

## 《软件实践》课程实验报告

### 暑期学校实验项目：高考志愿填报助手

|   |                     |    |      |     |        |    |          |
|---|---------------------|----|------|-----|--------|----|----------|
| 小组名称  | 组 4:知识图谱构建 B 组      |    |      |     |        |    |          |
| 姓 名   | 林欣                  | 专业 | 人工智能 | 班级  | 091182 | 学号 | 09118238 |
| 实验时间  | 2020.8.31-2020.9.23 |    | 指导教师 | 孔祥龙 |        | 成绩 |          |
| <b>一、实验背景和目的</b> <p>今年的暑期学校原定于 7 月开课，考虑到高考志愿填报在即，计划借这次软件实践课程实验的机会完成一个高考志愿填报助手，为学弟学妹谋福祉。虽然最终暑期学校推迟到了 9 月份，高考录取已经告一段落，但是高考年年都有，志愿填报助手依旧有着极大的价值，遂沿袭了最初的设定。这次的课程作业主要是基于 Django 框架，在多个小组分工协作下最终实现基于 Web 的高考志愿填报助手。</p>   |                     |    |      |     |        |    |          |
| <b>二、小组任务和个人任务</b> <p>我们小组负责以省份为中心的知识图谱的搭建及其可视化展示。我们小组选择以 neo4j 图数据库为核心，总体大致分为前端和后端两部分各自实现。我参与的是前端部分，负责省份知识图谱的网页展示，在小组网页的展示中起承上启下的作用。在组员完成知识图谱构建的基础上，我负责进一步美化知识图谱的展示，并结合第九组所选择的 UI 设计模板，美化该网页展示，与整体网站风格保持一致，也为之后分数线折线图的展示作准备。</p>   |                     |    |      |     |        |    |          |
| <b>三、个人任务需求分析</b> <p>1、知识图谱可视化</p> <p>我们小组选择了使用 Cytoscape.js 来可视化知识图谱。基本的知识图谱已经由组员构建完成，我需要做的是将知识图谱草图美化并展示在网页中。</p> <p>具体的需求包括：</p> <ul style="list-style-type: none"><li>(1) 将知识图谱居中显示；</li><li>(2) 设置合适的缩放比例及范围，保证图谱节点清晰可见并防止鼠标误触导致图谱跑位；</li><li>(3) 选择合适的节点及边的样式，不同类型的节点之间有所区别并保持美观；</li><li>(4) 设置适当的背景画布大小和颜色；</li><li>(5) 设计图谱布局，使得院校节点和年份科目节点规律地错开分布；</li></ul> <p>2、网页前端设计</p> <p>考虑到网站风格的整体性和统一性，我们采用了和第九组相同的 UI 设计模板，在此基础上完善省份知识图谱网页的设计。</p> <p>具体需求包括：</p> <ul style="list-style-type: none"><li>(1) 导入所需的静态文件模板，包括 css 文件、js 文件、图片和字体等；</li><li>(2) 参考第九组网页风格并设计省份知识图谱展示的网页。</li></ul> |                     |    |      |     |        |    |          |

## 《软件实践》课程实验报告

### 四、实验过程（需附上关键代码及相关说明）

#### 1. 网页前端设计

为保持整体风格一致，我们小组和第九组对接拿到了一个 htmlTest 文件夹，里面包括他们设计好的网页文件以及相应静态文件模板。在阅读了 index.html 和 about.html 之后，我对网站的整个格式有了基本的认识，分别加入了网站的顶部菜单和底部菜单，参考两个文件分别加入了网站的顶部菜单和底部菜单，也就是<section class="w3l-top-menu-1">，<section class="w3l-bootstrap-header">和<section class="w3l-footer-29-main">三个部分。

接着我尝试导入网页的 css 文件，参考了 index.html 里的代码（如下）：

```
<link rel="stylesheet" href="assets/css/style-starter.css">
```

我把含有所有静态文件的 assets 文件夹拷贝至 templates 文件夹目录下，然后运行代码，结果发现网页并没有发生改变，检查运行结果后得知导入 css 文件的路径是错误的。在多次的自我尝试仍以失败告终之后，我意识到问题在于我没有想清楚 Django 框架里路径的关系。于是我上网搜索了如何在 Django 框架里引入 css 文件，了解到：要先在 settings.py 文件里设置静态文件路径，并将静态文件存在对应的路径下。重新修改代码为：

```
<link rel="stylesheet" type="text/css" href="{% static 'css/style-starter.css'%}">
```

再次运行代码后，发现依旧没有成功导入 css 文件。于是在文件开头增加代码：

```
{% load static %}
```

问题解决，成功地导入了静态文件，初步形成了网站大体框架。

#### 2. 知识图谱可视化

我们小组选择了使用 Cytoscape.js 来可视化知识图谱，通过查询 Cytoscape.js 的属性可实现对知识图谱可视化的一系列美化操作。

(1) 首先考虑将知识图谱居中显示。此处修改涉及到知识图谱的整体布局，于是对应修改位置如下：

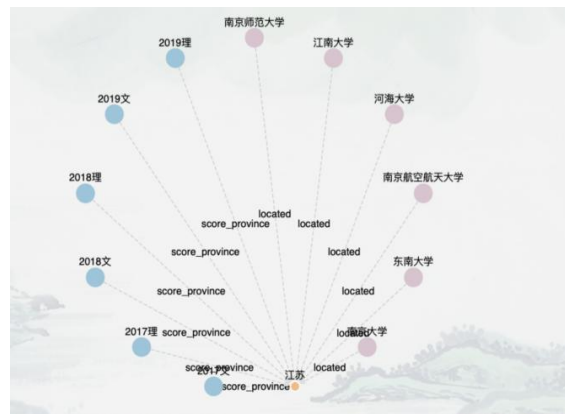
```
<style type="text/css">
    #cy { /* 在此处增加代码 */ }
</style>
```

在查阅相关资料后发现只需要增加一行代码：margin:0 auto; 同时，在此处还可以设置背景画布大小和颜色。考虑到美观性，打算将背景画布设置有半透明效果，于是改用 RGBA 来表示颜色。经过多次调整，最终增加代码如下：

```
height: 500px; width: 1100px; background-color: rgba(242,242,242,0.85);
```

(2) 接下来是需要设计图谱布局，使得院校节点和年份科目节点规律地错开分布。修改布局主要是针对 Cytoscape 中的 layout 属性。在这里，我先后尝试了均分布局、圆形布局和同心圆布局。三种布局区别主要在于父节点的位置以及父节点和子节点的相对位置，具体如下所示：

## 《软件实践》课程实验报告



由于在修改此处代码的同时，小组成员也在不断完善知识图谱的数据库，所以第一个布局和后两个布局的数据略有差别，此处特说明。

以上三种布局分别对应'grid'，'circle'，'concentric' 三种不同的布局名称，并可以通过修改其他布局属性来进一步修改布局的细节。由于最终呈现的节点还要增加一本线的信息，考虑到要将三种不同的节点错落有致地展示出来，我选择了视觉效果最直观的同心圆布局，代码如下：

```
cytoscape({
  layout: {
    name: 'concentric', /* 在此处增加了布局属性细节 */
    fit: true,
    padding: 30,
    avoidOverlap: true,
    nodeDimensionsIncludeLabels: false,
    spacingFactor: undefined,
    startAngle: 3 / 2 * Math.PI,
    sweep: undefined,
    clockwise: true,
    animate: false,
    animationDuration: 500,
    animationEasing: undefined,
    animateFilter: function ( node, i ) { return true; },
    transform: function (node, position ) { return position; },
    minNodeSpacing: 50,
  },
});
```

## 《软件实践》课程实验报告

```
... },  
});
```

在实现同心圆布局的过程中, 由于不同省份之间的大学数量差距较大, 导致了在自适应画布生成的圆形时半径相差甚远, 给浏览者阅读图谱造成了一定障碍, 同时图谱本身也欠美观。于是, 我在上述布局属性细节中针对同心圆节点之间的最小间距做出了设计, 在反复测试中选择了合适的数值, 增加代码: `minNodeSpacing: 50`, 解决了这项问题。

(3) 接着是设置合适的缩放比例及范围。在之前的网页 demo 中, 对于知识图谱的缩放比例并没有限制, 导致在查看图谱时常常发生小问题: 在上下滑动网页时不小心将知识图谱进行了缩放, 此外, 缩放比例过大导致在画布里找不到节点信息, 缩放比例过小导致节点信息无法清晰可见。但考虑到在查看图谱时浏览者有放大/缩小图谱的需求, 不能因此关闭浏览者缩放图谱的权限。因此我针对缩放比例做了一定的限制, 设置了初始缩放比例、最小缩放比例和最大缩放比例。代码修改如下:

```
cytoscape({  
  container: .....zoom: 1,  
    minZoom: 0.8,  
    maxZoom: 1.1,  
    zoomingEnabled: true,  
    userZoomingEnabled: true, .....  
});
```

(4) 选择合适的节点及边的样式。这一部分内容在之前构建好的知识图谱中已有简单的设置, 此处考虑到网页的美观性, 针对网页整体风格和背景颜色基调对节点的颜色、大小以及边的颜色、样式作出了一定的调整。其中我针对父结点和子结点的大小作出了调整: 将父节点缩小、子节点扩大, 以增加两者的区别度和辨识度, 同时给父节点(省份节点)增加了边框, 突出其中心位置。具体代码如下:

```
cytoscape({  
  style: [  
    ...  
    {  
      selector: 'node[label = "province"]',  
      css: {'background-color': '#f7ba81', 'content': 'data(name)',  
        /* 设置了省份结点样式 */  
        'border-color': '#fcfcfc', 'border-width': '2', 'width': '15px', 'height': '15px'  
      },  
    },  
    ... ]  
});
```

此外, 我针对子结点(包括高校结点、年份科目结点和分数线结点)都做了相应的颜色和大小设置, 针对边的样式(比如直线、点线, 箭头样式等)和颜色也作出了适当的调整, 以达到整体的和谐与美观。整体代码如下:

## 《软件实践》课程实验报告

```
cytoscape({
  style: [
    ...
    {
      selector: 'node[label = "college"]',
      css: {'background-color': '#dcc3cf', 'content': 'data(name)',
        'width': '30px', 'height': '30px'}
    },
    {
      selector: 'node[label = "province"]',
      css: {'background-color': '#f7ba81', 'content': 'data(name)',
        'border-color': '#fcfcfc', 'border-width': '2', 'width': '15px', 'height': '15px'}
    },
    {
      selector: 'edge',
      css: {'content': 'data(relationship)', 'target-arrow-shape': 'triangle',
        'line-color': '#bdc1bf', 'line-style': 'dashed', 'curve-style': 'straight'}
    }
  ],
  ... ]
});
```

3.在查看小组网页效果时，我在前一个网页 map 的基础上添加了背景图片，使得整体风格更加统一。具体操作为在 kgB\_index.html 文件中修改代码：

```
<body background="{%static 'images/3.JPG'%}">
```

# 《软件实践》课程实验报告

## 五、实验结果与分析

修改前的网页见左图（图 1），经过修改和美化后的网页如右图（图 2）

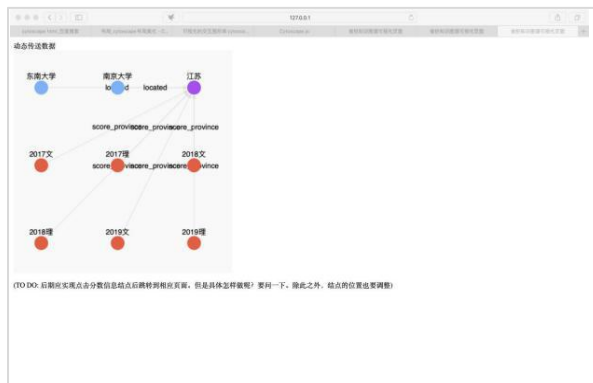


图 1



图 2

通过对比可以看到，与整体网站风格保持一致，网页上基本实现了最初的任务设计。

再看正中央的知识图谱：

- (1) 使用暗粉色、蓝色和橙色分别表示院校、年份科目和省份的节点，使用浅粉色、浅紫色和亮橘色分别表示不同的分数线节点，醒目美观，父节点（即表示省份的节点）用较小的尺寸和白色边框与周围子节点区别开，边使用了灰色的短横线形式，简洁大方；
- (2) 可以通过鼠标放大（或缩小）知识图谱，详细查看知识图谱的细节（或全局），缩放比例范围适中，不会出现比例过大而无法找到其他节点的情况；
- (3) 通过切换不同省份的知识图谱，如下图（图 3、图 4）可以看到：省份节点分布在圆心处、画布的中心，其他子节点在周围以同心圆的形式呈现，同时，当节点个数发生变化时，会自适应地调整节点及节点之间的距离使之均匀分布在画面上，较为美观。

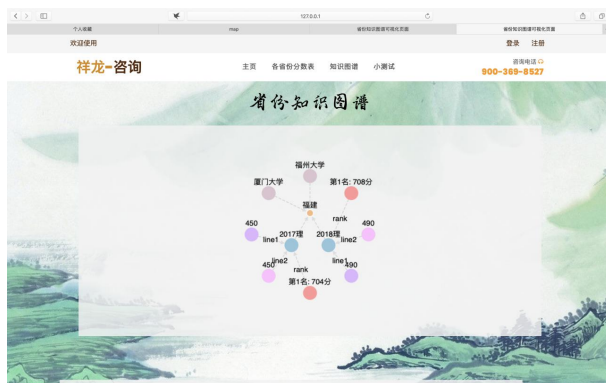


图 3

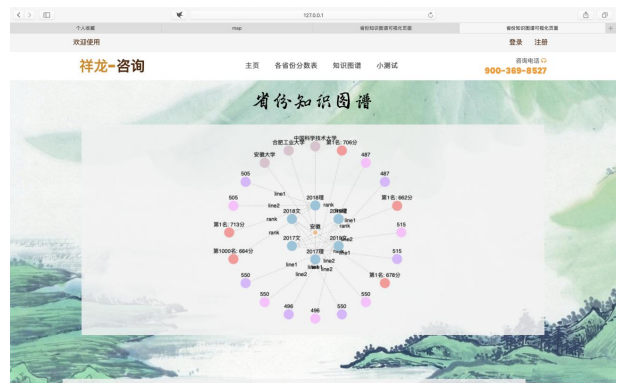


图 4



## 《软件实践》课程实验报告

### 六、实验总结与心得体会

在本次项目过程中，我初步学到了如何使用 Django 框架开发网站，掌握了一些基本的交互功能。

在项目过程中，我主要从需求出发，立即学习相应的知识再付诸实践，运用到项目开发中。

比如，在编写 html 文件的时候，我需要加入 UI 组编写的网页模板，但由于对路由系统掌握得不太好，在插入静态文件时写了错误的路径，出了不少 bug，后来重新温习了短学期第一次上课的课件并上网查了一些资料，顺利地导入了模板。在调试这个小 bug 的过程中，虽然走了一些弯路，浪费了很多时间，但是在摸索中对于 Django 框架的路由系统有了更深入的认识，收获颇丰。

在本次项目过程中，我还对网站前端开发和知识图谱可视化有了初步的认识。我们小组负责知识图谱构建并可视化展示，而我负责知识图谱的可视化及其网页展示。这也是我之前完全没接触过的内容，在这个过程中，我先设计了最终呈现效果，反向推导我需要完成的内容，再分解成小任务，逐个实现。我们小组这次采用了 Cytoscape.js 来可视化知识图谱，网上关于 Cytoscape.js 的介绍不是特别多。我主要是参考官方指南里对各项属性的介绍摸索着实现的。

从一开始的生疏到后来的熟练，在这次实践里我感受到了自学能力的重要性，也学到了很多。

在本次项目过程中，我也体会到了小组合作项目的团队协作感，感受到了自上而下合作的一些困难。一开始在讨论小组任务的时候，大家对我们小组最终要做出的内容没有一个明确的概念，讨论了很多种方案，也否定了很多种提议，最终暂时拟订了一个初步方案。在之后的课程推进中，我们小组不断讨论修正、完善我们的内容，也对最终呈现的结果有一个更加清晰的设计。在这个过程中我深刻感受到了需求分析的重要性，因为当你明确了要做的内容之后，动手实践起来其实是很快的。至于个人任务，我们在小组分工的过程中进行了很多的讨论和尝试。12 个人的小组，一半的人负责后端，剩下一半负责前端，每个人分到的任务量其实都不是很大。这也是我这次项目里最大的遗憾。同时，由于大家对前后端分离的不熟悉，没有提前设计好前后端的衔接，导致前端在后端完成以后才开始进行，时间上跨度比较大，没有很好地发挥前后端分离的优势。

总而言之，通过这次高考志愿填报助手的项目实践，我不仅学到了关于 Django 框架开发网站以及知识图谱可视化的相关知识，也体验到了合作开发项目的团队协作感，收获满满。感谢孔老师这次的任务设计，开启了我开发网站的大门，也让我对网站前端设计有了浓厚的兴趣。

2020 年 9 月制