容: 即标题
                    # News url
                    url = titleinfo['href'] #获得a标题链接
                    # Other info

                    otherinfo = head.find('span', {"class": "fgray_time"}).get_text() #其他信息
                    source, date, time = otherinfo.split() #空格分隔

                    # News abstract
                    abstract = div.find('p', {"class": "content"}).get_text() #摘要文本
```

```
##找到链接中的具体内容
content_html = requests.get(url)  #打开标题链接
content_soup = BeautifulSoup(content_html.content, 'html.parser')  #解码链接
content = content_soup.find('div', {'class': 'article'}).get_text()  #获得内容

newsData.loc[k,:] = [title, date, time, source, abstract, url, content]  #将数据放在一个数据框中

k += 1 #index更新
count_info += 1
print('用时: %.4fs,'%(TM.time() - time_keyword), '找到有关【%s】数据%s页, 共%s条。'%(compRawStr, page, count_info))
print('-----'*8, '\n用时: %.4fs共爬取%s条数据。'%(TM.time() - time_begin, k))
return newsData
```

获取前5页数据

```
In [3]: news_about_finance = sina_search(5)  #找到新浪新闻前5页内容
```

现在是北京时间: 2019-04-27 Saturday 14:29:54  
请输入新闻关键词（空格分开，回车结束）：金融市场 银行大事件 证券新闻 股票价格走势

Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.

用时: 17.0438s, 找到有关【金融市场】数据5页, 共25条。

Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.

用时: 19.6730s, 找到有关【银行大事件】数据5页, 共25条。

Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.

用时: 19.1747s, 找到有关【证券新闻】数据5页, 共25条。

Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.  
Some characters could not be decoded, and were replaced with REPLACEMENT CHARACTER.

用时: 19.1373s, 找到有关【股票价格走势】数据5页, 共25条。

-----  
用时: 75.0352s共爬取100条数据。



---

## 总结

- 该爬虫函数只能爬取新浪网的新闻内容，对于其他网页的新闻无法获得，必须重新定义函数。
- 对于新浪网的新闻，如果新闻的主体内容是音频文件，则无法获得相关资源，爬取能力亟待提高。
- 一定有更好的方法获得数据。