第01讲 爬虫基础

2018年8月

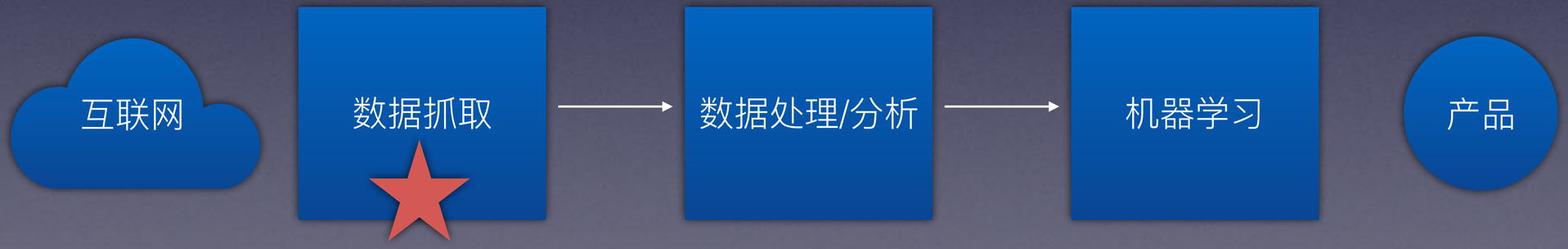
1.1 大数据+人工智能时代

- 吴军：《智能时代》（大数据与智能革命重新定义未来）；
- 与数据相比，大数据的特点：体量大、维度高、及时性强；配套工具飞速发展
- 举例：托勒密、哥白尼、第谷、开普勒（教会迫害科学or数据问题）
- 更多的例子：AlphaGo、谷歌搜索的数据优化、普拉达的销售策略、特斯拉的汽车智能终端定位、勇士队的崛起。。。

1.2 爬虫的价值

爬虫（英文：spider, crawler, gecco） —— 从互联网、文件等**资源**，按照一定规则，**抽取**所需信息的程序或脚本文件。

- 大公司与小公司之别
- 为网页展示提供内容
- 为数据处理提供原始数据
- 为量化分析提供时间序列数据
- 为自然语言处理提供语料



产品未动，数据先行。爬虫往往是产品线上面对外部网络的第一层：特异情况多且复杂，既要面对各种反爬，也要有一定的系统设计能力。

1.3 爬虫的合法性问题：关于爬虫协议

- 爬虫协议：robots.txt。全称是“网络爬虫排除标准”，网站通过Robots协议告诉搜索引擎哪些页面可以抓取，哪些页面不能抓取。
- 如何查看爬虫协议：网站根目录下 /robots.txt
- 举例1：今日头条 https://www.toutiao.com/robots.txt
- 举例2： 新浪新闻：https://news.sina.com.cn//robots.txt

robots.txt

- User-agent： 搜索引擎种类（用户代理：user-agent，可以让服务器识别客户使用的操作系统及版本、浏览器类型和版本）
- disallow：不允许抓取的部分
- 举例：

```
Disallow: /user/
Disallow: /group/
Disallow: /?*
Disallow: /*?*
```

如果不遵守爬虫协议的话。。。

- robots.txt是一个“君子协议”?
- 景点与“到此一游”
- 违反“爬虫协议”能否适用反不正当竞争法? （https://www.chinacourt.org/article/detail/2013/06/id/1001434.shtml）
- 百度与360“爬虫”之争（http://money.163.com/14/0807/18/A32LGORL00254TI5.html#from=relevant#xwwzy_35_bottomnewskwd）

1.4 爬虫分类：通用爬虫与聚焦爬虫

- 通用爬虫：爬虫从一个起始网址进入，抓取全站所有信息，用户主要是大型门户网站、搜索引擎等大型web服务；（谷歌、雅虎、百度、搜狗、Naver、Yandex）
- 聚焦爬虫：只抓取定义好要抓取的内容，适用于快速获取所需信息的业务；（头条的新闻爬虫、高德的地理信息爬虫、雪球的金融数据爬虫）

2.1 客户端与服务端

- 客户端（client）：为用户（浏览器、手机等终端使用者）提供本地服务的程序
- 服务端（server）：为服务器（云服务、主机）提供服务的程序
- 请求（request）：客户端发向服务端，寻求服务
- 响应（response）：服务端发向客户端，提供服务

2.2 HTTP协议与HTML

- HTML：全称HyperText Markup Language，超文本标记语言，诞生于1960年（Ted Nelson）（互联网诞生于1969年）；
- HTTP：全称HyperText Transfer Protocol，超文本传输协议（https://www.ietf.org/rfc/rfc2616.txt）
- 运输工具与高速公路的关系

2.3 URI和URL、URN

- URI（网络资源ID）：统一资源识别符（Uniform Resource Identifier）
- URN（只命名，不定位资源）：统一资源名称（Uniform Resource Name），举例：磁力链接、ISBN码
- URL（命名+定位资源）：统一资源定位符（Uniform Resource Locator）
- 三者的关系：URI = URN + URL

2.4 超文本（HyperText）

- 普通文本：纯文字的作文；
- 超文本：除了文字、还有图片、音频、视频、超链接等资源：<img>标签里面是图片、<audio>标签里面是音频，<vedio>标签里面是视频；
- www.jd.com

2.5 HTTPS

- URL开头可能是http、https、ftp、mailto、telnet
- 爬虫通常要抓取的就是http和https
- HTTPS（HyperText Transfer Protocol over Secure Socket Layer），是HTTP的安全加强版，即HTTP加入了SSL；SSL是网景公司研发的数据传说安全协议：通过加密的西西安全通道保证数据传输安全；https网站可以通过地址栏的锁头标志查看其信息
- 淘宝网 https://www.taobao.com/
- 人民网 http://www.people.com.cn/

2.6 HTTP请求过程

请求

响应

客户端

服务端

- 浏览器输入一个网址URL —— 看到网页。动作分解：
- 1. 浏览器向网站所在服务器发送一个请求；
- 2. 服务器对请求进行处理和解析
- 3. 服务器向浏览器返回响应
- 4. 浏览器对响应做解析，呈现网页

Chrome 开发者模式

- https://www.toutiao.com/
- Chrome浏览器，点击鼠标右键，选择检查；
- name, status, Type, Initiator, Size, Time, Waterfall
- 点击第一项：可以看到Headers, Preview, Response, Cookies, Timing

2.7 请求：请求方法

- 最常见的两种请求方法：get和post
- get方法：1.在网址栏里输入一个网址；2.在搜索框中输入搜索词
- 举例：https://search.jd.com/Search?keyword=python&enc=utf-8&wq=python&pvid=4cd526a2f9374b858c8c8d42ed9be27f
- post方法：表单登录

get和post比较

- get请求的参数包含在url中，可以在url中看到，post请求url不包含参数，数据都是通过表单传输的，包含在请求体中；post比get要安全；
- get请求提交的url长度有限制，不同浏览器不同版本的长度限制不同；post方法的数据大小限制，取决于表单的设置，可以远远大于get请求的数据。

其它请求方法

- HEAD：用于获取报头
- PUT：从客户端向服务器传送的数据取代指定文档中的内容
- DELETE：删除服务器中指定的内容
- 参考：http://www.runoob.com/http/http-methods.html

请求网址和请求体

- 请求网址：即url
- 请求体：post请求中的表单数据；get请求的请求体为空

请求头

- Accept：请求报头域，用于指定客户端可以接受哪些类型的信息；
- Accept-Encoding：客户端可接受的内容编码；
- Accept-Language：客户端可接受的语言类型
- Cache-control：客户端对缓存的设置
- cookie：网站为了跟踪用户而存放在用户本地的数据；
- Upgrade-insecure-requests：1 Chrome可以自动将http请求升级为https的安全传输；
- user-agent：用于让服务器识别客户使用的操作系统及版本、浏览器及版本；可用于伪装浏览器

2.8 响应：响应状态码

- 用于表示服务器的响应状态：
- 最常见的：
- 200：服务器成功处理了请求
- 301：网址重定向
- 403：服务器拒绝此请求
- 404：服务器找不到请求的资源
- 503：服务器目前无法使用

响应头

- date：日期和时间标记
- expires：指定响应的过期时间；
- server：服务器的名称、版本
- age：缓存处理的时间
- content-length：响应长度
- content-type：响应类型