基于大数据的个性化电商虚拟平台

创新实践大数据一组

胡杰、潘李凡、虞圣赟、施雯、杨彤、莫周婷

目录

一、 项目背景

目前各类电商平台如雨后春笋般出现，除了淘宝天猫、京东、亚马逊等大型电子商务平台外，例如聚美优品、1 号店等中小型或者是特卖型电商平台非常繁多。消费者在购买时往往会担心各种各样的问题，会不会是假货？一般多久可以收到？其他消费者的使用体验怎么样？这家店铺的整体情况如何？在什么地方买更便宜？作为一个第三方平台，我们的目的在于以更为直观的方式为用户呈现不同电商平台上的消费信息，使用户能够更快更好性价比更高地购买到自己所需要商品。

伴随“互联网+”概念的提出，国家对于互联网产业的大力支持，以及电子商务、互联网产业对于我国经济大力发展的促进作用。人们已经越来越依赖于互联网，慢慢习惯于网上购物、手机支付等新型互联网消费模式，对其的需求也越来越高。网购，一把新的双刃剑，一方面人们享受着它所带来的丰富多样与方便快捷，一方面也承受着假货，无法实际试用，退换麻烦等诸多传统购物中不太会碰到或是容易解决的问题。

本项目将会为用户展现不同电商平台中相同商品的实时比价，实时用户评论，商品的实时热度比较，品牌的实时热度排行，以及店铺的实时状况排行，建立包含类别、品牌、产品、店铺、评价等多维数据模型，为用户推荐最合适的商品。

二、 项目预期成果与效益

建立面向消费者的购物商品分析数据平台，之后面向企业推出我们的个性化服务，通过我们平台积累的数据信息，有效地帮助企业吸引消费者。

预期项目成果

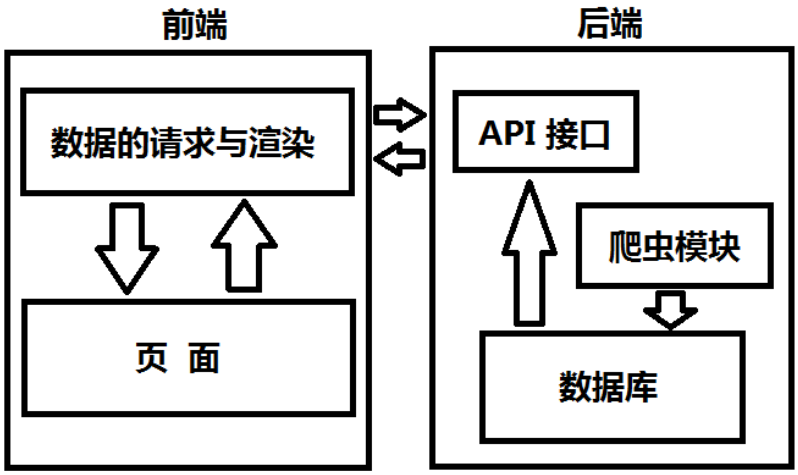
- 尽可能多、尽可能全面的电商平台的信息，从而吸引更多的消费者访问本平台，提高平台占有率。
- 更加高效的爬虫程序，提高了信息的实时性
- 更加精确的数据分析模型，提高推荐排行的准确性
- 个性化体验、更快速的页面请求响应，提高用户体验，增强用户粘度

盈利点剖析

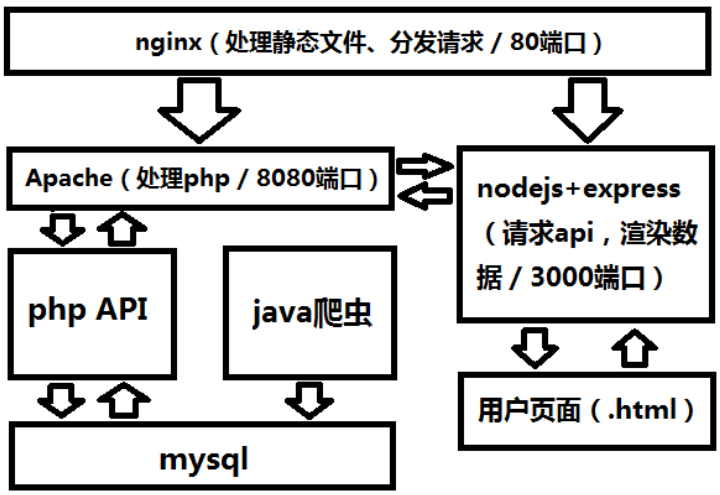
- 平台广告
- 个人 VIP 用户，针对个人用户的 VIP 定制化服务
- 企业用户，针对企业产品的个性化指导服务

三、 项目核心架构图

前后端分离示意图



服务器架构示意图



四、 项目演示地址

http://120.27.130.203

五、 项目核心技术说明

- 数据库: mysql
- 爬虫模块: java
- 数据处理模块: matlab / java
- API: php
- 数据渲染: nodejs
- 页面呈现: html + css + js

本项目采用**完全前后端分离**的技术架构，后端服务 API 化，前端服务 Node 化，后端以 LAMP 作为业务处理的主要手段，配合自主研发的爬虫模块与数据分析模块，进行数据的采集、存储与分析。前端利用 echart、d3 等 javascript 图像类库完成数据可视化。

六、 前端技术详细说明与页面操作指南

1) 页面设计与优化

首页 -> 搜索 -> 产品汇总页面 -> 产品详细页面

-> 品牌页面

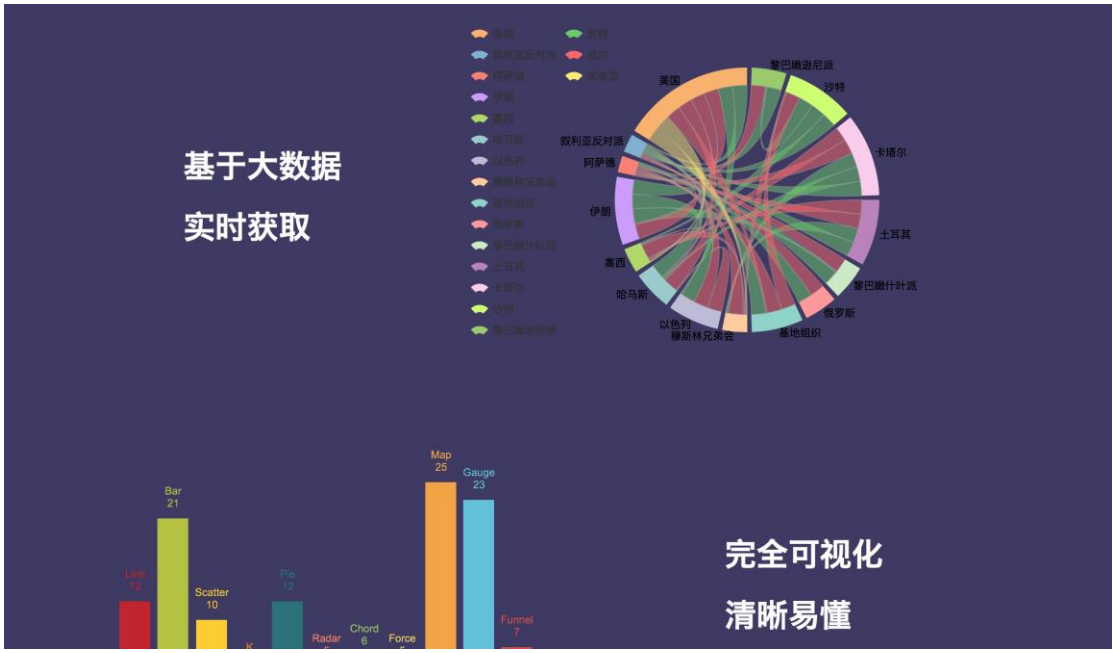
-> 类别页面

为加快页面响应速度，页面 css，js，图片都经过压缩处理。

- 图片压缩平台：https://tinypng.com
- css 压缩平台：http://tool.lu/css/
- js 压缩平台：http://tool.oschina.net/jscompress

- **首页**


基于海淘的数据分析与可视化
奶
搜索
产品
【Friso gold 美素佳儿金装】荷兰原装进口婴儿奶粉
立顿绝品醇奶茶40条装台式意式英式日式赠礼盒
【新升级配方】Nutrilon诺优能幼儿配方奶粉3段双罐装
品牌
旺仔牛奶
香飘飘奶茶
类别
奶粉
奶茶
店铺
小奶花旗舰店
同类品牌
旗下产品
同品比价
店铺PK

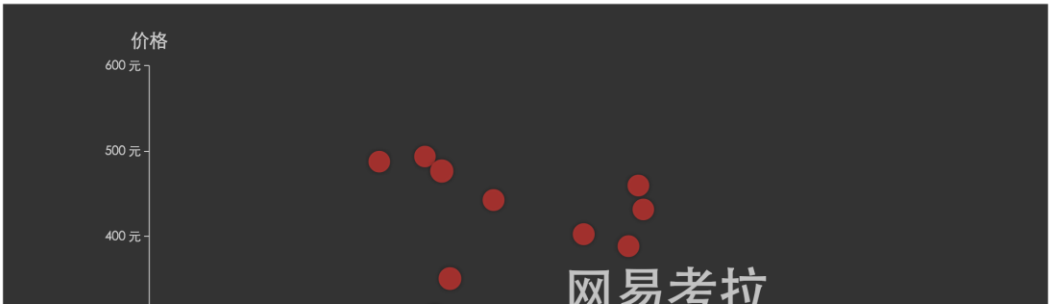

基于大数据
实时获取
美国
沙特
卡塔尔
土耳其
黎巴嫩什叶派
俄罗斯
基地组织
穆斯林兄弟会
以色列
哈马斯
塞西
伊朗
阿萨德
叙利亚反对派
Line 12
Bar 21
Scatter 10
Pie 12
Radar
Chord 6
Force
Map 25
Gauge 23
Funnel 7
完全可视化
清晰易懂

- **奶粉信息页面：**

2015/10/17 下午9:25:24　2015/10/18 上午10:16:07　2015/10/19 下午3:22:31

品牌编号	品牌名	当前销量	销量增值	排名变化
6	Cow & Gate	17590	15831	↑5
4	Bellamy's/贝拉米	8765	6489	↑2
5	Friso/美素佳儿	4000	3643	↑2
1	Nutrilon/诺优能(牛栏)	2345	361	↓3
3	Wyeth/惠氏	556	224	↓2
2	Aptamil	10435	81	↓4

奶粉销量动态图



- **产品汇总页面：**

旗下产品

- **详细产品页面：**


看看大家都在比
奶粉
￥180
奶瓶
￥69
尿不湿
￥130
婴儿车
￥380
学步鞋
￥78
NEUE REZEPTUR
Aptamil
Prebiotics
Kindermilch
产品名称：牛栏2段婴儿原装保税区荷兰进口奶粉二段诺优能2海外直邮
Nutrilon
所属品牌：Nutrilon/诺优能(牛栏)
产品规格：850g
适用年龄：6-10个月
产地：荷兰
比质比价
用户评价
价格最优top3
1.
天猫
TMALL.COM
天猫价格：￥149.56/500g
点此链接

- **品牌页面：**

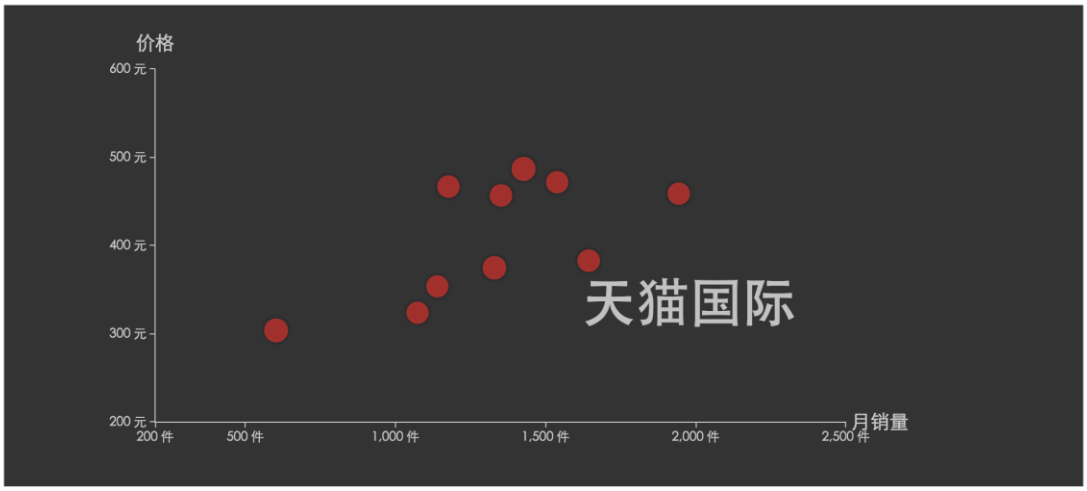

同类品牌
Pampers
“天王嫂”昆凌都惊呆的
“魔法纸尿裤”


奶粉 热门品牌 hot brand
Aptamil 爱他美
德国原产爱他美
Nutrilon 诺优能
Aptamil 爱他美
Abbott 雅培

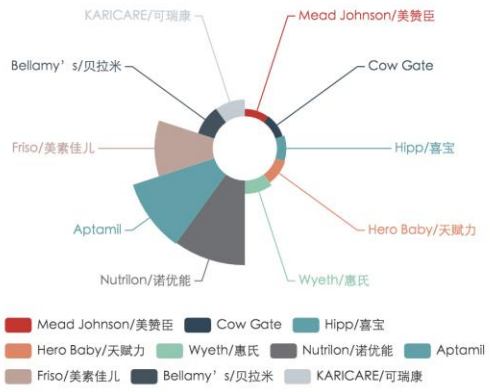
2) 数据可视化的探究

熟练使用数据可视化工具 d3，本学期新接触了一款开源、功能强大的数据可视化产品 echarts，是基于 Canvas 的，纯 Javascript 的图表库，提供直观，生动，可交互，可个性化定制的数据可视化图表。创新的拖拽重计算、数据视图、值域漫游等特性大大增强了用户体验，赋予了用户对数据进行挖掘、整合的能力。利用 echarts 对本学期奶粉比价项目中的相关数据进行了可视化，制作了一些实用炫酷的图表，如

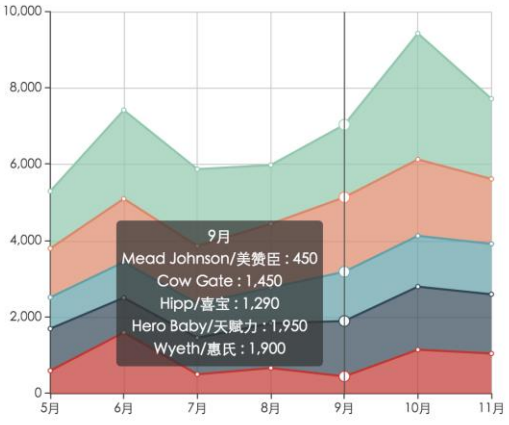
奶粉销量动态图

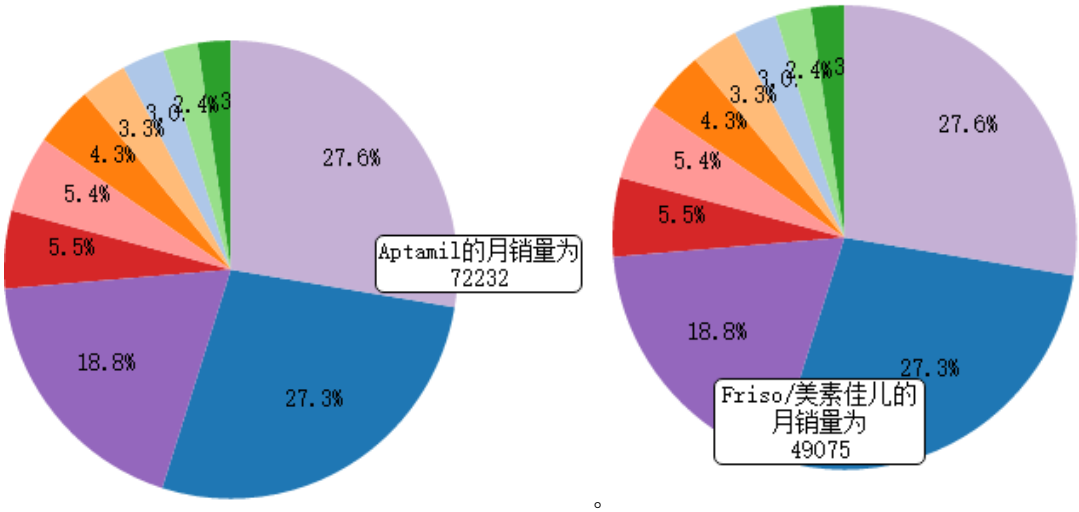


销量前十奶粉品牌占比

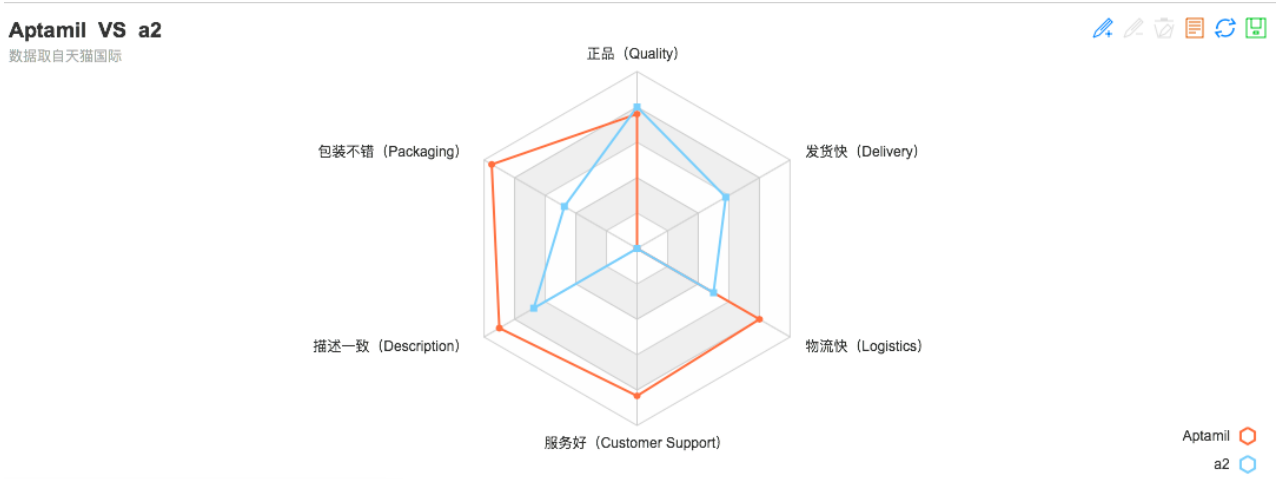


各奶粉品牌销量堆叠图





。



3) 前端 Node 的渲染

Node.js 是一个基于 Chrome JavaScript 运行时建立的平台，用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用事件驱动，非阻塞 I/O 模型而得以轻量和高效，非常适合在分布式设备上运行的数据密集型的实时应用。

Node 是一个 Javascript 运行环境(runtime)。实际上它是对 Google V8 引擎进行了封装。V8 引擎执行 Javascript 的速度非常快，性能非常好。Node 对一些特殊用例进行了优化，提供了替代的 API，使得 V8 在非浏览器环境下运行得更好。

本项目主要是利用 node 来请求相应页面的 API，获取数据后，渲染到前端页面。这样可以使前端专注于页面，后端专注于 API，前后端责任更加清晰；能够提高工作效率，完成 API 的制定后，能够完全并行开发；本地开发更加方便，不需要过多涉及不了解的内容。

七、 后端技术详细说明与 API 接口

1) API 的设计与说明

1. 获取类别列表

接口功能

> 获取商品类别列表

URL

> /api/typelist.php

支持格式

> JSON

HTTP请求方式

> GET

请求参数

> null

返回字段

返回字段		字段类型	说明
status		int	0: 正常; 1: 错误
type	id	int	类别编号
	name	string	类别名称

接口示例

> 地址：/api/typelist.php

```
{
    "status": 0,
    "type": [
      {
        "id": 1,
        "name": "奶粉"
      },
      {
        "id": 2,
        "name": "纸尿裤"
      }
    ]
}
```

2. 获取品牌列表

接口功能

> 获取商品品牌列表

URL

> /api/brandlist.php

支持格式

> JSON

HTTP请求方式

> GET

请求参数

参数	必选	类型	说明
type	true	int	商品类别的编号

返回字段

返回字段		字段类型	说明
status		int	0: 正常; 1: 错误
brand	id	int	品牌编号
	name	string	品牌名称

接口示例

> 地址：/api/typelist.php?type=1


```
{
    "status": 0,
    "brand": [
      {
        "id": 1,
        "name": "Nutrilon/诺优能(牛栏)"
      },
      {
        "id": 2,
        "name": "Bellamys贝拉米"
      }
    ]
}
```


3. 首页搜索

接口功能

> 首页搜索栏, 输入关键词, 获取类别、商品、品牌、店铺组合信息

URL

> /api/search.php

支持格式

> JSON

HTTP请求方式

> POST

请求参数

参数	必选	类型	说明
key	true	string	输入的关键词

返回字段(product, brand, type, shop每项至多5条)

返回字段		字段类型	说明
status		int	0: 正常; 1: 错误
product	id	int	产品编号
	name	string	产品名称
brand	id	int	品牌编号
	name	string	品牌名称
type	id	int	类别编号
	name	string	类别名称
shop	id	int	店铺编号
	name	string	店铺名称

接口示例

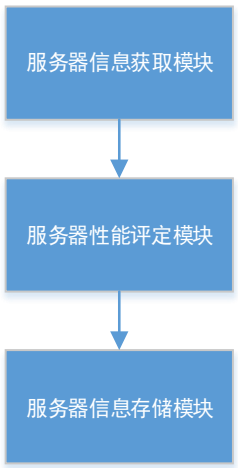
地址：/api/search.php


```
{
    "status": 0,
    "product": [
      {
        "id": 1,
        "name": "【Friso gold 美素佳儿金装】荷兰原装进口婴儿奶粉"
      },
      {
        "id": 2,
        "name": "立顿绝品醇奶茶40条装台式意式英式日式赠礼盒"
      },
      {
        "id": 3,
        "name": "【新升级配方】 Nutrilon诺优能幼儿配方奶粉3段双罐装 荷兰进口"
      }
    ],
    "brand": [
      {
        "id": 1,
        "name": "旺仔牛奶"
      },
      {
        "id": 2,
        "name": "香飘飘奶茶"
      }
    ],
    "type": [
      {
        "id": 1,
        "name": "奶粉"
      },
```

```
        {
          "id": 2,
          "name": "奶茶"
        }
      ],
      "shop": [
        {
          "id": 1,
          "name": "小奶花旗舰店"
        }
      ]
}
```


2) 爬虫模块的设计与实现

- **代理服务器相关模块设计**



- **信息获取模块**

功能：从提供代理服务器信息的网站，爬取一系列代理服务器信息。

技术手段：由于代理服务器网站提供的信息相对比较少，也变化不大，所以使用简单的爬取逻辑，爬取周期就可以满足此功能需要。本模块采用 Apache 提供的 http 网络访问 API 获取网页源码，使用 JSOUP 框架方便的将网页源码解析成可供使用的 Document 对象，进而实现对目标信息的提取。

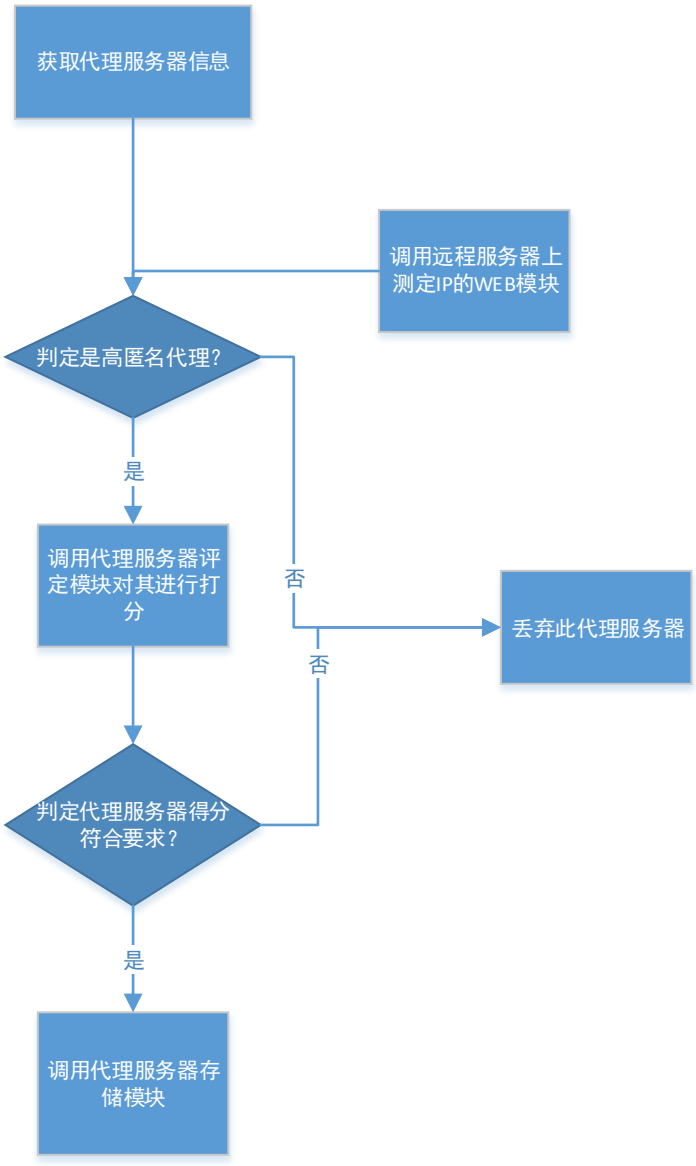
缺陷：由于不同网页页面布局不同，导致我们的目标信息分布不同，需要人工对不同网页进行适配工作，已实现自动对不同网页进行解析的功能

- **服务器性能评定模块**

功能：为了获取更加优良的代理服务器资源，本模块对采集到的代理服务器信息进行评定，给出最后得分，分数越低的代理服务器，性能越优良。

技术手段：

总体概要设计：



详细设计

- **远程服务器测定 IP 的 web 模块**

功能：因为在本项目中，只有高匿名的代理服务器才真正有用。为了获取代理服务器的类型，我们采用测定 IP 的方式判定。

技术手段：本模块基于 JSP 动态网页技术标准，使用 JSP 提供的 API 可方便的获取请求 URL 的用户 IP，并且，本模块已经搭载到远程的基于 linux 的 Tomcat web 应用服务器上，可以实现实时对目标用户 IP 的测定。

测试 URL: http://120.27.130.203:8080/GetIP/getip

- **代理服务器评定模块**

功能：对获取的代理服务器信息进行性能评估，并计算得分。

技术手段： 初步建立代理服务器性能评价模型 L=W1*T+W2*R

符号说明：

L 值越小，代理服务器性能越好

T 代表平均连接时间（初步定为 10 次测定的平均时间）

R 连接失败率（测定失败次数/测定总次数）

W1,W2 分别为 T 和 R 的权值

- **代理服务器信息存储模块**

功能：对符合要求的代理服务器信息进行数据持久化的存储操作。方便爬虫程序的调用

技术手段：本模块基于 MySQL 数据库进行实现，使用 JAVA 提供 JDBC 的 API，实现程序对数据库的各种操作。

数据库表结构初步设计如下：

PsHost	PsProtocol	PsType	PsLocation	PsUpdatetime	PsLinkTime	PsFailRate	PsGoal
110.73.82.142:8123	0	1	广西南宁	2015-12-02 13:43:30	0.4202	0	0.3291
218.90.216.170:63000	0	1	江苏泰州	2015-12-02 13:43:31	0.4325	0	0.2693
182.90.21.132:80	0	1	广西梧州	2015-12-02 13:43:32	0.447	0	0.5237
122.232.224.122:3128	0	1	浙江嘉兴	2015-12-02 13:43:32	0.5094	0	0.3597
171.38.26.9:8123	0	1	广西玉林	2015-12-02 13:43:32	0.4389	0	1.2782
182.89.11.247:8123	0	1	广西柳州	2015-12-02 13:43:32	0.4768	0	0.8666

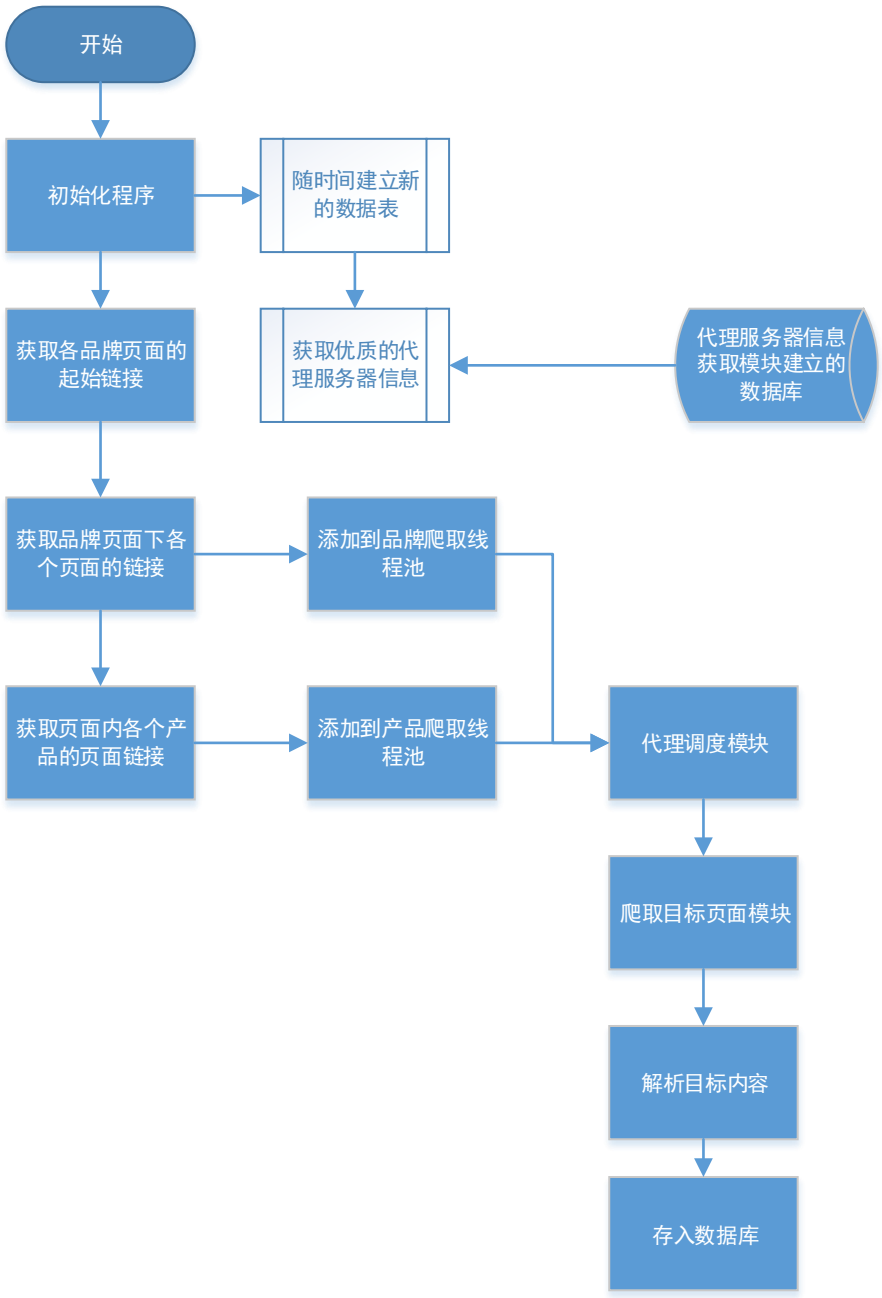
- **大数据项目爬虫逻辑设计**

利用上学期掌握的爬虫技术，结合我们这个学期的项目需求，设计了比较完善稳定的后台多线程自动切换最适合代理爬取网站内容，解析，存取数据的逻辑。其中主要功能均已经实现。

设计思想：

为了能够更高的使用利用多线程并发，提高爬虫运行效率的特性，本逻辑采用双线程池的设计，1.用来爬取品牌下各页面的线程池；2.用来爬取产品信息的线程池。

业务流程图：



详细设计：

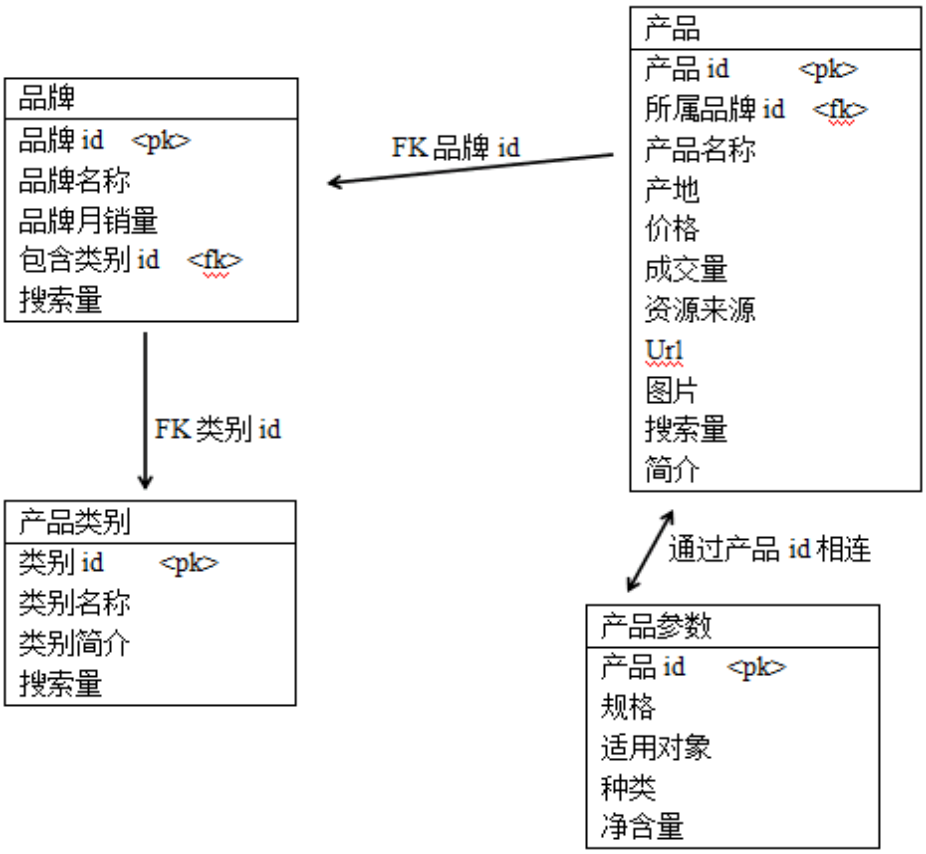
网页的爬取主要采用 Apache 提供的网络访问 API 实现；解析目标信息主要采用 JSOUP 框架 API 非常方便的将网页源码转化为 Document 树，进而获取到期望数据；目前本项目的数据库操作还是使用 JAVA 提供的 JDBC 的 API，实现对 MySQL 的方便操作。

本学期最核心的就是一直解决如果使用代理，使得目标服务器不容易发现本项目的爬虫，进而使得多线程爬虫能够正常长时间运行，实现业务需要的问题。

针对这个问题，我们专门在爬虫逻辑内，添加了代理服务器调度模块。本模块的思想主要是在获取优质代理服务器中随机调用，且同一代理服务器相邻的调用时间间隔也随机确定，此外还严格控制调用的代理服务器与使用本代理的线程运行数量，最大限度的降低代理服务器被目标服务器屏蔽的风险。

3) 数据库的设计与优化

- **数据库表结构设计**



- **在服务器上用存储过程实现数据库操作**

```
create procedure deleteProductPara(dpp_nature varchar(20),
dpp_naturedata varchar(20))
begin
declare my_sql1 varchar(500);
set my_sql1='delete from productPara where ';
set my_sql1=concat(my_sql1,dpp_nature,'=',dpp_naturedata);
set @ms=my_sql1;
prepare s1 from @ms;
execute s1;
deallocate prepare s1;
end
//
```

```
List<productPara> pups = new ArrayList<productPara>();
CallableStatement stmt = null;
try {
    stmt =(CallableStatement) conn.prepareCall("{call selectProductPara(?,?)}");
    stmt.setString(1, selectNature);
    stmt.setString(2, selectNatureData);
    stmt.execute();
    ResultSet rs = stmt.executeQuery();
    while (rs.next()) {
        productPara pup = new productPara();
        pup.setProId(rs.getLong("ProId"));
        pup.setProSize(rs.getString("ProSize"));
        pup.setProObject(rs.getString("ProObject"));
        pup.setProType(rs.getString("ProType"));
        pup.setProNetContent(rs.getString("ProNetContent"));
        pups.add(pup);
        System.out.println(rs.getLong("ProId")+"  "+rs.getString("ProSize")+"  "+rs
    }
} catch (Exception e) {
```

1.

此方法依旧用之前的 jdbc 连接方式进行连接，与 jdbc 不同的是在服务器上建立存储过程，在 java 工程中调用时，只需要直接调用已建好的存储过程即可，相比之前的算法更加高效。

- **用 hibernate 框架实现数据库操作**

Hibernate 为将数据库表与对象之间进行映射的框架，运用此框架可以使对数据库的操作更加模块化，更加安全。

4) 数据分析模块的探究

- **研究商品相似度比对算法**

在项目进行过程中，我们遇到了一个问题：同一商品,在各购物网站的关键词可能不同。例如:平台一：华为 U8860 (honor 荣耀)高清高速云服务手机(晶钻黑,联通定制机)下单立减 500 元；平台二：华为(HUAWEI) Honor U8860 3G 手(黑)WCDMA/GSM 非定制。两者描述的是

同一件商品,但是描述上却有差异。如果系统无法识别两个不同描述的产品为同一产品，就无法比价，系统就无法实现。

因此,还需要对查询到的结果进行比对和匹配,确定哪些是相似的商品。

查找了很多相关论文后，我们找到了一个通过词频比对产品相似性的方法。

1、识别查询接口。搜索框和查询按钮通常都是邻接在一起的；搜索框和查询按钮在页面上的位置处于页面靠上,高度一般位于 Opx~192px 的范围内,并且在页面靠近中间的位置

2、发送查询请求&提取商品价格信息。找到搜索框和查询按钮的位置,然后将查询的文本赋值给搜索框,最后触发查询按钮的点击事件即可实现&中文分词

3、商品价格比较。根据商品的描述进行关联程度的分析。商品描述主要由商品的特征及规格等词项组成，比对后向量空间模型来判断相似度（奶粉：品牌产地、适合年龄、含量

4、信息存储与索引模块的设计。采用数据库和索引文件两种存储方式,数据库保存信息,调用显示产品信息, 索弓 I 文件提供关键词索弓 I。调用 Lucene 搜索机制通过索引文件进行查询,获得数据库中的信息,返回符合查询条件的结果。

有了思路之后，我们用 java 实现了 TD－IDF 关键词比较的模型，实现了比较各个网站关键词的频率，但是知道了关键词的频率对判断出它们是同一个产品远远不够，我们遇到了新的问题：关键词太多且同一关键词词频差异较大，系统无法直接比较词频大小判断同款产品；分词词库的限制，有些词语被拆开失去了语义，比如奶粉品牌，这给产品匹配带来了巨大困难。我们试图通过建立测试集，利用 matlab 对测试结果进行分析以优化模型，但是受限于分词技术中的词库和本学期创新实践课的时间，只得另辟蹊径。

- **构想了商品匹配的算法**

通过**规格、重量、适用年龄、适用阶段、原产国**；

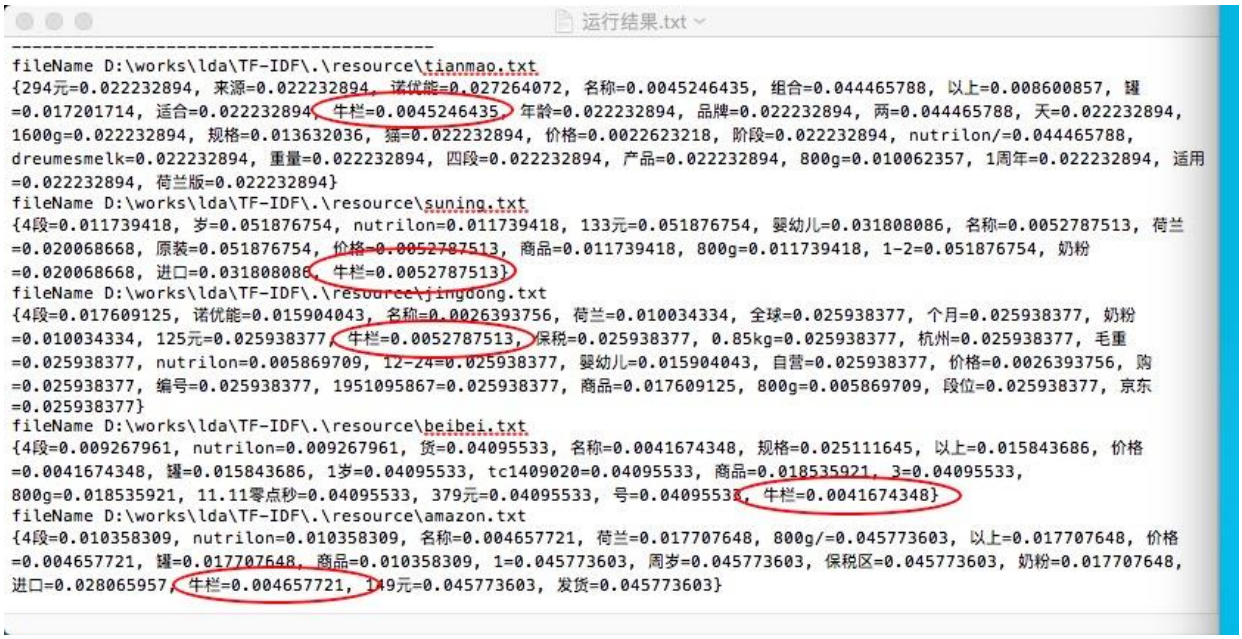
和**品牌、系列、包装种类、分类**这两部分字段进行关键字比对，

1. 第一部分关键字的值如果存在，必须完全一样才判断为同一产品；

2. 第一部分判断完毕后如果符合条件的产品很多，判断第二部分

3. 对第二部分筛选出匹配度达 X％的结果（x 值根据数据库中存入数据的结果情况而定）

PS：对已判断为同一产品的商品进行聚合，减少搜索的时间



八、 项目日志

第十四、十五周 (16.1.7)

- 学期项目总结
- 整体项目文档的撰写

第十二、十三周 (15.12.24)

- 前端页面的压缩与优化 s
- 服务器部署完善，爬虫模块的迁移
- Node 与 api 调用
- 利用 echart3 重构部分图表
- 实现了 hibernate 单表映射
- 对于产品比对的进一步探究

第十一周 (15.12.10)

- 类别与产品页面的完善
- Echart3 的了解
- 热搜 API 的实现
- 修改数据库表结构
- 学习 hibernate 框架，并实现基本操作
- Matlab 的学习，对于测试结果的分析与优化
- 实现了 web 程序，自动返回 ip 的功能
- 在调用代理服务器的逻辑中加入了随机时间

第十周 (15.12.3)

- 前后端分离的思考
- 建立了代理服务器性能评价模型
- 完善代理服务器数据表的设计
- 建立商品比对测试集，利用 TF-IDF 算法产生测试结果

第九周 (15.11.26)

- 查找关于代理服务器方面的专利信息
- 关键词比较模型的建立
- 利用 php 将数据库数据自动生成 json
- Mysql 存储过程的学习
- 平台 API 的设计

第八周 (15.11.19)

- RESTful API 的构想
- 综合文档筹划
- 测试集建立

第七周 (15.11.08)

- 研究商品相似度（同种商品出现在不同的电商平台中，如何判断是完全相同商品）
- 针对奶粉建立数据模型，拟合不同电商平台上的奶粉产品
- 爬虫的代理访问
- 爬虫的模拟登录

第五、六周 (15.11.02)

总体方向：做多网站产品比价平台 (海淘)

前端:

- 首页界面的完善
- 数据图 + 动态表格 + 首页三者整合 "首页" -> "品牌页面" -> "产品页面 (含有比价)"
- 数据图的完善与补充 (增加价格的数据维度，比价数据展现的构想)

后端:

- 从天猫国际 - 奶粉开始，爬取奶粉比价需要的各个属性
- 根据比价需要的属性完善数据库设计
- 完成多平台—奶粉数据的爬取工作
- 配合前端数据可视化需要，实现一定程度的与前端交互

第四周 (15.10.19)

数据源：天猫国际板块 - 母婴用品 - 奶粉商品

提取数据特征：奶粉品牌与其对应的销量

分析与呈现：

- 品牌销量比重 = 该品牌销量 / 总销量
- 销量增长值 = 当前销量 - 上一次采集销量

根据销量比重绘制柱状图，向用户推荐奶粉品牌

根据销售增长量绘制动态表隔，呈现近期奶粉品牌的销量增长趋势

需要做的工作有：

- 天猫国际奶粉商品相应品牌与销量的数据采集
- 采集数据的存储
- 销量数据图、数据表的绘制
- 呈现页面的设计与美化

九、 项目计划

- 大数据一二组的大整合，其中包括：服务器静态资源（可以共用图表类库）、页面 UI、数据存储仓库等等。
- 前端工程化，尝试使用一些自动化构建工具，甚至是框架。
- 结合我们项目的实际需求，配合项目前端与业务逻辑层，实现对各个模块需要的数据的精确爬取解析，完成平台前后端的无缝连接。