

实验十二 大数据应用实例——互联网金融大数据平台

(一) 实验目的

1. 理解大数据可视化相关图例；
2. 会使用可视化工具编写大数据可视化模块；
3. 会执行该程序；
4. 自行分析执行过程。

(三) 实验环境

1. 大数据分析实验平台（FSDP）；
2. Python 2.7；
3. django 1.11.10；
4. Echart 4.0。

(二) 实验要求

1. 基于Python的Django框架和Vue构建可视化平台架构；
2. 基于Echart实现各种常用的可视化图例，包括：直方图、线图、饼图、地图等。

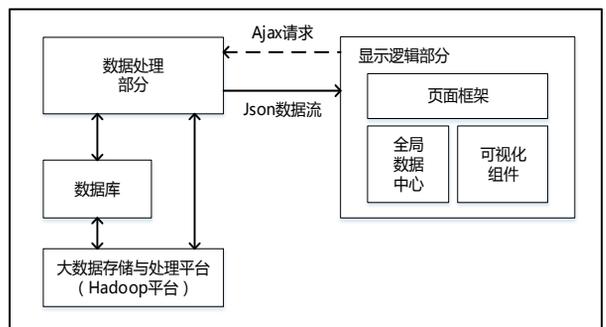
(四) 实验步骤

1. 互联网金融大数据平台简介；
2. 使用Django框架构建数据操作子平台；
3. 使用Echart实现大数据可视化。

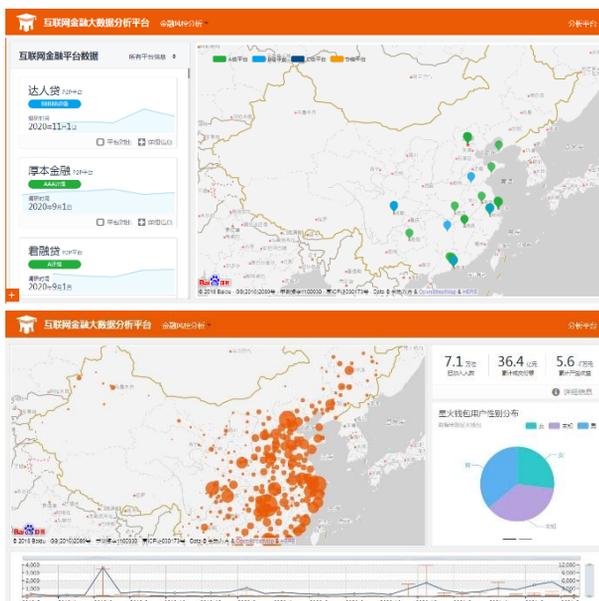
1、互联网金融大数据平台简介

平台实现了互联网金融大数据的数据分析以及风控管理，包括：基本数据的分析及显示，金融平台盈利、产品和用户相关数据的分析及显示、风控分析及其结果显示。同时，还实现互联网金融用户分析结果显示以及用户渠道分析及结果显示：用户特性的分析显示，包括性别、地域、背景等；完成用户渠道效果分析。

该平台可视化实现框架将数据操作代码与可视化代码分离。数据操作子平台由Python的Django框架构建仅和数据库交互，提取数据；可视化子平台采用Vue与Echarts，在浏览器上构建可视化界面；并通过AJAX方式获取服务器端数据予以显示。



互联网金融大数据可视化平台架构图



2、使用Django框架构建数据操作子平台

使用Django框架构建数据操作子平台，主要负责数据的读取与数据预处理。

- (1) 使用django-admin命令新建一个django项目。

```
[root@fsmanager hadoop_example]# django-admin.py startproject project_name
```

- (2) 进入项目目录，然后运行该目录下的manage.py，执行startapp子命令。（下面的命令已经在项目目录下则不需要 cd project_name。）

```
[root@fsmanager hadoop_example]# cd project_name
[root@fsmanager hadoop_example]# python manage.py startapp app_name
```

实验十二 大数据应用实例——互联网金融大数据平台

▶ (3) 创建数据库表 或 更改数据库表或字段。

```
[root@fsmanager hadoop_example]# python
manage.py makemigrations
[root@fsmanager hadoop_example]# python
manage.py migrate
```

▶ (4) 运行开发服务器。

```
[root@fsmanager hadoop_example]# python
manage.py runserver
```

▶ 3、使用Echart实现大数据可视化

▶ (1) 视化子平台采用Vue与Echarts，在浏览器上构建可视化界面，其中Vue文件架构如下。

```
<div id="app">
  <div v-html="message"></div>
</div>
<script>
  new Vue({
    el: '#app',
    data: {
      message: '<h1>基本架构</h1>'
    }
  })
</script>
```

▶ (2) Echarts基本形式：Echarts的图形化呈现主要是通过配置方法来实现的(setOption)，首先进行图形配置，然后对图形标签进行初始化，最后把配置通过方法(setOption)赋值到初始化图形中。

```
<script type="text/javascript">
//指定图标的配置和数据
var option = {
  title:{ text:'Echarts 数据统计' },
  tooltip: {},
  legend: { data:['用户来源'] },
  xAxis: {
    data: ["Android", "IOS", "PC", "Ohter"]
  },
  yAxis: {},
  series: [{
    name: '访问量',
    type: 'line',
    data: [500, 200, 360, 100]
  }]
};
//初始化echarts实例
var myChart = echarts.init(
  document.getElementById('chartmain'));
```

```
//使用制定的配置项和数据显示图表
myChart.setOption(option);
</script>
```

▶ (3) 基本的直方图：主要修改series属性中的type。

```
series: [{
  name: '现金流',
  type: 'bar',
  stack: '总量',
  barWidth: '50%'
}]
```



▶ (4) 基本的饼图：主要修改series属性中的type。

```
series: [{
  name: '用户性别分布',
  type: 'pie',
  radius: '55%',
  center: ['50%', '60%'],
  data: this.values,
}]
```

