基于 Python 的图片爬虫程序设计

文/云洋

摘要

互联网中包含大量有价值的数据，网络爬虫通过既定规则可以自动地抓取互联网数据并下载至本地存储。研究网络爬虫的工作原理和基于 Python 网络信息爬取技术模块功能，基于 Requests-BeautifulSoup 技术构建图片爬虫实现网页图片爬取，详细阐述了百度贴吧美图吧图片爬虫程序的采集、解析、爬取和存储过程，实验结果证明基于 Python 的 Requests-BeautifulSoup 技术可快速构建图片爬虫程序实现对网页图片数据的自动解析和爬取，程序简单有效并且数据采集速度快。

【关键词】Python 语言 网络爬虫 Request-BeautifulSoup 图片爬取

网络爬虫 (Web Crawler) 又称网络蜘蛛 (Web Spider) 是一个能够根据既定规则自动提取网页信息的程序，它模仿浏览器发出 HTTP 请求访问网络资源，自动获取用户需要的网页数据。已有一些定向网站的网络爬虫 , 如 QQ 空间爬虫一天可抓取 400 万条日志、说说、个人信息等数据；知乎爬虫爬取各种话题下的优质答案；淘宝商品比价定向爬虫爬取商品、评论及销售数据。

Python 是一种面向对象、解释型、带有动态语义的高级程序设计语言，其语法简洁清晰，并具有丰富和强大的类库，Python 语言支持覆盖信息技术各领域的 10 万个函数库，依靠开源快速开发，形成了全球最大的编程社区。2017 年 7 月 IEEE 发布的编程语言排行榜中 Python 高居首位，基于 Python 的应用也在计算机各领域大放异彩。Python 包含优秀的网络爬虫框架和解析技术 , Python 语言简单易用且提供了与爬虫相关的 urllib、requests、BeautifulSoup、Scrapy 等模块。Urllib 模块提供了从万维网中获取数据的高层接口，Requests 模拟浏览器自动发送 HTTP/HTTPS 请求并从互联网获取数据，BeautifulSoup 解析 HTML/XML 页面获取用户需要的数据。本文基于 Python 的 Requests-BeautifulSoup 技术构建图片爬虫程序实现对百度贴吧美图图片的快速爬取，并将这些图片保存在本地，方便用户离线浏览和进一步使用。

1 网络爬虫工作原理与Python爬虫技术模块功能

网络爬虫是按照一定规则能自动抓取互联网数据的程序或者脚本。网络爬虫通过网络请求从 Web 网站首页或指定页面开始解析网页获取所需内容 , 并通过网页中的链接地址不断进入到下一个网页 , 直到遍历完这个网站所有的网页或满足爬虫设定的停止条件为止。Python 语言第三方网络请求库 Requests 模拟浏览器自动发送 HTTP/HTTPS 请求并从互联网获取数据。BeautifulSoup 解析获取的 HTML/XML 页面为用户抓取需要的数据，Beautiful Soup 自动将输入文档转换为 Unicode 编码，将输出文档转换为 utf-8 编码，从而节省编程时间。

1.1 网络爬虫的工作原理

网络爬虫爬取页面就是模拟使用浏览器获取页面信息的过程，其爬取流程一般包含如下 4 个步骤：

（1）模拟浏览器发起请求：通过目标 URL 向服务器发起 request 请求，请求头 header 一般包含请求类型、cookie 信息以及浏览器类型信息等；

（2）获取服务器页面响应：在服务器正常响应的情况下，用户会收到所请求网页的 response，一般包含 HTML、Json 字符串或其他二进制格式数据（如视频，图片）等；

（3）获取页面内容解析：用相应的解析器或转换方法处理获取的网页内容，如用网页解析器解析 HTML 代码，如果是二进制数据（如视频、图片），则保存到文件进一步待处理；

（4）存储数据：网页解析获取的数据可以用 CSV、Json、text、图片等文件存储，也可以 sqlite、MySQL 或者 MongoDB 等数据库存储。

1.2 Python第三方库Requests模块

Requests 是用 Python 语言编写 , 使用 Apache2 Licensed 许可证的 HTTP 库。Python 标准库中自带的 urllib2 模块和 httplib 模块提供了所需要的大多数 HTTP 功能，Requests 使用 urllib3 模块，支持 HTTP 连接保持和连接池，支持使用 cookie 保持会话，支持文件上传，支持自动确定响应内容的编码，支持国际化的 URL 和 POST 数据自动编码。

通过 pip 命令（$pip install requests）安装 Requests 模块。urllib 提供了一系列用于操作 URL 的功能，urllib 的 request 模块可以方便地访问抓取 URL(统一资源定位符) 内容，urllib.request 模块中常用的函数方法如表 1 所示。使用 requests 方法后，会返回一个 response 对象存储服务器响应的内容，如 r.status_code（响应状态码）、r.text（字符串方式的响应体，会自动根据响应头部的字符编码进行解码）、r.json（Requests 中内置的 JSON 解码器）、r.content（字节方式的响应体，会自动为你解码 gzip 和 deflate 压缩）等。

1.3 Python第三方库Beautiful Soup模块

Beautiful Soup 是用 Python 写的一个 HTML/XML 的解析器，它可以处理不规范标记并生成分析树 (parse tree)，同时提供了简单的 python 函数处理导航（navigating）、搜索并修改分析树。

通过 pip 命令安装（$ pip install beautifulsoup4）Beautiful Soup 模块。BeautifulSoup 将 HTML 文档转换成一个树形结构 , 每个节点都是 Python 的对象 , 所有对象可归纳为 4 种，如表 2 所示。

2 帖吧图片爬虫程序设计

百度贴吧是全球最大的中文社区。贴吧是一种基于关键词的主题交流社区，贴吧结合搜索引擎建立一个在线的交流平台，让那些对同一个话题感兴趣的人们聚集在一起，方便地展开交流和互相帮助。设计爬虫程序爬取百度帖吧（http://tieba.baidu.com）内的美图吧图片，运行爬虫程序时提示用户输入想要爬取网站的 url，爬虫程序修改请求头信息，模拟浏览器对贴吧内的帖子依次使用 get 请求，进入帖子后根据规则找到所有图片标签，获取帖子内的图片资源 url，并将其依次下载到本地存储 , 所有帖子爬取完成后按 enter 退出，运行中途也可以使用 ctrl+c 强制退出程序。

基于 Python 的 Requests-BeautifulSoup 技术构建图片爬虫程序，使用 requests 模拟浏览器请求网页，用 random 生成随机数选取模拟的浏览器，用 BeautifulSoup 支持的 Python 内置标准 HTML 解析库解析请求网页返回的数据，使用 urllib.request.urlretrieve( ) 下载图片和各种网络请求。

2.1 爬虫准备

开发图片爬虫程序使用 Python3.6 版本，主要用到了 urllib 的 requests 模块、BeautifulSoup 模块和 random 模块，模块是包含变量、函数或类的定义程序文件，使用模块前通过 import 导入这些模块。定义了两个全局变量 null 和 true 并初始化，以避免当访问网址 url 中出现 null 和 true 字样时，Python 会将 null 和 true 当成变量未初始化而报错。

```
import urllib.request
from bs4 import BeautifulSoup
import random
global null  # 设置了两个全局变量 null 和 true 并初始化
null=''
global true
```

表 1：urllib.request 模块中的主要方法

方法	功能
urllib.request.urlopen(url[,data[,proxies]])	打开一个 url，返回一个文件对象，可以对文件对象进行 read()、readline()、readlines()、fileno()、close()、info()、geturl() 等文件操作。
urllib.request.urlretrieve(url,filename,mine_hdrs)	将 url 定位到的 html 文件下载到你本地的硬盘中。urlretrieve() 返回一个二元组 (filename,mine_hdrs)
urllib.request.Request(url)	用 Request 构建一个增加 headers 信息的完整的 url 请求。

表 2：Beautiful Soup 模块中的对象

对象	功能
Tag	Tag 对象定义了多种函数和属性，如 name 和 attribute。每个对象有自己的名字，通过逗号操作符可引用 name 属性，直接赋值字符串可修改 name 属性。attributes 属性 : tag <b class="test"> This is the test begin..."> 有一个 “class" 的属性 , 值为 "test"。
NavigableString	NavigableString 类来装 tag 中的字符串 , 字符串是用 unicode 进行编码的 , 可以通过 string 的函数 replace_with 进行字符串的替换。
BeautifulSoup	
	BeautifulSoup 属性表示文档所有的内容 , 一般可当作 Tag 对象用 , 包含一个值为 “[document]” 的特殊属性 name，如 BeautifulSoup(html,’ html.parser’) 解析 html 文档。
Comment	Comment 属性是一个特殊类型的 NavigableString 对象 , 当出现在 HTML 文档中时 , Comment 对象会使用特殊的格式输出。

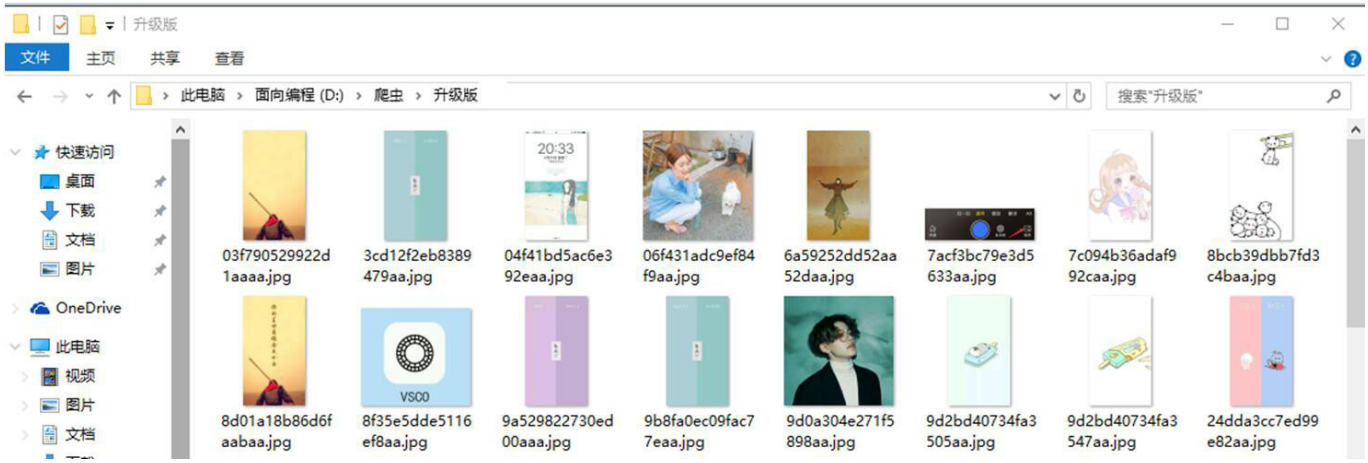

图 1：抓取的百度贴吧美图图片

true="

2.2 定义图片抓取函数

定义图片抓取函数 get_images()，使用 BeautifulSoup 解析获取网页，找到所有图片标签，从每个图片标签中下载图片并重命名到本地保存，其代码如下所示。

```
def get_images(info):        # 定义图片抓取函数
    soup=BeautifulSoup(info,'html.parser')    # 创建 beautifulsoup 对象 soup
    all_img=soup.find_all('img',class_='BDE_Image')    # 找到所有图片标签
    for img in all_img:        # 从每一个图片标签中下载图片并重命名
        image_name='%s.jpg'%img['src']
        image_name=image_name.replace('.jpg','a')
        image_name=image_name+'.jpg'
        image_name=image_name[-22:]
        urllib.request.urlretrieve(img['src'],image_name)        # 下载图片保存到本地文件 image_name
        print(' 成功抓取到图片 ',img['src'])
    print(' 抓取完成 !')
```

2.3 模拟浏览器访问网站

爬虫程序模拟浏览器发送 HTTP/HTTPS 请求并从互联网获取数据。用户代理 User Agent 是 Http 协议的一部分，是请求头信息的一部分。random 是 Python 标准库的模块，用户可以直接调用 random 的方法 random.randint(a,b) 生成一个指定范围内的整数，其中参数 a 是下限，参数 b 是上限，下限必须小于上限。可用该函数生成随机整数用于选取模拟的浏览器。访问网站时通过用户代理向服务器提供用户使用的浏览器版本及类型，通过改写 User-Agent 将 Python 爬虫模拟成浏览器。如下代码模拟不同浏览器型号并进入百度帖吧首页。

```
Agent=['Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_8; en-us) AppleWebKit/534.50 (KHTML, like Gecko) Version/5.1' ,
        'Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.0; Trident/4.0)',
        'Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Maxthon 2.0)',
        'Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; The World)',
        'Mozilla/5.0 (Windows NT 6.1; rv:2.0.1) Gecko/20100101 Firefox/4.0.1']
yemian=input(' 输入要抓取的贴吧地址 :')
req = urllib.request.Request(yemian)    # 用 Request 构建添加 headers 信息的完整 URL 请求
req.add_header('Host','tieba.baidu.com')    # 添加请求头信息 Host、Referer、User-Agent
req.add_header('Referer','http://tieba.baidu.com/')
req.add_header('User-Agent',Agent[random.randint(0,4)])  # 用 random 生成随机数选取模拟的浏览器
```

2.4 进入帖子爬取图片核心代码

爬虫主体程序核心代码如下，通过 urllib

<< 下转 244 页

人工智能两大趋势：自主能力和生物特性

文/陶阳明

摘要

人工智能（Artificial Intelligence，AI）是目前发展非常迅速的一门学科，并且已经开始应用于社会和生活，如人脸识别、语言翻译和聊天机器人等。然而人工智能的潜力决不仅仅于此，文章首先创造性的对人工智能进行了新的分类；然后通过对实现人工智能算法的概述，总结出人工智能的一般发展规律；接着对目前人工智能的缺陷进行了分析，并给出了突破现有人工智能局限的关键所在；最后阐述了人工智能未来发展的两大趋势：自主能力和生物特性。

【关键词】人工智能 自主能力 生物特性 机器学习 遗传算法 发展趋势

目前所谓的“人工智能”仍然非常初级。世界公认的人工智能、机器学习界的泰斗Hinton也对自己提出的反向传播理念深感质疑；很多科技界的其他知名人士也开始反思目前流行的深度学习很多特点并不符合实际的大脑机制原理，更加无法做到像人类一样轻取复杂问题。

由此，笔者也引发了一些思考，主要表现为这些“人工智能”并没有自主性和生物特性。下面依次阐述文章的观点。

1 人工智能分类

目前来说，科学界对人工智能多是从应用领域方面进行分类，文章创造性的从人工智能发展趋势及技术层面等综合因素对人工智能进行分类：狭义的人工智能和广义的人工智能。

1.1 狭义的人工智能定义

狭义人工智能是以计算机为载体，用一组程序或者指令把所有预测情况表达出来，并且通过判断在相应的条件下给出最佳选择。狭义的人工智能绝大部分是被动性的，因为所有算法都在可预测当中，如果超出预测就直接中断执行。

狭义的人工智能具有机械特性，高效率性和高准确率性，然而这种智能也失去了灵活性和自主性。

1.2 广义的人工智能定义

广义的人工智能是以计算机或其它物体（比如具备生物特性的传感器）作为载体，能够自主根据所处不同环境而自发编写程序或指令，并且能够产生一个合适的算法并能自主执行。

广义的人工智能包含狭义的人工智能所有的优秀特性，并且具有可控的自主特性如自主学习能力、自主编程能力等。广义的人工智能应该包含生物特性，比如生物最基本新陈代谢特性。

1.3 以上两类人工智能的对比

（1）人工智能的从狭义的人工智能过渡到广义的人工智能是必然的趋势。

（2）广义的人工智能包含狭义的人工智能的所有优秀特性，如高效率性、高准确性等等。

（3）广义的人工智能具备生物特性和狭义人工智能具备机械特性是两者之间的根本区别。

2 人工智能算法概述

实现人工智能的方法目前主要分为两大类：工程学方法和模拟法。工程学方法仅从逻辑层面去设计并编程算法去实现人工智能，不会考虑是否与人或动物机体所用的方法相同或者相似；而模拟法会更多的通过模拟人或动物机体所用的方法来设计算法并最终实现人工智能。

<< 上接 242 页

的 requests 和 urlopen() 方法模拟浏览器访问网站获取网页数据，用 BeautifulSoup 的 find_all() 解析获取的网页数据，进入帖子内抓取带有图片标签的图片文件并下载保存在本地磁盘。运行图片爬虫程序后爬取下载的帖吧图片存储如图 1 所示。

```
    html = urllib.request.urlopen(req)    # 打开一个 url 请求返回一个文件对象 html
    string = html.read()
    soup=BeautifulSoup(string,'html.parser')  # 用 Python 标准库解析请求网页返回的数据
    all_img = soup.find_all('li',class_='j_thread_list')  # 返回文档中符合条件的所有标签 tag
    for img in all_img:        # 从图片连接中下载图片
        a = eval(img['data-field'])
        b = a['id']
        url2='http://tieba.baidu.com/p/%s'%b
        req2 = urllib.request.Request(url2)
        req2.add_header('Host','tieba.baidu.com')
        req2.add_header('Referer','http://tieba.baidu.com/')
        req2.add_header('User-Agent',Agent[random.randint(0,4)])
        html2=urllib.request.urlopen(req2)
        info = html2.read()
        get_images(info)
    print(' 全部抓取完成 .')
    input()  # 程序运行结束后等待用户输入回车键再退出
```

3 结束语

本文研究了网络爬虫的工作原理和Python构建爬虫的相关技术模块，讨论了Python构建爬虫的模块urllib、BeautifulSoup和random的功能用法。以百度贴吧图片爬虫构建为例，从爬虫准备、模拟浏览器登陆网站、定义图片爬取函数、进入贴吧解析网页、爬取图片存储等方面详细阐述了采用Python的Requests-BeautifulSoup技术构建图片爬虫程序抓取百度贴吧美图吧图片的过程。实验结果证明基于Python的Requests-BeautifulSoup技术可快速有效地构建图片爬虫程序实现对网页图片数据的自动解析和爬取。

参考文献

[1] 郭丽蓉．基于Python的网络爬虫程序设计[J]．电子技术与软件工程，2017(23)：248-249.

[2] 贾棋然．基于Python专用型网络爬虫的设计及实现[J]．电脑知识与技术，2017(12)：47-49.

[3] 刘艳平，俞海英，戎沁．Python模拟登录网站并抓取网页的方法[J]．微型电脑应用，2015,31(01)：58-60.

[4] 涂辉，王峰，商庆伟．Python3编程实现网络爬虫[J]．电脑编程技巧与维护，2017(23)：21-22.

[5] 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.

作者简介

云洋，女，山东省青岛市人。单位为山东省青岛第五十八中学。研究兴趣为Python程序设计、Web应用。

作者单位

山东省青岛第五十八中学　山东省青岛市 266101