

## 大数据时代下基于 Python 的网络信息爬取技术

文/刘顺程 岳思颖

## 摘要

在大数据时代下, 各行各业都需要大量数据的支持, 如果所有数据都经过人工搜索、分析、提炼, 则会大大增加工作难度。基于 Python 的网络信息爬取技术可以自动完成网络数据的收集、解析、格式化存储, 从而提升工作效率。本文以网络信息爬取技术为研究重点, 分别介绍网络爬虫的基础架构与运行流程, 以及基于 Python 的网络爬取技术实现。

【关键词】网络爬虫 Python 大数据

随着“互联网+”概念不断普及, 网络信息量呈突发性暴增, 导致传统搜索引擎普遍存在搜索结果附带大量无关信息的问题, 加大了收集专用数据的难度。于是, 网络信息爬取技术(后简称网络爬虫技术)应运而生。

使用网络爬虫技术可以自动完成网络数据的挖掘与分析工作。现今的大数据时代, 在许多新兴产业中, 通过爬虫爬取下来的信息可以作为数据仓库多维展现的数据源, 也可作为数据挖掘的来源。所以网络爬虫技术是目前大数据时代下的重要基础应用。

## 1 网络爬虫的架构与流程

## 1.1 网络爬虫架构

网络爬虫架构主要有以下三个基础部分: 网络爬虫调度端; 网络爬虫主程序; 价值数据。

爬虫调度端能监控整个爬虫程序的运行情况; 其中爬虫主程序包括:

- (1) URL 管理器, 管理将要爬取的 URL 以及已经爬取过的 URL;
- (2) 网页下载器, 根据待爬 URL 将指定的网页下载下来, 并存储为字符串数据;
- (3) 网页解析器将网页字符串数据进行数据抽取, 一方面提取出价值数据, 另一方面提取出新的关联 URL 传递给 URL 管理器。三个部分循环进行, 只要 URL 管理器还有待爬取的 URL, 就会循环进行下去, 最终提取出所有价值数据。

## 1.2 网络爬虫流程

基于以上架构的网络爬取流程, 首先是调度端询问 URL 管理器, 是否有待爬取的 URL, 如过返回是, 调度端会取得第一个待爬取 URL 地址, 并将其传送给网页下载器进行网页下载, 调度端接收到网页下载内容后立即将其传送给网页解析器, 解析后返回价值数据和新的 URL 列表给调度端, 一方面将价值数据传递给应用进行收集, 另一方面将新 URL 列表增加到 URL 管理器中。只要 URL 管理器有待爬取 URL, 以上过程会循环进行。最终调度端会将应用中的价值数据进行处理并输出为需要的格式。

## 2 基于Python的爬虫模块技术实现

URL 管理器能管理待爬 URL 列表和已爬 URL 列表, 能够有效防止重复抓取和循环抓取, 在 Python 中的实现方式有三种: 通过内存, 将 URL 列表存储在 Python 内存中, 使用两个 set() 数据结构分别存储待爬取与已爬取列表, Python 中的 set() 能自动去除集合中重复的元素, 从而有效防止重复抓取。第二种是将 URL 存储在关系数据库中, 比如 MySQL, 可以建立一张名为 url\_list 的表, 字段为 (url, is\_crawled) 分别表示 URL 地址和标识该 URL 是否已被爬取, 这样就使用一张表将待爬取和已爬取都进行了存储。第三, 在大型互联网公司中常常使用缓存数据库来搭建 URL 管理器, 是由于其高效率处理大量数据的能力, 例如 redis, 同样支持 set 数据结构, 也就可以将待爬取与已爬取 URL 存储在两个 set 集合中。

网页下载器能将指定 URL 的网页下载到本地存储成本地文件或字符串格式, 以便进行后续步骤的数据分析, 故网页下载器是爬取程序的核心模块。在 Python 中我们可以使用 urllib2 网页下载器。这是一个 Python 官方基础模块, 它提供了网页下载、提交用户数据、登录 cookie 处理、代理访问处理等强大功能; 我们还能使用功能更为强大的 requests, 它是一个 Python 的第三方插件, 同样支持网页下载、登录、文件上传等功能。当我们请求的 URL 网页需要用户登录或验证登录时, 便可使用网页下载器提供的特殊处理器, 例如在登录操作中, 通常需要操作 cookie 才能成功登陆, 于是需要使用特殊的处理器如

HTTPCookieProcessor, 将爬虫程序伪装成用户使用浏览器正在登录该网站, 随后即可顺利获取网页内容。

网页解析器是一个能从网页字符串文件中解析出价值数据的处理器, 对于的专业爬虫来说就是提取出待爬取 URL 列表和提取出价值数据。Python 中有许多网页解析器, 其中使用最为广泛的是 BeautifulSoup 这个第三方插件, 它首先进行网页字符串的结构化解析, 利用 HTML 与 DOM 的映射关系, 将 HTML 文档转化为 DOM 树, 对其进行基于结构的过滤和基于语义的剪枝操作, 使用树形结构能很精准定位到某个节点、属性、文本内容; 接下来即可使用 find\_all 或 find 方法查询相应节点, 访问节点名称、属性、文字; 从而提取出价值信息进行分析。

## 3 结束语

进入大数据时代, 众多行业都急需价值数据。网络爬取技术能帮助客户有效地收集网络上的相关价值信息, 大大降低人力搜索的工作量。同时基于 Python 的网络爬取技术不仅简单易学, 而且拥有强大的爬虫框架作为优势, 使得开发者能更快地开发出拥有指定功能的爬虫程序。

## 参考文献

- [1] 罗刚. 自己动手写网络爬虫 [M]. 北京: 清华大学出版社, 2010.
- [2] 王琦, 唐世渭, 杨冬青, 王腾蛟. 基于 DOM 的网页主题信息自动提取 [J]. 计算机研究与发展, 2004 (10): 1786-1792.

## 作者简介

刘顺程 (1997-), 男, 大学本科在读。就读于重庆邮电大学软件工程学院。主要研究方向为网络安全与大数据。

岳思颖 (1997-), 女, 大学本科在读。就读于重庆邮电大学软件工程学院。主要研究方向为网络技术。

## 作者单位

重庆邮电大学软件工程学院 重庆市 400065