爬虫05-技术选型与爬虫能做什么

大家好，今天主要想和大家讨论一下编写爬虫的技术选型，以及爬虫到底能干啥。

首先，我们来看下爬虫编写的技术选型，粗略地分为两类进行对比，分别是Scrapy和requests + Beautifulsoup。

在讨论这两种方式的区别之前，我们先来讨论下库与模块的区别：

> **一个库可能由多个模块组成，而一个模块就是一个py文件。**

接下来，我们再来看下Scrapy和requests + Beautifulsoup之间的区别在哪里：

> 1. requests和Beautifulsoup都是库，而Scrapy是框架。
>
> 两者根本不是一个层级的东西，每个框架都是集成了很多库的，在这些库的基础上进行了多次封装，做了很多处理。
>
> 2. Scrapy框架中可以加入requests和Beautifulsoup
>
> 在Scrapy框架中，我们仍然可以使用requests和Beautifulsoup，但是极其不建议在Scrapy框架中使用Beautifulsoup，原因后面会讲到。
>
> 3. Scrapy是基于Twisted开发的，是一个异步的框架，性能是它最大的优势
>
> 这里简单提一下**同步和异步的区别**：
> 举个简单的例子来说，我们现在向https://www.baidu.com发起请求，书写了以下代码：

```
import requests
res = requests.get('https://www.baidu.com')
print(res.text)
```

> 上述代码就是**同步**的代码，当代码执行到第2行的时候，其实是堵在这里的，因为从发起请求道服务器返回响应的过程是需要一定的时间的，只是这个时间很短，我们感觉不

到，但却是实实在在存在的，只有服务器返回响应之后，代码才会继续往下执行，这就是同步，就相当于我们在等待服务器返回响应的那段时间是浪费了的。

而**异步**是怎么样的，异步就是很好地利用了服务器返回响应的那段时间，利用那段时间驱动程序去执行另外的代码，等服务器返回响应之后又回过来，接着继续往下执行。

这就是同步和异步的区别，虽然异步的代码的性能很高，但是编写起来就会相应的很复杂。

4. Scrapy方便扩展，提供了很多内置的功能

Scrapy本身提供了很多扩展供我们使用，另外还支持我们自定义自己的扩展，去实现我们自己想实现的功能，关于Scrapy我们课程的最后会给予讲解。

5. Scrapy内置的css 和 xpath selector选择器非常方便，而Beautifulsoup的最大缺点就是慢

因为xpath是用C语言写的，而Beautifulsoup是用纯python写的，所以性能上会存在巨大的差异，一两条可能感觉不大， 当要爬取的数据量过大时，这种差距就会越明显。

讲到这里，肯定就会有同学有疑问了，既然Scrapy这么优秀，我干嘛还要学习requests和Beautifulsoup呢，既然存在而且还没有被淘汰，自然有他的道理。

我们再来看下这两种方式的使用场景：

requests + Beautifulsoup：适合于爬取少量数据，临时用用，不适合爬取大量数据。

就比如说，我最近想看看书，但是不知道看什么书，就想去爬取豆瓣的高分书籍，就爬取1000条数据，从中选取基本评分高的来看看。这样的爬取数据量较小，而用requests + Beautifulsoup又很简单，这是一种最方便的方式。

Scrapy：适合于爬取大量数据，更适合商用。

因为Scrapy的配置还是要麻烦一点的，要先创建项目，然后又是各种配置，有点麻烦，用它去爬取少量数据就有点不划算，但是用来爬取大量数据，这个优势就很明显了。

讲完了爬虫编写的技术选型，接下来，我们再来看看爬虫能够具体做些什么：

1. 搜索引擎——百度、google、垂直领域搜索引擎
   搜索引擎就是不停地在网络中爬取所有的数据，然后对这些数据进行分析，最后给用户提供一个搜索接口。

2. 推荐引擎——今日头条
   推荐引擎也是不停地在网络中爬取数据，但它是有目的地在爬取，它是事先知道要爬取哪些数据的，然后爬取到数据后，通过一定的算法，将用户感兴趣的内容推送给用户。

3. 机器学习的数据样本
   机器学习、深度学习等等人工智能领域都需要大量的数据来训练写好的算法，使得算法的准确率越来越高，而这些数据从哪里来呢，就可以用爬虫来获取。

4. 数据分析（例如金融数据分析）、舆情分析等
   数据分析同样需要样本，而这些数据也可以用爬虫去网络上爬取。