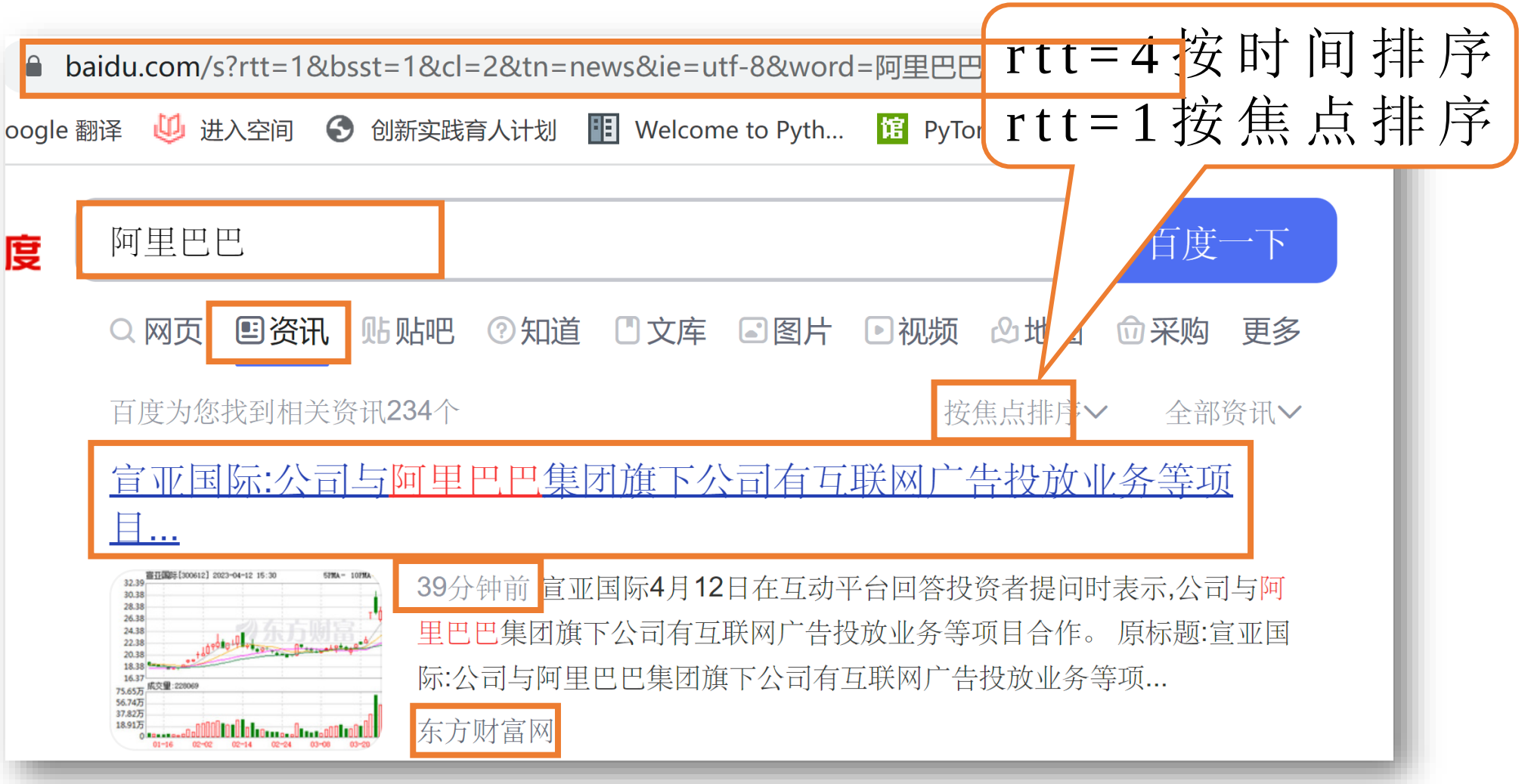
Python与金融数据挖掘(8)

文欣秀

wenxinxiu@ecust.edu.cn

案例分析


rtt=4 按时间排序
rtt=1 按焦点排序
baidu.com/s?rtt=1&bsst=1&cl=2&tn=news&ie=utf-8&word=阿里巴巴
oogle 翻译
进入空间
创新实践育人计划
Welcome to Pyth...
PyTor
阿里巴巴
百度一下
网页
资讯
贴吧
知道
文库
图片
视频
采购
更多
百度为您找到相关资讯234个
按焦点排序
全部资讯
宣亚国际:公司与阿里巴巴集团旗下公司有互联网广告投放业务等项目...
39分钟前
宣亚国际4月12日在互动平台回答投资者提问时表示,公司与阿里巴巴集团旗下公司有互联网广告投放业务等项目合作。 原标题:宣亚国际:公司与阿里巴巴集团旗下公司有互联网广告投放业务等项...
东方财富网

正则表达式修饰符含义

修饰符	描述
re.I	使匹配对大小写不敏感
re.L	做本地化识别（locale-aware）匹配
re.M	多行匹配，影响 ^ 和 $
re.S	使 . 匹配包括换行在内的所有字符
re.U	根据Unicode字符集解析字符。这个标志影响 \w, \W, \b, \B.
re.X	该标志通过给予你更灵活的格式以便你将正则表达式写得更易于理解。

输出搜索到的全部链接

```
import requests
import re
import time
headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36'}
def baidu(company):
    url = 'http://www.baidu.com/s?tn=news&rtt=4&wd=' + company
    res = requests.get(url, headers=headers).text
    p_href = '<h3 class="news-title_1YtI1 "><a href="(.*?)"'
    href = re.findall(p_href, res, re.S)
    print(href)
    …
baidu('阿里巴巴')
```

rtt=4按时间排序
rtt=1按焦点排序

输出搜索到的标题、日期、来源

```
    …
    p_title = '<h3 class="news-title_1YtI1 ">.*?>(.*?)</a>'
    title = re.findall(p_title, res, re.S)
    print(title)
    p_date = '<span class="c-color-gray2 c-font-normal c-gap-right-\
xsmall" .*?>(.*?)</span>'
    date = re.findall(p_date, res)
    print(date)
    p_source = '<span class="c-color-gray" .*?>(.*?)</span>'
    source = re.findall(p_source, res)
    print(source)
```

部分搜索结果展示

['<!--s-text-->宣亚国际:公司与阿里巴巴集团旗下公司有互联网广告投放业务等项目...<!--/s-text-->', '<!--s-text-->张勇:阿里巴巴所有产品未来将接入大模型全面改造<!--/s-text-->', '<!--s-text-->读特专稿|放权动真格?解读阿里巴巴“分家式”组织变革<!--/s-text-->', '<!--s-text-->小摩:阿里巴巴股价上行空间具吸引力 上季度经调整EBITA或升48%至...<!--/s-text-->', '<!--s-text-->概念动态|特发服务新增“阿里巴巴概念”<!--/s-text-->', '<!--s-text-->中金:维持阿里巴巴“跑赢行业”评级,目标价137港元<!--/s-text-->', '<!--s-text-->宣亚国际:公司与阿里巴巴集团旗下公司有互联网广告投放业务等项目...<!--/s-text-->', '<!--s-text-->中金:维持阿里巴巴跑赢行业评级 目标价137港元<!--/s-text-->', '<!--s-text-->国信证券维持阿里巴巴买入评级<!--/s-text-->', '<!--s-text-->阿里巴巴所有产品未来将接入「通义千问」,将推企业专属大模型|最...<!--/s-text-->']
['1小时前', '12小时前', '3小时前', '4小时前', '1小时前', '10小时前', '5小时前', '5小时前']

数据清洗常见方法

- 用strip()函数删除空格及换行符等非相关符号

```
>>> res='  华能信托本年实现利润32.05亿元  '
>>> res=res.strip()
>>> res
'华能信托本年实现利润32.05亿元'
```

数据清洗常见方法

- 用split()函数截取需要的内容

```
>>> date='2019-01-20 10:10:10'
>>> date=date.split(' ')[0]
>>> date     '2019-01-20'
```

数据清洗常见方法

- 用sub()函数进行内容替换

```
>>> import re
>>> title='阿里<em>巴巴</em>人工智能再发力'
>>> title=re.sub('<.*?>', '',  title)
>>> title     '阿里巴巴人工智能再发力'
```

短语标签，用来呈现为被强调的文本

搜索结果清洗及输出

```
for i in range(len(date)):
    title[i] = title[i].strip()
    title[i] = re.sub('<.*?>', '', title[i])
    if ('小时' in date[i]) or ('分钟' in date[i]):
        date[i] = time. strftime("%Y-%m-%d")
    else:
        date[i] = date[i]
    print(str(i + 1) + '.' + title[i] + '(' + date[i] + '-' + source[i] + ')')
    print(href[i])
```

爬取多公司多页数据核心代码

```
# 爬取多个公司的多页, 可以给函数传入两个参数
def baidu(company, page):
    num = (page-1) * 10  # 参数规律是（页数-1）*10
    url = 'http://www.baidu.com/s?tn=news&rtt=4&wd='+company+'&pn='+str(num)
    res = requests.get(url, headers=headers).text
    print(res)
```

爬取多公司多页数据核心代码

```
companys = ['阿里巴巴', '万科集团', '百度集团', '腾讯', '京东']
for company in companys:
    for i in range(10):  # 这里一共爬取了10页
        baidu(company, i+1)  # i+1表示第几页
        print(company + '第' + str(i+1) + '页爬取成功')
        time.sleep(3)
```

Python支持的数据库

- SQLite
- MySQL
- MongoDB
- Redis
- Microsoft SQL Server 2000
- ….

常用数据库—

SQLite：是一个开源的关系型数据库，具有零配置、自我包含、便于传输等优点。它将整个数据库的表、索引、数据都存储在一个单一的.db文件中，不需要网络配置和管理，没有用户帐户和密码，访问依赖于文件所在操作系统。

SQLite数据库连接

- 和数据库建立连接
- 执行sql语句，接收返回值
- 关闭数据库连接

常用SQL语句

- 创建一个新的数据表

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''create table student(code char(4) not null,
        name char(10),age int, primary key("code"))'''
conn.execute(SQL)
conn.commit()
conn.close()
```

常用SQL语句

- 往一个表中插入数据

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''insert into student (code, name, age)
                    values('9001', '张芳' , 20)'''
conn.execute(SQL)
conn.commit()
conn.close()
```

常用SQL语句

- 更新数据表中的数据

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''update student set name="张三", age=25
                                    where code='9001' '''
conn.execute(SQL)
conn.commit()
conn.close()
```

常用SQL语句

- 从一个表中删除数据

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''delete from student where code='9001' '''
conn.execute(SQL)
conn.commit()
conn.close()
```

常用SQL语句

- 删除表

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''drop table student'''
conn.execute(SQL)
conn.commit()
conn.close()
```

人民网爬虫存入数据库中

爬取人民网链接和标题

```
import requests
import re
url="http://www.people.com.cn"
html=requests. get(url)
html. encoding=html. apparent_encoding
data=html. text
reg=r'<a href="(.*?)" target="_blank">(.*?)</a>'
urls=re.findall(reg, data)
print(urls)
```

爬虫结果存入数据库

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL='''create table information(name char(30) not null,
        link char(20), primary key("name"))'''
conn.execute(SQL)
conn.commit()
```

爬虫结果存入数据库

```
for item in urls:
    SQL="""insert into information(name,link)
                values('%s', '%s')""" %(item[1],item[0])
    conn.execute(SQL)
    conn.commit()
conn.close()
```

从数据库中查询部分记录

```
import sqlite3
conn=sqlite3.connect("school.db")
SQL=""select * from information where name like "人民%" ""
aList=list(conn.execute(SQL))
conn.commit()
for line in aList:
    print(line)
conn.close()
```

常用数据库二

MySQL：是一个关系型数据库管理系统，是最流行的关系型数据库管理系统之一，在WEB 应用方面，MySQL是最好的RDBMS应用软件。

谢　谢