网络爬虫

- 什么是爬虫
  - 网络爬虫又称网络蜘蛛、网络蚂蚁、网络机器人等，可以自动化浏览网络中的信息，当浏览信息的时候需要按照我们规定的规则进行，这些规则称之为网络爬虫算法。使用Python可以很方便地写出爬虫程序，进行互联网的自动化检索。
- 为什么学习爬虫
  - 私人订制一个收索引擎，并且可以对收索引擎的数采集工作原理进行更深层次的理解
  - 获取更多的数据源，并且这些数据源可以按我们的目的进行采集，去掉很多无关数据
  - 更好地进行SEO（收索引擎优化）
- 网络爬虫的组成
  - 控制节点叫做爬虫中央控制器，主要负责根据URL地址分配线程，并调用爬虫节点进行具体的爬行
  - 爬虫节点按照相关的算法，对网页进行具体的爬行，主要包括下载网页以及对网页的文本处理，爬行后会将对应的爬行结果保存到对应的资源库。
  - 资源库构成存储爬虫去到的相应数据，一般为数据库
- 爬虫设计思路
  - 首先确定需要爬取的网页URL地址
  - 通过HTTP协议来获取对应的HTML页面
  - 提取HTML页面里的有用数据，如果是需要的数据就保存起来，如果页面里面是其它的URL，那么就继续执行第二步
- 需要技能
  - 如何抓取页面
    - HTTP请求处理，urllib处理后的请求可以模拟浏览器发送请求，获取服务器响应的文件
  - 解析服务器响应的内容
    - re(正则)、xpath、BeautifulSoup4、jsonpath、pyquery
    - 目的是使用某种描述性语法来提取匹配规则的数据
  - 如何采取动态HTML、验证码的处理
    - 通用的动态页面采集
      - Selenium+PhantomJS（无页面浏览器），模拟真实浏览器加载js、ajax等非静态页面
      - Tesseract
        - 机器学习库、机器图像识别系统（识别图片中的文本）
  - Scrapy框架
    - 中国常见的框架Scrapy、Pyspider
    - 高定制性高性能（异步网络框架twisted），所以数据下载数度非常快，提供了数据存储、数据下载、提取规则等组件
  - 分布式策略
    - Scrapy-redis

        - 在Scrapy的基础上添加了一套以Redis数据库为核心的一套组件，让Scrapy框架支持分布式的功能，主要在Redis里做请求指纹去重、请求分配、数据临时存储
- 爬虫与反爬虫与反反爬虫三角之争
    - 最头痛的人
        - 爬虫做到最后，最头痛的不是复杂的页面，也不是晦涩的数据，而是网站另一头的反爬虫人员
    - 反爬虫技术
        - User-Agent
        - 代理
        - 验证码
        - 动态数据加载
        - 加密数据
- 通用网络爬虫
    - 概念
        - 收索引擎用的爬虫系统
    - 用户群体
        - 收索引擎用的爬虫系统
    - 目标
        - 尽可能把互联网的所有页面下载下来，放到本地服务里形成备份。再对这些网页做相关处理（提取关键字、去掉广告等），最后提供一个用户检测接口
    - 抓取流程
        - 选取一部分已有的url，把这些url放到待爬队列
        - 从队列中提取这些url，然后解析DNS找到主机IP，然后去这个IP对应的服务器里下载HTML页面，保存到收索引擎的本地服务器里，之后把爬过的url放入已爬取队列
        - 分析这些页面，找出页面里的url链接，继续执行第二步，直到爬取条件结束
    - 收索引擎如何获取一个新网站的url
        - 主动向收索引擎提交网站（百度站长平台）
        - 在其它网站里设置网站的外链接
        - 收索引擎会和DNS服务商合作，可以快速收录新的网站
    - 爬虫需要遵守的规则
        - Robots协议
    - 通用爬虫工作流程
        - 爬取网页
        - 存储数据
        - 内容处理
        - 提供检索、排名服务
    - 收索引擎排名
        - PageRank值
            - 根据网站的流量
        - 竞价排名
            - 谁钱多谁排名高

- 通用爬虫的缺点
  - 只能提供和文本相关的内容（HTML、Word、PDF）等，但是不能提供多媒体（音乐、图片、视频）和二进制文件（程序、脚本）等
  - 提供结果千篇一律，不能针对不同人群提供不同的收索结果
  - 不能理解人类语义上的检索
- 聚焦网络爬虫
  - 概念
    - 爬虫程序员写的针对某种内容的爬虫
  - 特点
    - 面向主题爬虫、面向需求爬虫
      - 会针对某种特定的内容去爬取信息，而且会保证信息和内容需求尽可能相关
- 增量式网络爬虫
- 深层网络爬虫
- URL
  - 统一志愿定位符，是互联网上的资源地址