浅谈利用 paython 语言完成电商网站商品信息的爬取代码设计

梁思远 成都市铁路中学

摘要：随着大数据时代到来，爬虫的需求呈爆炸式增长，以淘宝、京东为代表电商网站拥有大量的商品信息和价格，但是对于同类商品的比价问题是一个比较麻烦的问题，我们以京东为研究对象，利用 Python 语言实现指定网页地址的商品的信息和价格的爬取和展示。

关键词：网上购物　购物网站　数据分析　Python 爬虫

1 引言

现在网上购物已成为人们生活的一部分，各类购物网站中蕴含着巨大商品信息和商品价格。但是，因为在购物网站中存在大量的商家，同一个商品的报价存在着差异，对于购买客户来说价格比较是一个比较枯燥烦琐的问题，因此，许多技术成熟的科研团队自行开发爬虫系统来获取商品信息和价格供购买客户进行价格比较，我们在这儿只是探讨一下爬虫技术的简单实现。

Python 作为一个语法简洁的程序设计语言，对于爬虫开 发上有得天独厚的优势，在模拟浏览器行为登入网站时，Python 相比于 Java，C#，C++ 等拥有更简洁抓取接口，当模拟 session/cookie 的存储和设置时，Python 提供诸多优秀的第 三方包譬如 Requests。在进行网页抓取后的处理工作时，Python 提供的 BeautifulSoup 库能用极简短的代码完成过滤 html 标签，提取文本的工作。

2 利用 Python 语言实现的爬虫代码

```
# -*- coding: utf-8 -*-
from urllib.request import urlopen
from bs4 import  BeautifulSoup
import re
import json
class JdSpider(object):
    def __init__(self, url):
        self.url = url
    def get_html(self):
        doc = urlopen(url = self.url).read()
        con = BeautifulSoup(doc, "lxml")
        return con
    def get_id(self):
        id = self.url.split('/')[3].split('.')[0]
        return id
    def get_name(self):
        product_info = self.get_html()
        name = BeautifulSoup(str(product_info.find_all("div", class_="sku-name")), "lxml").get_text()
        name_re = re.compile(r"\w.*\w")
        name = re.findall(name_re, name)[0]
        return name
    def get_price(self):
        info_url = 'http://p.3.cn/prices/mgets?skuIds=J_{0}&type=1'.format(self.get_id())
        price_json = json.loads(urlopen(info_url).read().decode('utf8'))[0]
        if price_json['p']:
            price = price_json['p']
        return price
    def get_goodrate(self):
        comment_url = 'https://club.jd.com/comment/productPageComments.action?' \
                      '&productId={0}&score=0&sortType=5&page=0&pageSize=10'.format(self.get_id())
        goodrate_json = json.loads(urlopen(comment_url).read().decode('gbk'))["productCommentSummary"]
        goodrate = goodrate_json["goodRateShow"]
        return goodrate
    def printout(self):
        print(" 商品 id: {0}".format(self.get_id()))
        print(" 商品名称: {0}".format(self.get_name()))
        print(" 商品价格: {0}".format(self.get_price()))
if __name__ == "__main__":
    url = input(" 请输入京东商品网址: ")
    jd = JdSpider(url = url)
    jd.printout()
```

参考文献

[1] 林晓丽，胡可可，胡青．基于 Python 的微博用户关系挖掘 研究 [J]. 情报杂志 ,2014,33(6):144-148.

[2] 陈政伊 袁云静 贺月锦 武瑞轩基于 Python 的微博爬虫系统研究 [J]. 大众科技 ,2017 年 8 月第 19 卷 216 期 :8-11.

[3] 周中华，张惠然，谢 江．基于 Python 的新浪微博数据爬虫 [J]. 计算机应用，2014，34(11) : 3131 - 3134

[4]Shih-Yu Huang,Yeuan-Kuen Lee,Graeme Bell,Zhan-he Ou,et al. An efficient segmentation algorithm for CAPTCHAs with line cluttering and character warping[J].Multimedia Tools and Applications,2009,48(2):267-289.