

**《大数据处理技术》综合实验报告**

系统名称： 硕士研究生招生数据分析系统

班 级： 软件1705(创新）

学生姓名： 薛程元

**2019年 12 月 20 日**

目录

[1. 系统概述 1](#_Toc27741788)

[2. 数据采集模块开发 2](#_Toc27741789)

[3. 数据预处理模块开发 8](#_Toc27741790)

[4. 数据仓库开发 9](#_Toc27741791)

[5. 数据分析模块开发 11](#_Toc27741792)

[6. 数据导出模块开发 16](#_Toc27741793)

[7. 报表展示模块开发 18](#_Toc27741794)

[8. 系统特色及总结 21](#_Toc27741795)

# 1. 系统概述

1.1 系统背景介绍

随着本科生越来越多，本科学历已经不足以满足部分用人单位的需求了，所以用人单位就提高了招聘的门槛，纷纷打上了“研究生以上学历”的标签。因此，很多考生为了提升自己的就业竞争力，纷纷加入考研的队伍。硕士研究生招生数据分析系统的数据来源于研招网，所有公开的招生信息均由各自招生单位提供，已在各招生单位网站进行了公开公示。研招网只有单纯的信息展示，并没有对数据进行统计分析，所以为考生以及其他关注考研的人们提供数据分析结果，在此本系统获取到2020年全国硕士研究生招生考试数据，并对数据进行处理与分析以及最后结果展示。

1.2 系统架构设计

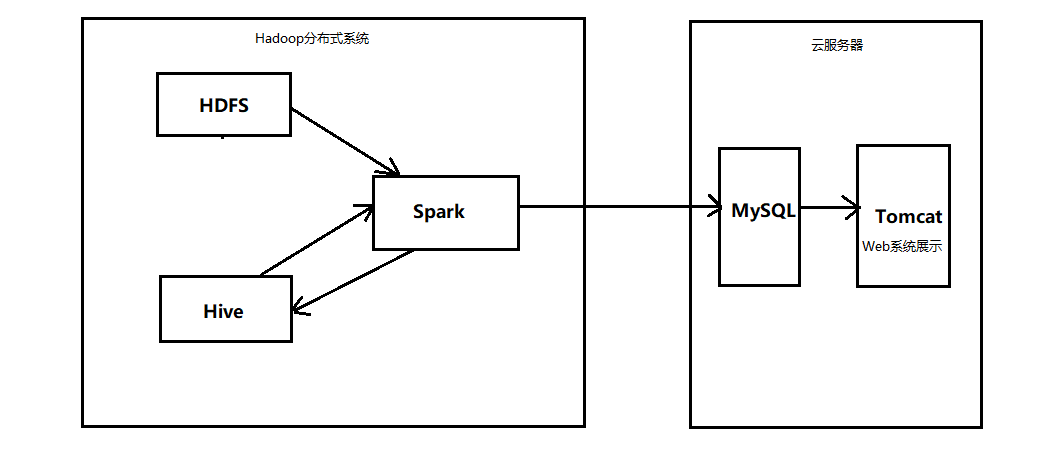


图1.1 硕士研究生招生数据分析平台

* 首先，在服务器运行的网络爬虫程序获取数据以json格式保存，同时使用shell脚本分批导入Hadoop分布式系统中的HDFS中。
* 其次，使用Spark在Hive中建表，随后从HDFS批量向Hive导入数据。
* 接着，使用Spark从Hive中读取、处理、分析数据，并将结果存入结果表中。
* 再次，使用Spark从Hive中读取数据，将数据存入服务器的MySQL中。
* 最后，通过Web系统，实现数据可视化。

# 2. 数据采集模块开发

使用Python3作为编写网络爬虫的语言，主要使用了lxml、json、request等工具。

硕士研究生招生数据的采集分为三部分：

## 2.1爬取省市代码和名称，以及学科包含关系

直接用post请求爬取<https://yz.chsi.com.cn/zsml/pages/getSs.jsp>获得省市代码和名称，并存到已经建好表的mysql内，供Web前台使用。

经分析，网站<https://yz.chsi.com.cn/zyk/>的专业代码有三层关系，例如：网页li标签id为'1008'的'工学'代码为'08'，点击该标签会以属性method=subCategoryMl&key=1008使用post请求<https://yz.chsi.com.cn/zyk/specialityCategory.do> 获得属于工学的下一级学科，其中网页li标签id为'100812'的'计算机科学与技术'代码为'0812'，点击该标签会以属性method=subCategoryXk&key=100812使用post请求同上链接获得再下一级专业，其中有一列为'计算机系统结构'代码为'081201'，于是发现了08>0812>081201这样的包含关系。

由网页很简单得获得了两位数专业代码与名称对应关系:

xkjson={'01': '哲学','02': '经济学','03': '法学','04': '教育学','05': '文学','06': '历史学','07': '理学','08': '工学','09': '农学','10': '医学','11': '军事学','12': '管理学','13': '艺术学'}

再遍历爬取所有专业，并将其包含关系转换成字典，并以json格式保存，格式见图2.1，这里的叶子节点的value参数置0是为后续数据分析（5.4节）做准备.由于数据量较小，所以使用移动指针标记免费代理列表，关于免费代理的其他使用问题将在第二步讲解。

## 2.2爬取每个省市的学校信息及该学校所有考试科目id

经综合考虑后，选择探究研招网考试目录的含表单网页<https://yz.chsi.com.cn/zsml/zyfx_search.jsp>，其中有六个表单项，学科类别为必选项，其他至少填写一项。以post请求爬取<https://yz.chsi.com.cn/zsml/pages/getZy.jsp>获得学科类别列表，这里的学科类别为四位数学科，在选择省市作为其他一项，这里的省市数据在第一步里已经拿到了，所以下一步应该是交叉遍历省市列表与四位数学科类别列表进行查询。每一个查询后，会显示符合要求的学校列表，数据多则需要翻页，以及其是否有研究生院、博士点，是否自主划线，再次点击学校名后会显示这个学



图2.1

校的这门学科的考试科目(研究方向)列表，数据多则需要翻页，但这里没有具体的初试考试信息，需要点击每个考试科目(研究方向)的“查看”才能看到。由于整个过程涉及四种网页，所以打算先获取各考试科目(研究方向)“查看”超链接中的id属性,例如：<https://yz.chsi.com.cn/zsml/kskm.jsp?id=1000121023010102011> 直接访问即可得到所有这个考试科目(研究方向)的信息，将这部分放在第三步，这里的第二步将是获取学校信息及其所有考试科目的id列表。

因为访问的页面数实在太多，经测试要每个IP访问间隔大于3秒才能保证不被封，也不想浪费时间去使用大量的延迟控制访问频率，于是在网上找到了14个免费代理IP，提高采集效率。以下是我的使用方法：

在代码中，编写省市类、学校类。遍历省市列表，为每一个省市创建省市对象，省市对象执行方法遍历四位数学科类别列表，每查询一次即可得到很多学校。如果省市对象中没有创建过该学校，则为每个学校创建对象存到省市对象中并使用此对象；否则取出该学校对象并使用此对象。每个学校对象都执行获取该学校该学科访问包含考试科目列表网页的方法，由于这些访问是异步执行，所以都只用一个IP一定会因频率过高被网站封杀，但只有14个代理IP，使用循环按顺序或者随机获取代理IP都有可能因固定的延迟分配不均而突然暴毙。于是我经独立思考使用了这样的方法：创建了简易的IP代理池，结构是队列；为每个学校从代理池队首分配一个代理IP，这个学校对象在使用这个代理IP时的每个网页访问延迟时间设置为固定的3秒，连续五次访问失败则重新获取新代理IP，在访问完包含该学校的该学科考试科目列表的所有网页后，归还IP到代理池的队尾。这样既避免了代理IP惨遭封杀，又使用巧用异步多线程大大提高了数据采集效率，在第二步仅仅爬取了7小时左右。

拿到数据后，以省市代码作为json文件名，json结构为如图2.2所示。



图2.2

数据含义：

schdm：学校代码

schmc：学校名称

yjsy：是否含研究生院

zzhxyx：是否为自主划线院校

bsd：是否含博士点

kskm\_id\_list：考试科目id列表

爬取文件展示如图2.3，共31个省市，848所高校。

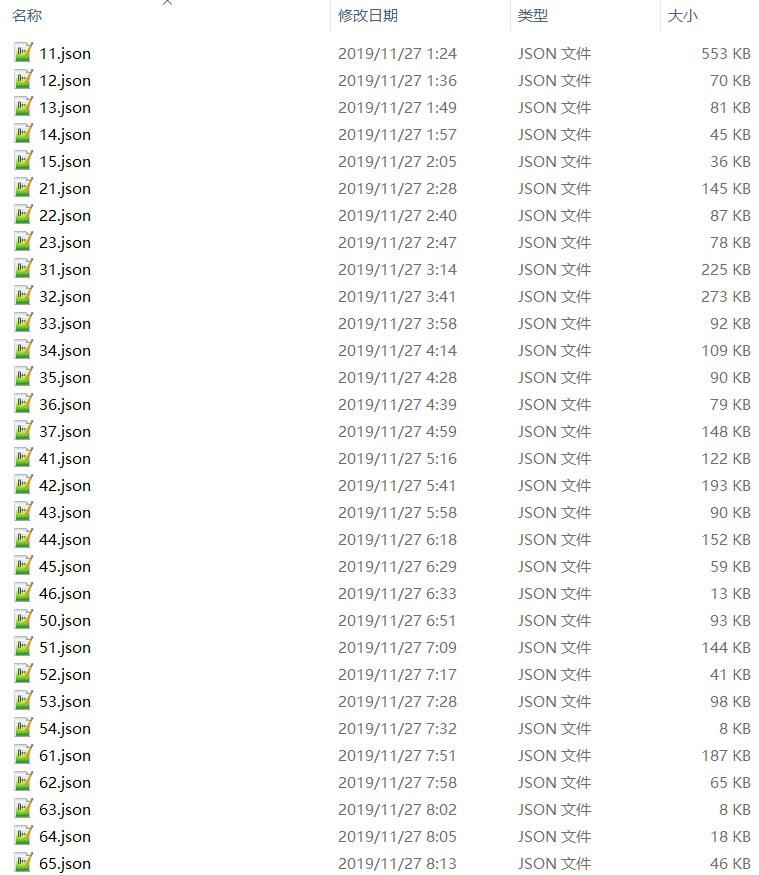


图2.3

## 2.3爬取所有初试考试科目信息

遍历读取第二步得到的学校数据，将其中的kskm\_id\_list获取，并遍历id以get请求https://yz.chsi.com.cn/zsml/kskm.jsp?id=访问网页获取数据。图2.4为其中一张网页，上半部分为考试科目基本信息，其中有可能会有空缺，但也有“跨专业”是个超链接，能获得详情信息，如图2.5；下半部分为数个考试范围，有的只有三门考试但也是以政治、外语、业务课一和业务课二形式展示，业务课二有可能展示为“无”，也应获取到。



图2.4



图2.5

如同第二步的免费代理使用方法，这里也使用了该方法爬取考试科目详情信息，数据格式如图2.6。



图2.6

数据含义：

kskmid：考试科目id（解析考试科目id后，有以下分别对应的规律：

10001 21 023 010104 00 1

学校代码 未知 院系所 专业(六位学科代码) 研究方向 全日制）

schdm：学校代码

ksfs：考试方式

yss：院系所

kzy：跨专业

yjxkdm：专业，研究学科代码，即六位数学科

xxfs：学习方式，全日制1，非全日制2

yjfx：研究方向

zdls：指导老师(可能没有老师名)

nzrs：拟招人数(数据处理较复杂，有些是本专业所有研究方向只招这些人，

有些是本研究方向就招这些人，而且不同学校的书写格式不同，

专业：3(不含推免) 、专业：1(含推免) 、研究方向：1(不含推免)等不规则文本，暂时没有时间处理)

bz：备注

ksfw\_list：考试范围列表

考试范围列表：

zz：政治

wy：外语

ywk1：业务课一

ywk2：业务课二

存储文件与第二步相同，以省市代码保存到另一文件夹下，共148441条考试科目信息，第三步爬取时间约为8小时。

# 3. 数据预处理模块开发

由于数据预处理都在数据采集时就已经完成了，不需再处理，在这里只叙述如何将数据从服务器本地上传到HDFS。

数据存放在两个文件夹中，它们分别是存放学校数据的“school”文件夹与存放考试科目的“kskm”文件夹，其中的每个文件都是由其所在省市的代码命名，在这里编写Shell脚本将数据从本地批量上传至HDFS，脚本如图3.1。

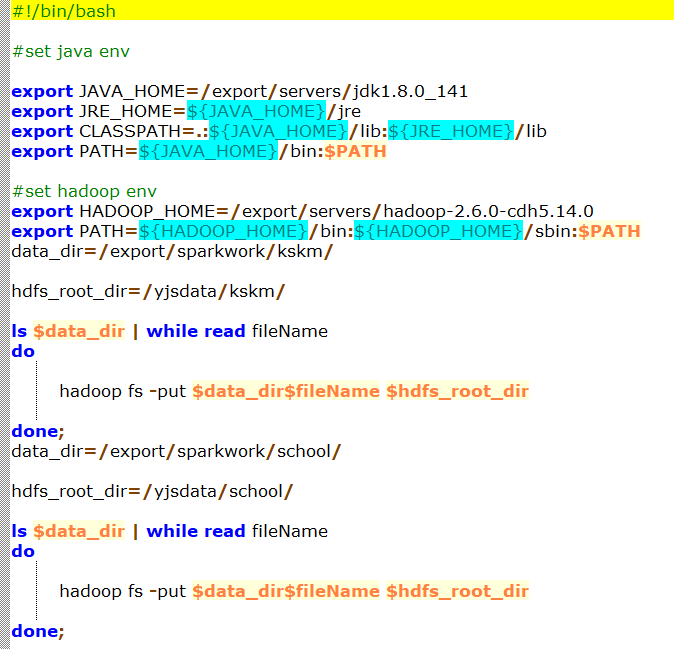


图3.1

# 4. 数据仓库开发

传统的Hive使用MapReduce作为计算引擎，现使用Spark作为计算引擎较传统的MapReduce更适用于迭代运算，要同时使用Hive与Spark则需要将Hive的计算引擎改为Spark。

在此，本系统统一使用Spark的Python3 API，需要使用SparkSQL在Hive创建两张原始表：

表4.1 school\_origin

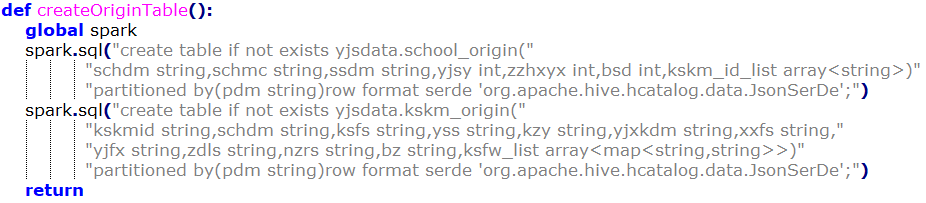
|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **schdm** | **学校代码** | **zzhxyx** | **是否为自主划线院校** |
| **schmc** | **学校名称** | **bsd** | **是否有博士点** |
| **ssdm** | **省市代码** | **kskm\_id\_list** | **考试科目列表** |
| **yjsy** | **是否有研究生院** |  |  |

school文件夹的数据按省市代码分区，将其导入如表4.1所示表。

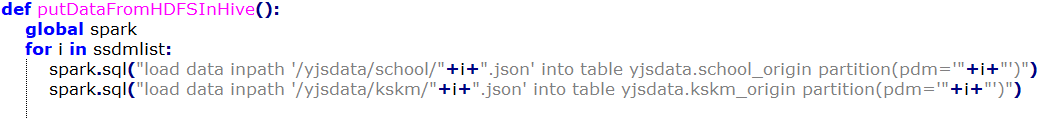
表4.2 kskm\_origin

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **kskmid** | **考试科目id** | **xxfs** | **学习方式** |
| **schdm** | **学校代码** | **yjfx** | **研究方向** |
| **ksfs** | **考试方式** | **zdls** | **考试科目列表** |
| **yss** | **院系所** | **nzrs** | **拟招人数** |
| **kzy** | **跨专业** | **bz** | **备注** |
| **yjxkdm** | **研究学科代码(六位数学科代码)** | **fsfw\_list** | **考试范围列表** |

kskm文件夹的数据按省市代码分区，将其导入如表4.2所示表。



使用Hive里的jar包：org.apache.hive.hcatalog.data.JsonSerDe，并将其导入Spark的jar包目录，即可直接将json文件使用SparkSQL导入Hive表。



# 5. 数据分析模块开发

## 5.1模块一

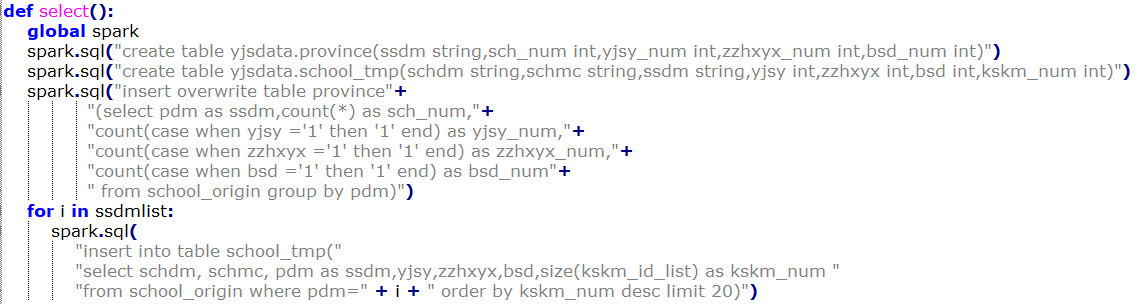
统计每个省市的学校个数、研究生院个数、自主划线院校个数与博士点个数（如表5.1），以及每个省市的考试科目最多的前二十院校信息（如表5.2）。

表5.1 province

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **ssdm** | **省市代码** | **zzhxyx\_num** | **自主划线院校个数** |
| **sch\_num** | **学校名称** | **bsd\_num** | **含博士点学校个数** |
| **yjsy\_num** | **含研究生院学校个数** |  |  |

表5.2 school\_tmp

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **schdm** | **学校代码** | **zzhxyx** | **是否为自主划线院校** |
| **schmc** | **学校名称** | **bsd** | **是否有博士点** |
| **ssdm** | **省市代码** | **kskm\_id\_list** | **考试科目列表** |
| **yjsy** | **是否有研究生院** |  |  |

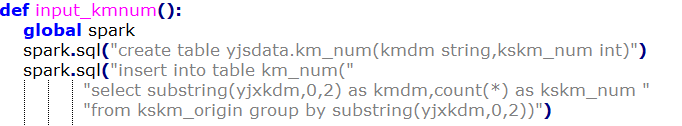


## 5.2模块二

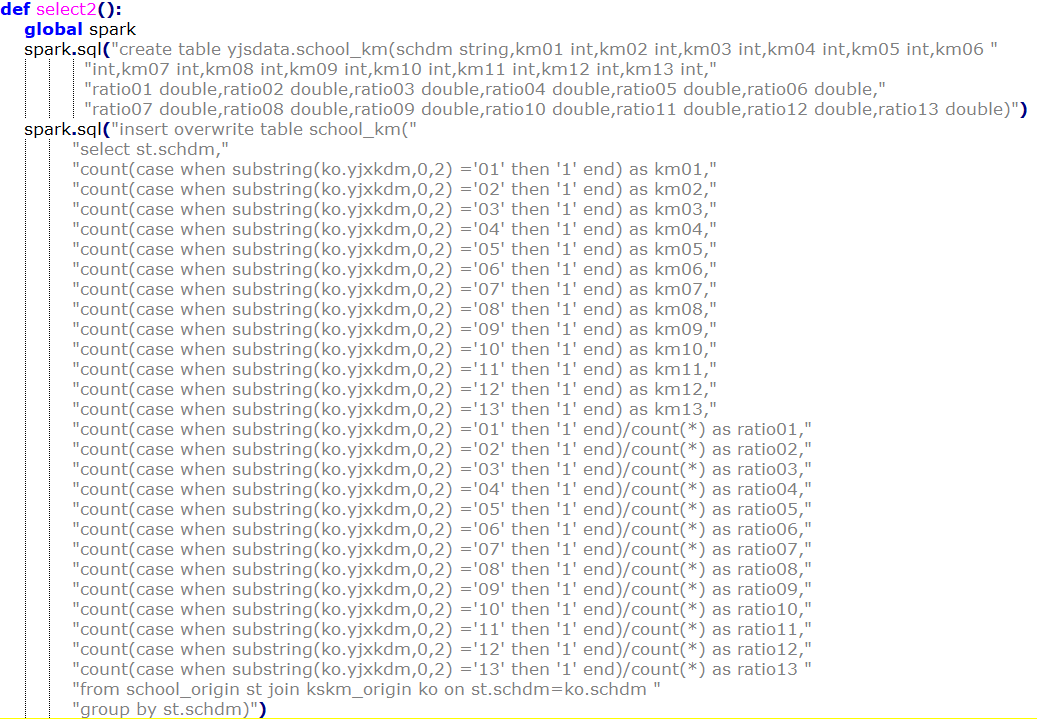
统计各个两位数学科代码（如08工学）在全国范围的考试科目数（如表5.3），与该学科考试科目数占比前十的学校信息。

表5.3 km\_num

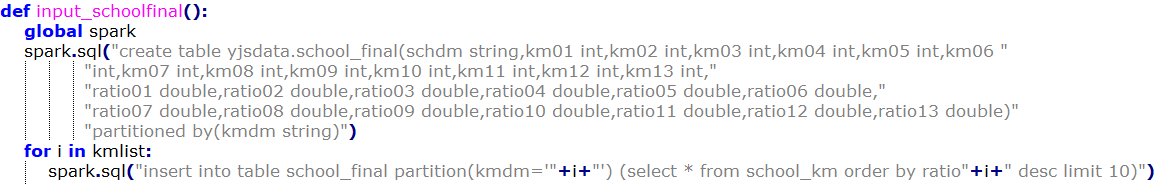
|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **kmdm** | **科目代码（两位数）** | **kskm\_num** | **考试科目数** |



中间表"school\_km"存储所有院校各个学科考试科目数与其所占比例，如下图所示。



从中间表取每个学科的占比前十院校的信息到"school\_final"表（如下图），并按照科目代码分区。该表不是最终导入数据库的最终表，将在数据导出时对此表进行加工。



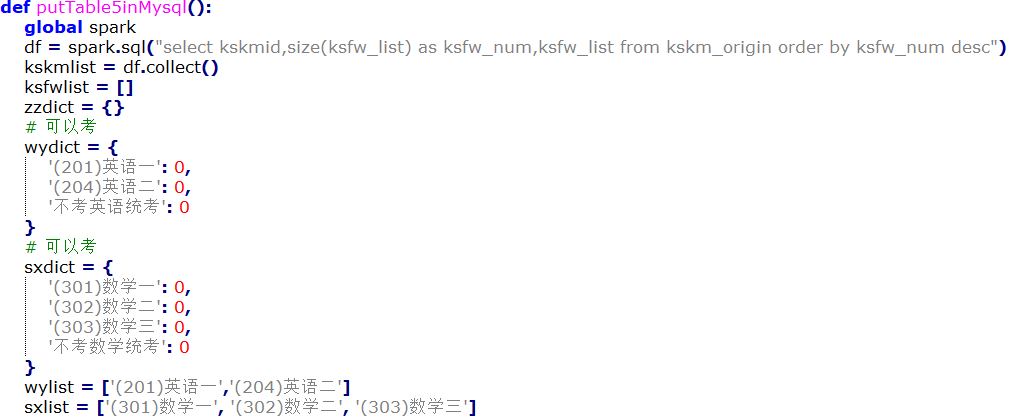
## 5.3模块三

关于考试范围的分类统计：

1. 对政治考试分类统计，最终得到的是“允许考XXX的考试科目数”。
2. 对外语考试分类统计，由于考试科目种类太多只统计大多数人关注的‘(201)英语一’与‘(204)英语二’，最终得到的是“允许考XXX的考试科目数”，以及不考上述科目的数量。
3. 对数学考试分类统计，只统计大多数人关注的‘(301)数学一’、‘(302)数学二’与‘(303)数学三’，最终得到的是“允许考XXX的考试科目数”，以及不考上述科目的数量。

该模块对考试范围下手，每个考试科目的考试范围列表ksfw\_list由若干条考试范围构成，每条考试范围由四门考试组成（数据结构详见2.3节），由于原始表是由json文件直接导入，使用SparkSQL语句直接读取分析array<map<string,string>>型数据并不方便，所以采取以下方式处理数据。

先将数据读入内存，并将统计初值置零。



分类统计每个考试科目的政治、外语、数学。



最后将统计结果导入mysql数据库表5.4。

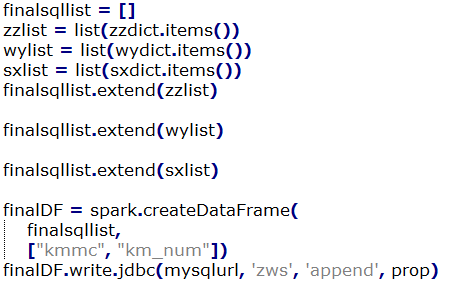


表5.4 zws表(MySQL)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **kmmc** | **科目名称** | **km\_num** | **含该科目的考试科目数** |

## 5.4模块四

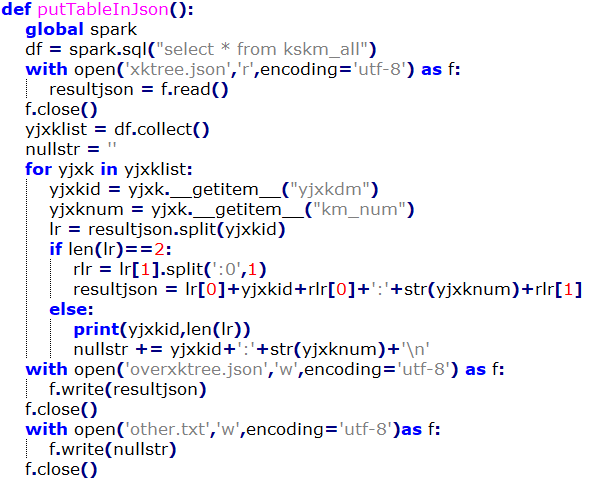
统计六位数学科的考试科目数（如表5.5），并填入2.1节的学科包含关系json文件xktree.json的value值中。

表5.5 kskm\_all

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **yjxkdm** | **研究学科代码（六位数学科）** | **km\_num** | **考试科目数** |



由于json文件的学科代码都是统一标准的代码，很多院校自创学科与代码，所以有一些无法填入的学科代码，只将符合官方统一标准的yjxkdm的value导入json文件。



# 6. 数据导出模块开发

通过Spark连接服务器的MySQL，直接将DataFrame对应列名插入建好的MySQL数据表中。表5.1、表5.2与表5.3将直接导入MySQL，第5章模块二中"school\_final"表的schdm（学校代码）将与"school\_origin"表（如表4.1）的schdm（学校代码）自然连接，在

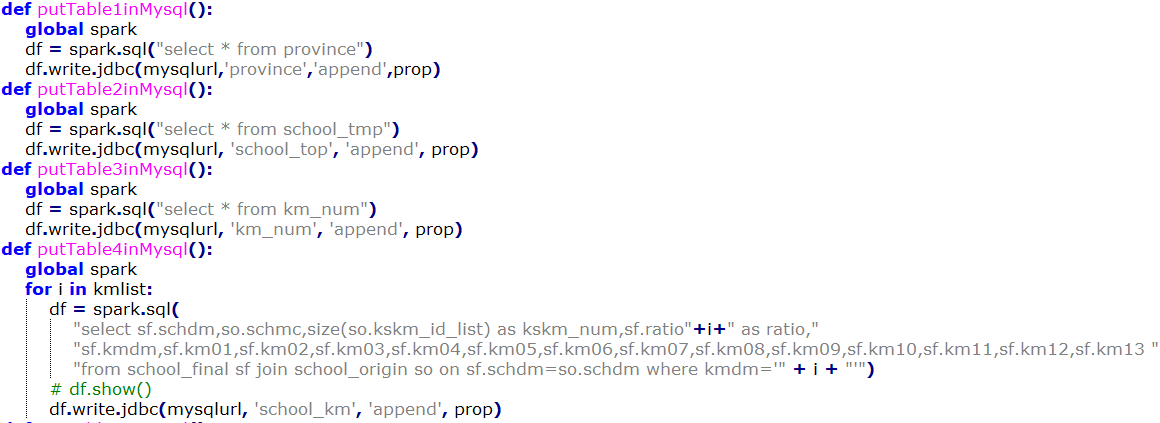


表6.1 school\_km表(MySQL)

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **描述** | **字段** | **描述** |
| **schdm** | **学校代码** | **km05** | **文学考试科目数** |
| **schmc** | **学校名称** | **km06** | **历史学考试科目数** |
| **kskm\_num** | **考试科目数** | **km07** | **理学考试科目数** |
| **ratio** | **优势学科考试科目占有率** | **km08** | **工学考试科目数** |
| **kmdm** | **优势科目代码(两位数学科)** | **km09** | **农学考试科目数** |
| **km01** | **哲学考试科目数** | **km10** | **医学考试科目数** |
| **km02** | **经济学考试科目数** | **km11** | **军事学考试科目数** |
| **km03** | **法学考试科目数** | **km12** | **管理学考试科目数** |
| **km04** | **教育学考试科目数** | **km13** | **艺术学考试科目数** |

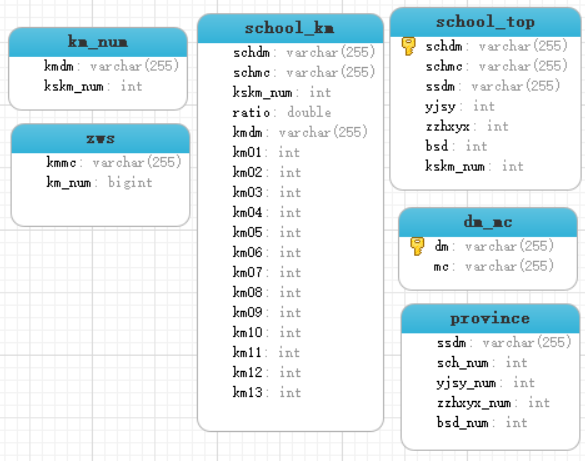


图6.1 MySQL数据库

“dm\_mc”是在2.1节存储省市代码dm与省市名称mc对应关系的数据表，“school\_km”见表6.1，“zws”见表5.4，“school\_top”见表5.2，“km\_num”见表5.3，“province”见表5.1。

# 7. 报表展示模块开发

硕士研究生招生数据分析展示是一个纯粹的Java EE项目，使用SSM框架搭建，工程资源结构如下图：

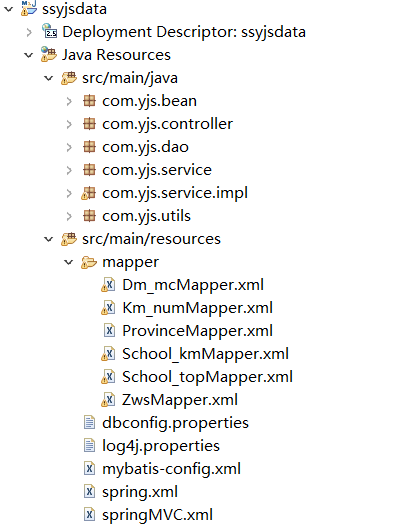
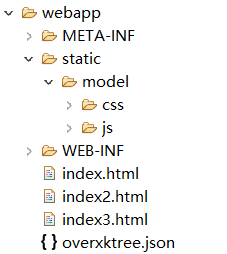
 

图7.1 工程资源结构

数据层连接图6.1所示数据库，业务逻辑层从数据层取数据并加工，控制层则接受来自前端页面异步请求并返回数据。

三模块大致图：

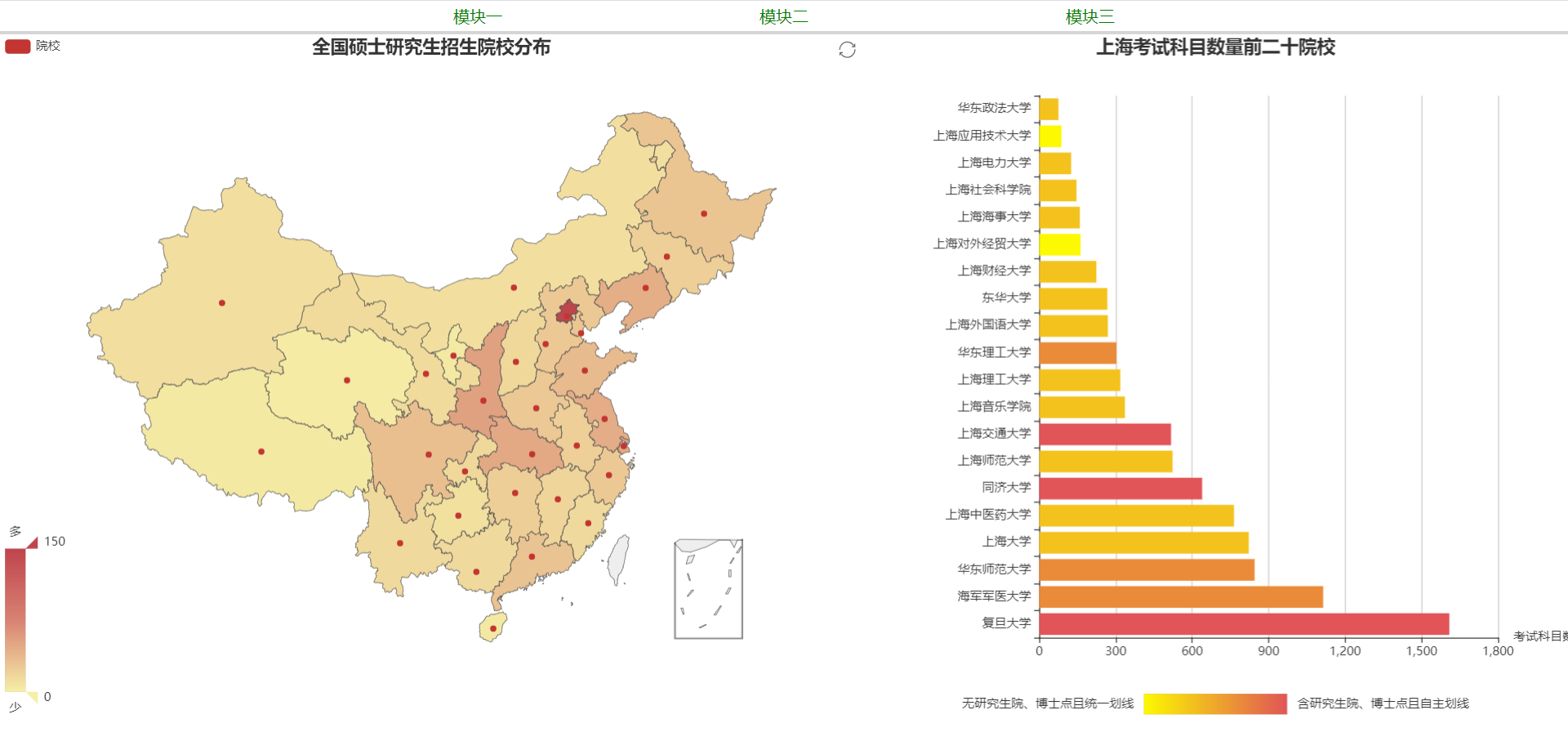


图7.2 index.html模块一

index.html即模块一对应第5章的模块一：鼠标悬停在左图任意省市可查看该城市相关信息，点击任意省市即可将右图换成对应省市的数据视图，右图鼠标悬停在任意学校可查看学校相关信息。该模块全部数据都是从后台服务器异步获取，点击事件触发前端发送相应异步请求，获取到精确数据，不至于将全部数据一次获取完造成页面请求时间过长。

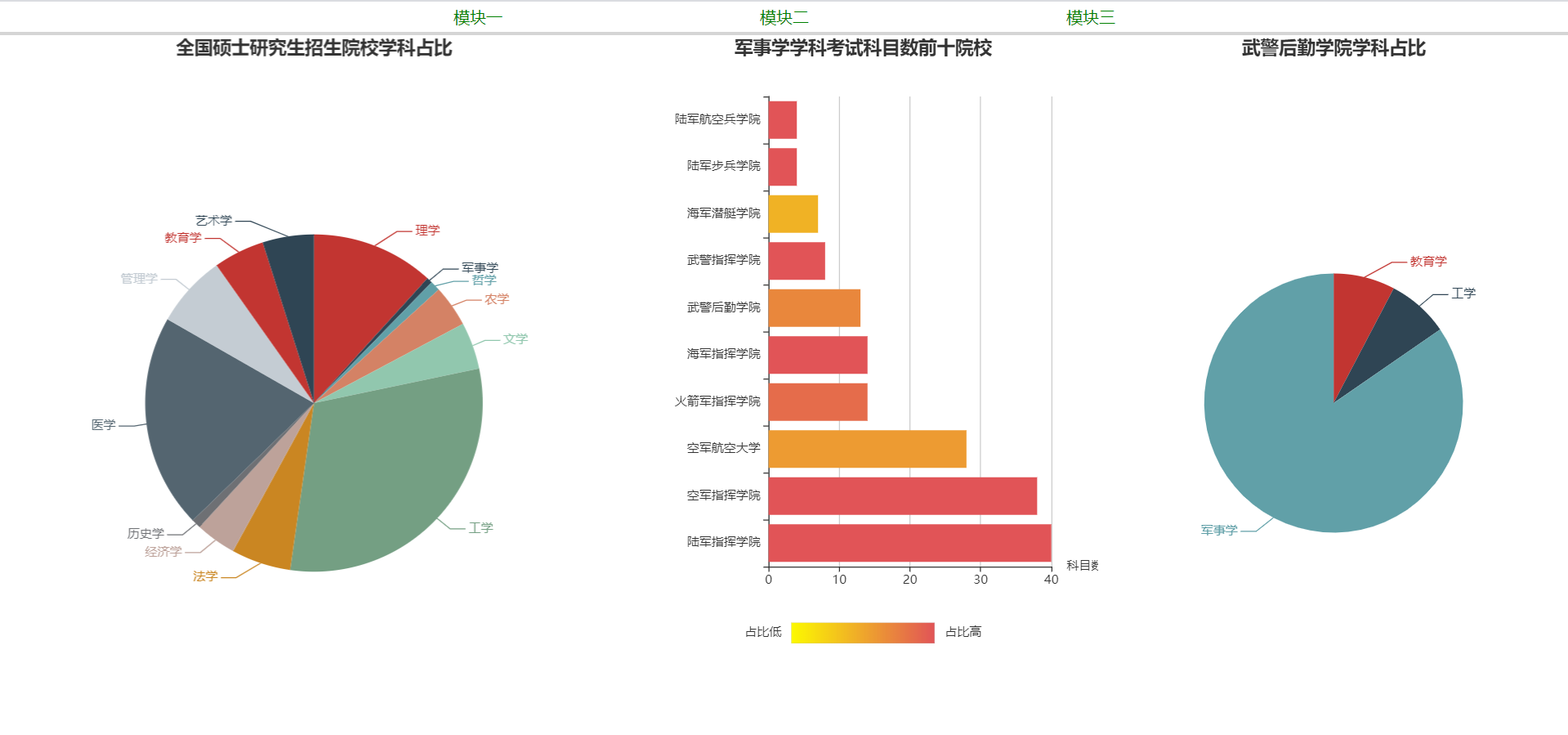
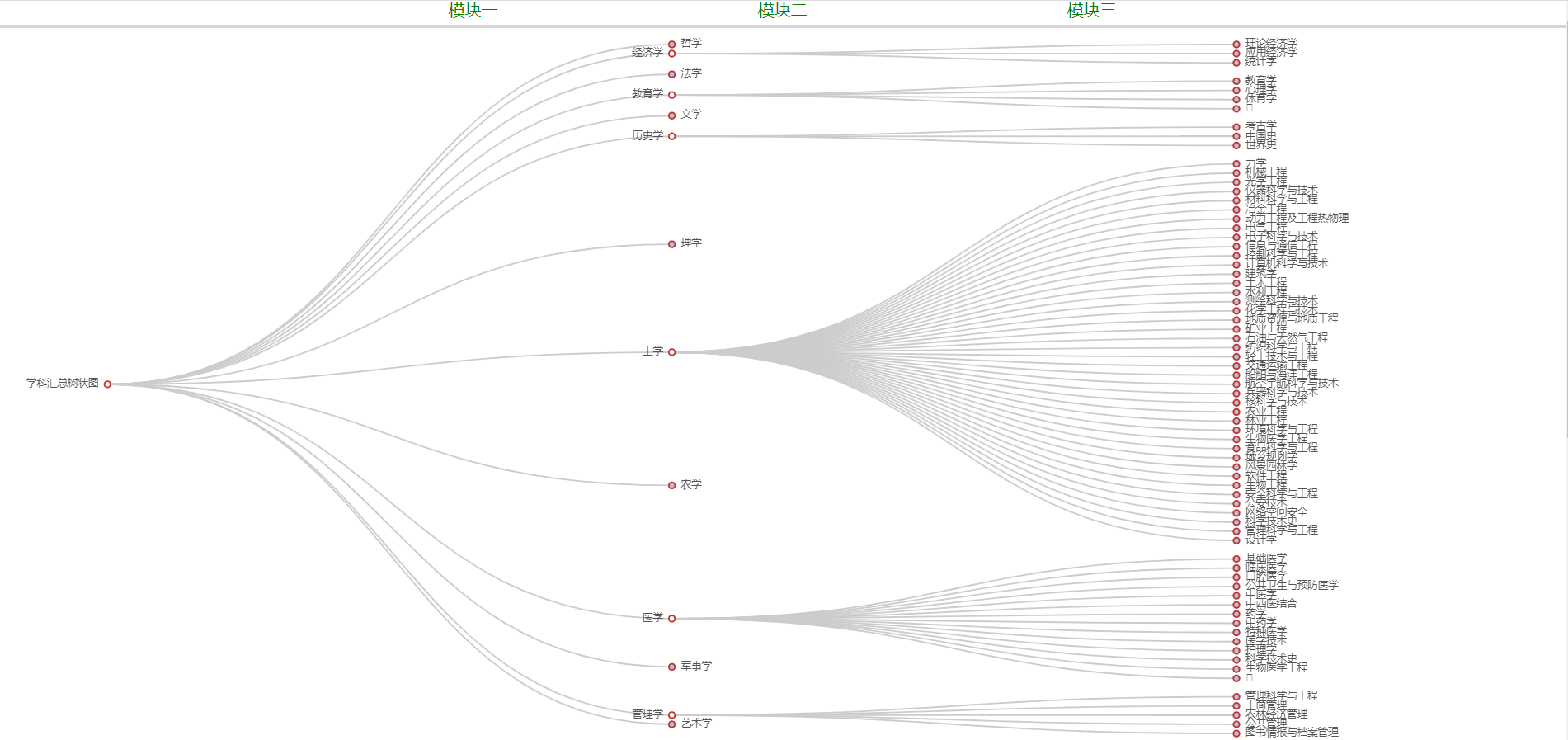


图7.3 index2.html模块二

index2.html即模块二对应第5章的模块二：鼠标悬停在左图任意学科即可查看该学科考试科目数量即所占比例，点击后可将中图替换成该学科考试科目占比前十院校数据视图，将右图换为中图里考试科目最多的学校的各学科在其学校所占比例视图；鼠标悬停在中图可以查看院校相关信息，点击院校则也会将右图替换成该院校相关数据视图。该模块全部数据都是从后台服务器异步获取，点击事件触发前端发送相应异步请求，获取到精确数据，不至于将全部数据一次获取完造成页面请求时间过长。



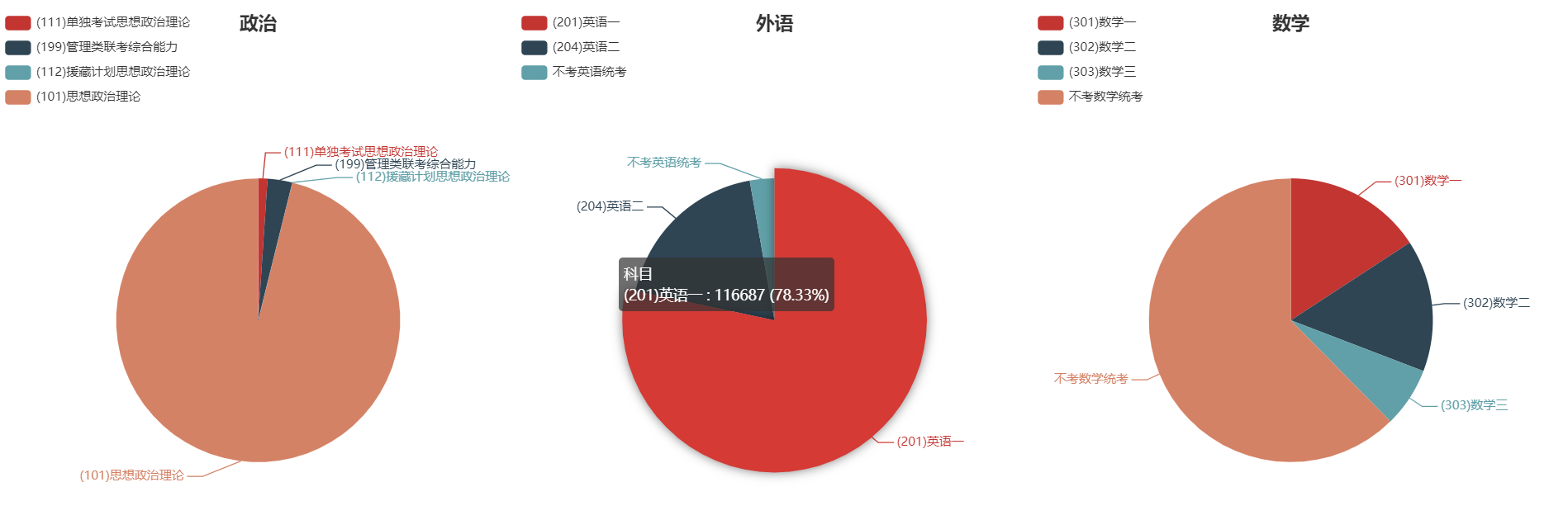


图7.4 index3.html模块三

index3.html模块三对应第5章的模块三与模块四，上图为学科包含关系视图，树状图叶子节点鼠标悬停可见该学科考试科目数量，其他节点可以使用折叠与展开操作；下面三图则是政治、外语和数学这三个最受关注的考试科目，展示了其各自占比情况。由于数据量过小以及源文件为json格式，该模块全部数据来源于静态文件，所以本模块没有设置点击事件取异步请求后台服务器。

最终将打包好的Java EE项目与数据库文件上传至服务器并部署，即可从外网访问本系统展示平台。

# 8. 系统特色及总结

特色：

在数据采集方面，使用Python3作为编程语言，构建代理池爬取数据，在反爬措施较强的网站爬取大量数据时，爬取速度较之前翻了近十倍。

在数据预处理方面，使用Shell脚本将多文件导入HDFS。

在数据分析与导入方面，使用Spark作为计算引擎的硕士研究生招生数据分析系统较其他使用MapReduce作为计算引擎的系统更加趋于主流，计算速度更快，选用Python3作为编程语言，运用SparkSQL分析和处理数据。

在数据展示方面，使用SSM框架完成Java EE项目，大量使用Echarts作为前端数据展示视图，大部分数据从数据库获取，为节省网络开销，只从数据库异步请求获取用户点击部分的相关数据。

总结：

起初想到这个题目是因为需要指定自己下一年的考研计划，在选题时无意蹦出了这个想法，找到了研招网这个最为官方的硕士研究生招生网站。

好在今年暑假自学了Spark编程，用起来也比MapReduce更容易上手，结合暑假参加山东省软件设计大赛的比赛经验，花了一天时间设计出了本系统的设计方案，并在之后的制作中陆续修改设计方案。

前期在制作本系统时，就发现数据只有官网的招生计划数据，没有实际的具体招生数据，而且只有2020年的数据，所以无法分析出走势以及制作出较为可靠的推荐系统，这也是本系统的不足之处。在考试范围数据中，很多数据没有在本系统的数据分析里用到，相信这些数据一定有更多的可挖掘价值，小生不才，在有限的时间里只能想到以及做到这三大模块了，包括上课时间一共花了十天完成了本系统，对自己来说制作本系统也是一种挑战，有难度才能成长。