

《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	知识图谱构建 A 组						
姓 名	陈翼张	专业	人工智能	班级	1 班	学号	09118129
实验时间	2020.8.31-2020.9.23		指导教师	孔祥龙		成绩	

一、实验背景和目的

实验背景：

高考是人生的重要关口，如何选择自己最适合的学校是历年各个考生都关注的重要问题。如何选择专业，如何观察历年录取分数线变化，如何了解各个学校在不同省份的录取分数差异，是每年高考结束以后学生与家长共同的关注焦点。

实验目的：

通过构建知识图谱，训练人工智能算法实现出一个推荐算法，满足学生与家长对高考志愿填报的咨询需求。

二、小组任务和个人任务

小组任务：

小组的名称为“知识图谱构建 A 组”，小组任务为以高校为中心，围绕专业以及其所对应的一级学科设计知识图谱，从而为后续知识推理提供可靠的底层支持，完成更多的查询功能。

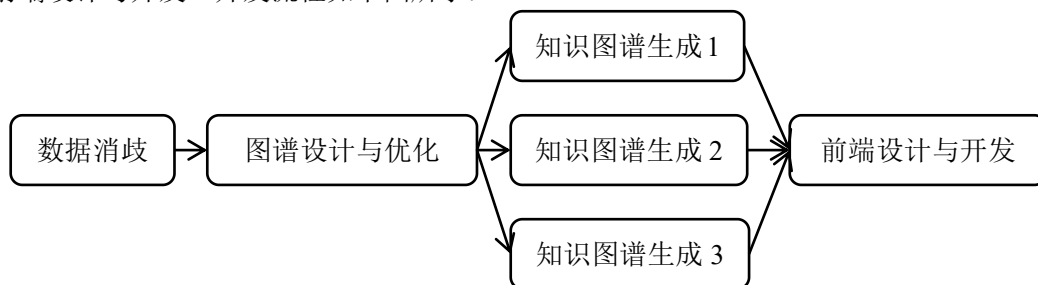
个人任务：

作为组长，负责调度组员，分配任务，与其他组进行对接。同时负责前端开发与知识图谱生成。具体的来说，我要负责协调组员，显示网页的开发，前端网页与数据库的连接，以及协助组员生成知识图谱。

三、个人任务需求分析

1. 需求分析，初始设计以及小组任务

首先作为组长需要与其他组进行对接，其次需要分配任务，由于本次的任务是生成知识图谱，从已有的数据库完善性和易用性上，我选择了 Neo4j 图数据库，在此基础上构建图谱只需要将数据转换成 Neo4j 可接受的格式，就可以直接生成知识图谱，同时 Neo4j 也提供了完善的前端页面供用户使用和查询。我认为构建知识图谱并完成知识图谱显示需要四步：数据消歧，知识图谱设计与优化，将数据导入 Neo4j 并生成知识图谱，前端设计与开发。开发流程如下图所示：



《软件实践》课程实验报告

具体任务与分工设计流程如下：

一. 知识图谱需求分析

利用已有的数据构建一个小型的报考知识图谱(知识库)，通过调用该图谱可以实现如下功能：

1. 已知自己某分数能上什么学校
2. 某个特定的专业哪个学校分数最高
3. 已知自己的分数判断自己能学什么样的专业
4. 查询某学校的特定专业
5. 我只想学 XX 专业，能去什么学校？

二. 初始设计

使用基于图数据库的 Neo4j 生成此次的知识图谱

知识图谱设计：

实体：学校，专业，分数，年份，对照表

属性：（专业）属于某学校的文/理专业，（属于哪个专业的）对照表

对照表属性设计：开设该专业的所有学校，每年的录取分数

三. 小组任务

任务 1：数据源

本项目需要用到的数据源：是第一组清洗的包含学校，专业，省份，分数，年份的 csv 文件。

任务：从之前的小组获取报考省份及相应分数线的讯息

任务 2：知识图谱设计与优化

利用已有的数据构建一个小型的报考知识图谱(知识库)，通过调用该图 谱可以实现如下功能：

1. 已知自己某分数能上什么学校
2. 某个特定的专业哪个学校分数最高
3. 已知自己的分数判断自己能学什么样的专业
4. 查询某学校的特定专业
5. 我只想学 XX 专业，能去什么学校？

任务 3：知识图谱数据准备

此任务包括：

1. 对专业名称进行消歧处理
2. 为有需要的实体生成标识符

任务 4：创建可以导入 Neo4j 的 csv 文件

《软件实践》课程实验报告

在第一个任务里,我们已经分获取了包含所有信息的 csv,但这些文件不能直接导入到 Neo4j 数据库。所以需要做一些处理,并生成能够直接导入 Neo4j 的 csv 格式。我们需要生成这几个文件:暨设计环节决定的实体及这些实体之间相互关系对应的 csv 文件。对于格式的要求,参考:<https://neo4j.com/docs/operations-manual/current/tutorial/import-tool/>

任务 5: 利用上面的 csv 文件生成数据库

使用 neo4j 命令把所有的数据导入到 Neo4j 中,数据默认存在 graph.db 文件中,重启 Neo4j 服务,通过 localhost:7474 观察知识图谱。

使用自带的命令进行简单查询的测试,如:

```
# 查询 node

MATCH (n:Concept) RETURN n LIMIT 25# 查询 relationship

MATCH p=()-[r:industry_of]->() RETURN p LIMIT 100
```

任务 6: 基于构建好的知识图谱,构建显示网页

由于生成知识图谱的过程较为复杂,因此我将知识图谱的生成分为三部分进行平行开发,以加快开发进度。

2. 构建显示网页:

作为前端设计与开发人员,在拿到知识图谱的生成结果后,我认为首先应该制作一个导向页面,该导向页面应支持一些常见问答的储存与展示,同时由于 Neo4j 官方给出的知识图谱前端显示已经足够完善,因此不另外造轮子。

3. 知识图谱的生成:

在知识图谱的生成方面,我主要是给出了现成的知识图谱构建项目供组员们按图索骥。

四、实验过程(需附上关键代码及相关说明)

首先选定前端的显示风格,本次开发我选取了 next 极简风格。下载模板文件,并在页面中进行调用,同时设计一些常用的显示字体:

```
<!DOCTYPE html>

<!-- saved from url=(0023)http://www.cyz.cn/ -->{% load static %}

<html class="theme-next pisces use-motion" lang="zh-Hans"><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">

<meta http-equiv="Cache-Control" content="no-transform">

<meta http-equiv="Cache-Control" content="no-siteapp">

<meta name="keywords" content="个人网站, next 主题, ">

<meta name="description" content="django 框架建立的 next 主题">
```

《软件实践》课程实验报告

```
<link href="../../static/css_js/jquery.fancybox.css" rel="stylesheet" type="text/css">
<link href="../../static/css_js/font-awesome.min.css" rel="stylesheet" type="text/css">
<link href="../../static/css_js/main.css" rel="stylesheet" type="text/css">
<link href="../../static/css_js/extra.css" rel="stylesheet" type="text/css">
<link rel="alternate" href="../../static/image/fc.png" title="我的问题" type="application/atom+xml">
<link rel="shortcut icon" type="image/x-icon" href="../../static/image/logo.ico">
<link rel="canonical" href="http://www.cyz.cn/">
<style>
    span.highlighted {
        color: red;
    }
</style>

<script src="../../static/css_js/push.js"></script>
<script src="../../static/css_js/hm.js"></script>
<script type="text/javascript" id="hexo.configurations">
    var NexT = window.NexT || {};
    var CONFIG = {
        root: '/',
        scheme: 'Pisces',
        sidebar: {"position":"left","display":"always"},
        fancybox: true,
        motion: true,
        duoshuo: {
            userId: '114514',
            author: 'cyz'
        },
        algolia: {
            applicationID: '',
            apiKey: '',
            indexName: '',
            hits: {"per_page":10},
            labels: {"input_placeholder":"Search for Posts","hits_empty":"We didn't find any results for the search: ${query}","hits_stats":"${hits}
results found in ${time} ms"}
        }
    };
</script>

<script type="text/javascript">
    var _hmt = _hmt || [];
    (function() {
        var hm = document.createElement("script");
        hm.src = "https://hm.baidu.com/hm.js?c5127ebccef09f61efbb5ce4885b2c1f";
    })();
</script>
```

《软件实践》课程实验报告

```
var s = document.getElementsByTagName("script")[0];
s.parentNode.insertBefore(hm, s);
})();
</script>
```

```
<title> {% block title %}{% endblock %} </title>
```

```
<style>
```

```
    .a{
        font-size: 8px;
        color: black;
        font-weight: bold;
    }
```

```
    .b{
        font-size: 30px;
        color: antiquewhite;
        font-weight: bolder;
        line-height: inherit;
    }
```

```
    .c{
        font-size: 15px;
        color: antiquewhite;
        font-weight: bolder;
    }
```

```
    .d{
        font-size: 15px;
        color: red;
        font-weight: bold;
    }
```

```
</style>
```

为了保证前端代码的通用性，使用模板与继承改写前端：

```
<title> {% block title %}{% endblock %} </title>
```

```
{% block content %}{% endblock %}
```

```
{% extends 'base.html' %}
```

```
{% block title %}
```

```
知识图谱导向
```

```
{% endblock %}
```

```
{% block content %}
```

《软件实践》课程实验报告

在基本框架搭建完毕后，考虑如何对 QA 内容进行显示与保存，从易用性与稳定性上说，本次使用 MySQL 进行储存，设置如下：

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'myblog',          #数据库名
        'USER': 'root',           #用户名
        'PASSWORD': 'abc123456',  #密码
        'HOST': '127.0.0.1',      #本机地址
        'PORT': '3306',           #端口
    }
}
```

QA 显示设置如下：

```
{% for blog in all_blog.object_list %}

<article>

<h1>  {{ blog.title }}</h1>

<h5 style="text-indent:2em;">{{ blog.create_time|date:'Y-m-d' }}</h5>

<h5 style="text-indent:4em;">{{ blog.category.name }}</h5>

<p style="text-indent:6em;">{{ blog.content }}</p>

</article>

{% endfor %}
```

显示分页设计：使用 pure pagination:

```
PAGINATION_SETTINGS = {
    'PAGE_RANGE_DISPLAYED': 3,    #中间显示的个数
    'MARGIN_PAGES_DISPLAYED': 2,  #两边显示的个数
    'SHOW_FIRST_PAGE_WHEN_INVALID': True,
}
```

Html 分页设置：

```
<nav class="pagination">
    <div class="pageturn">
        <ul class="pagelist">
            {% if all_blog.has_previous %}
                <li class="long"><a href="?{{ all_blog.previous_page_number.querystring }}">上一页</a></li>
            {% endif %}
            {% for page in all_blog.pages %}
                {% if page %}
                    {% ifequal page all_blog.number %}
```

《软件实践》课程实验报告

```
<li class="active"><a href="{{ page.querystring }}">{{ page }}</a></li>

{% else %}

    <li><a href="{{ page.querystring }}" class="page">{{ page }}</a></li>

{% endifequal %}

{% else %}

    <li class="none"><a href="">...</a></li>

{% endif %}

{% endfor %}

{% if all_blog.has_next %}

    <li class="long"><a href="{{ all_blog.next_page_number.querystring }}">下一页</a></li>

{% endif %}

</ul>

</div>

</nav>
```

五、实验结果与分析

前端显示如图:

知识图谱

导向页面

□ 首页

□ 提出问题

□ 搜索



第三小组

□ 图谱链接

□ GitHub

□ 知乎

□ neo4j安装...

□ Links

W3school

隔壁组知识图谱

鸣谢：知识图谱设计&优化

2020-09-25

表彰

朱佳涛&白劭宸

贡献者：消歧&数据处理

2020-09-25

表彰

李浩天和他的朋友们

问题4

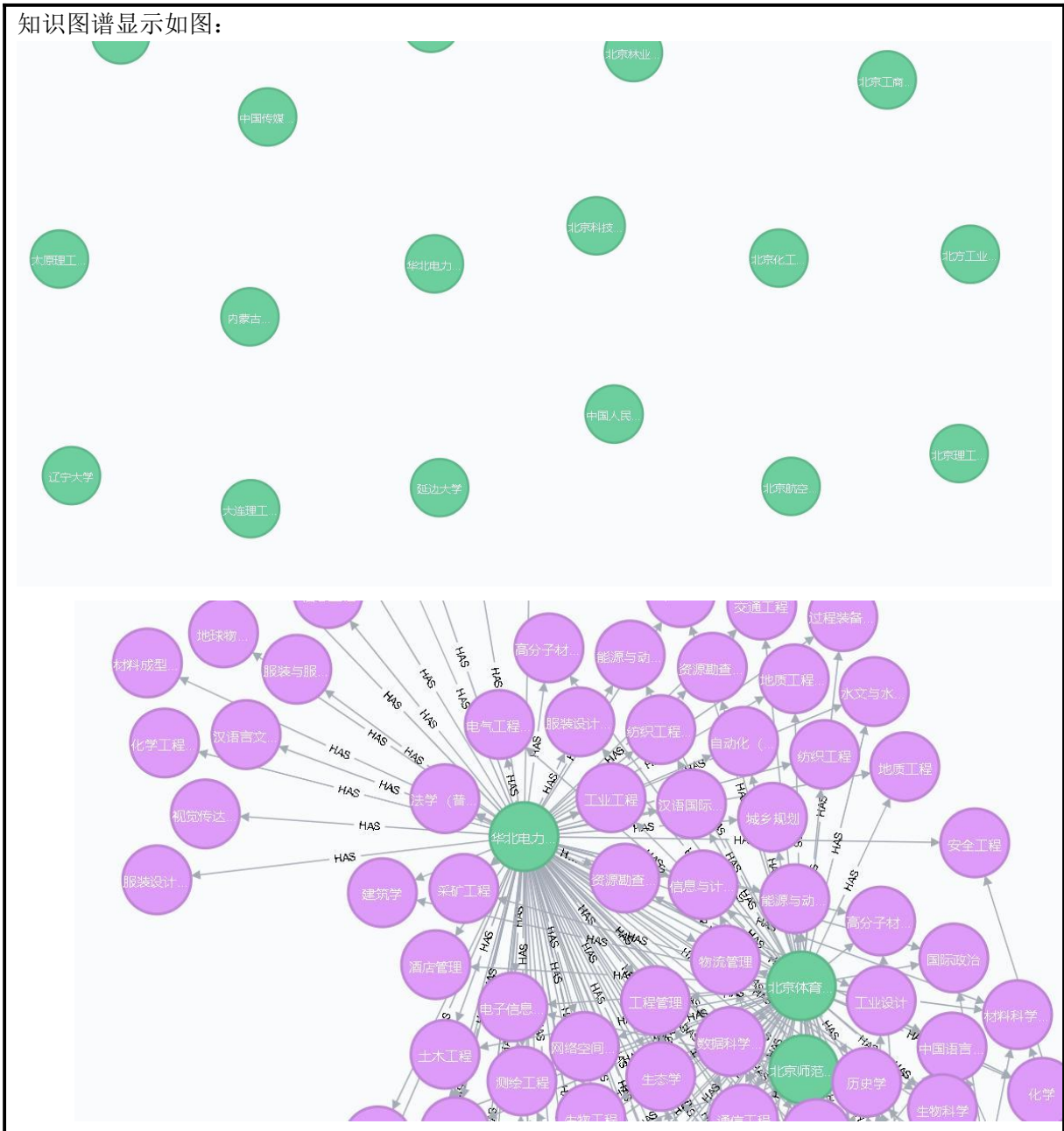
2020-09-24

问题

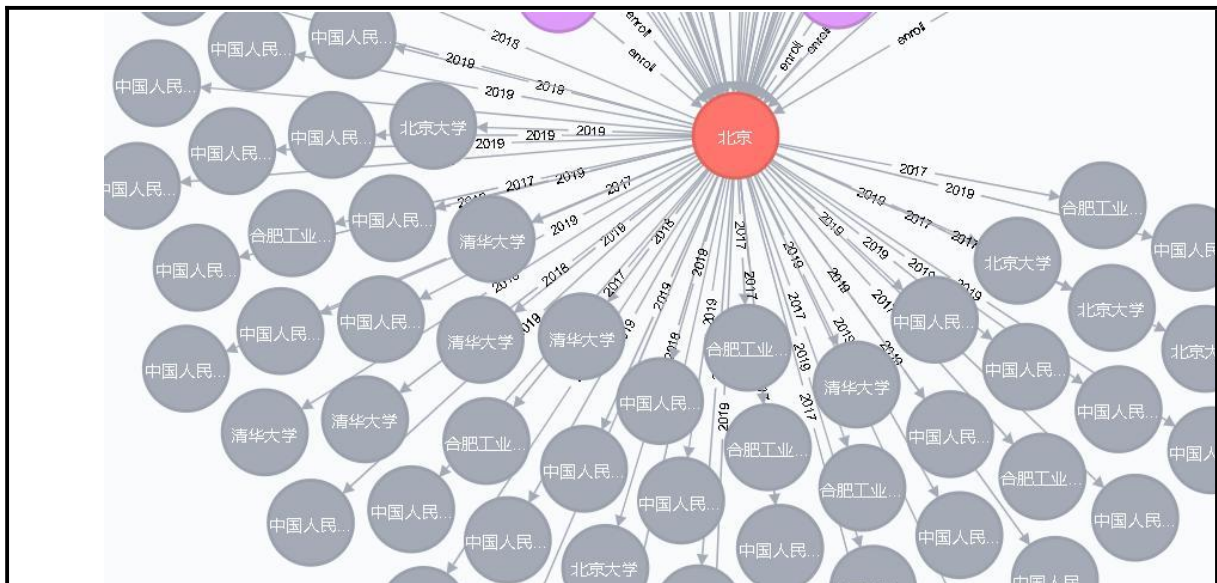
所有浏览器都支持 <iframe> 标签。定义和用法 iframe 元素会创建包含另外一个文档的内联框架（即行内框架）。HTML 与 XHTML 之间的差异 在 HTML 4.1 Strict DTD 和 XHTML 1.0 Strict DTD 中，不支持 iframe 元素。提示和注释：提示：您可以把需要的文本放置在 <iframe> 和 </iframe> 之间，这样就可以应对无法理解 iframe 的浏览器。

《软件实践》课程实验报告

知识图谱显示如图：



《软件实践》课程实验报告



总体而言，实验效果明显，基本实现了设计预期的功能

六、实验总结与心得体会

本次实验我主要完成了如下任务：小组分工设计与任务安排，前端设计与开发，辅助生成知识图谱。提交了如下文件：两个前端 html 页面，一个包含了 css 文件，图标与图片的文件夹和数个相关的设置文件。通过这次实验，我完整体验了一次多人项目实现的流程，学会了多人合作下 git 仓库的使用，同时也切身体会到不同团队之间，同一团队的成员之间即时沟通和协商合作的重要性。这次课程的经验将为我以后的项目经历提供很大的帮助

2020 年 9 月制