**本 科 毕 业 设 计 (论 文)**

**基于招聘网站信息的数据分析与展示系统的设计与实现**

**Design and Implementation of Data Analysis and Display System Based on Recruitment Website Information**

2019年 6 月

**毕业设计（论文）中文摘要**

|  |
| --- |
| 基于招聘网站信息的数据分析与展示系统的设计与实现  摘 要：  在信息化的时代，数据意味着价值，通过网络爬虫获取到互联网上有意义的数据进行可视化处理已成为研究数据的主要方法，而Python作为一门随着人工智能的火爆而逐渐成为潮流的语言，在数据获取上有着它独特的优势，通过Python自带的库编写网络爬虫程序，既简单又高效。而纵观整个招聘网站行业，作为随着计算机技术的一同飞速发展以及智能手机等移动终端的普及而逐渐被广大人们所正视的网站，其拥有的数据量不可忽视，但绝大多是招聘网站并没有对这些数据进行统计分析，研究其发展趋势，市场分布等情况。  本文首先针对市场上的招聘网站现状进行分析，通过对网络爬虫技术的了解和对招聘网站的使用估测了爬取招聘网站岗位信息的可行性，进行了基于招聘网站信息的数据分析与展示系统的需求分析，并按照软件开发流程，一步一步的完成了爬虫模块，用户功能模块，管理员功能模块这几个部分，最终通过Flask框架完成了网站的基本搭建，主要用于展示爬取到的岗位信息。  关键词：Python；Flask；爬虫；招聘网站；数据分析； |

**毕业设计（论文）外文摘要**

|  |
| --- |
| Design and Implementation of Data Analysis and Display System Based on Recruitment Website Information  **Abstract:**  In the era of informationization, data means value, and it has become the main method of researching data to obtain meaningful data on the Internet through web crawlers, and Python has become a trend with the popularity of artificial intelligence, which has its unique advantages in data acquisition and whose library to write a web crawler program through, both simple and efficient. In view of the entire recruitment website industry, as a result of the rapid development of computer technology together with the popularity of mobile terminals such as smart phones, websites that are being viewed by the majority of people gradually have a large amount of data, but most of them are recruitment websites without statistics analysis and studying its development trend, market distribution, etc.  This article first analyzes the current situation of recruitment websites in the market, estimates the feasibility of crawling job site information through the understanding of web crawler technology and the use of recruitment websites, and makes an analysis of users’ requirements based on the data analysis and display system of recruitment website information. In accordance with the software development process, the reptile module, the user function module, and the administrator function module are completed step by step, and the basic construction of the website is finally completed through the Flask framework, whose function lies in exhibiting the crawled post information.  **Key words:**. Python; Flask; crawler; recruitment website; data analysis; |

目 录

[1 绪论 1](#_Toc30319_WPSOffice_Level1)

[1.1 课题研究的意义 1](#_Toc21440_WPSOffice_Level2)

[1.2 发展水平状况和发展趋势 1](#_Toc26912_WPSOffice_Level2)

[1.3 可行性分析 2](#_Toc17233_WPSOffice_Level2)

[1.3.1 经济可行性 2](#_Toc26811_WPSOffice_Level3)

[1.3.2 社会可行性 2](#_Toc20282_WPSOffice_Level3)

[1.3.3 技术可行性 2](#_Toc9531_WPSOffice_Level3)

[1.3.4 操作可行性 2](#_Toc12395_WPSOffice_Level3)

[1.4 关键技术介绍 3](#_Toc23740_WPSOffice_Level2)

[2 需求分析 3](#_Toc1200_WPSOffice_Level1)

[2.1 需求概述 3](#_Toc29621_WPSOffice_Level2)

[2.2 功能介绍 4](#_Toc13198_WPSOffice_Level2)

[2.3 数据流分析 4](#_Toc6757_WPSOffice_Level2)

[2.4 数据需求 7](#_Toc5296_WPSOffice_Level2)

[3 总体设计 8](#_Toc25954_WPSOffice_Level1)

[3.1 系统总体功能结构设计 8](#_Toc29579_WPSOffice_Level2)

[3.2 数据结构设计 9](#_Toc7987_WPSOffice_Level2)

[3.2.1 数据概要结构设计 9](#_Toc1208_WPSOffice_Level3)

[3.2.2 数据逻辑结构设计 12](#_Toc14161_WPSOffice_Level3)

[3.3 开发环境与运行环境设计 14](#_Toc28485_WPSOffice_Level2)

[4 详细设计与实现 15](#_Toc22938_WPSOffice_Level1)

[4.1 系统首页设计与实现 15](#_Toc17118_WPSOffice_Level2)

[4.2 拉勾网爬虫模块设计与实现 16](#_Toc1491_WPSOffice_Level2)

[4.2.1 反爬虫机制介绍 16](#_Toc15147_WPSOffice_Level3)

[4.2.2 爬虫实现过程 16](#_Toc23504_WPSOffice_Level3)

[4.3 信息查询与展示模块设计与实现 20](#_Toc10280_WPSOffice_Level2)

[4.3.1 简单信息查询与展示 20](#_Toc11744_WPSOffice_Level3)

[4.3.2 简单统计信息查询与展示 21](#_Toc23781_WPSOffice_Level3)

[4.3.3 精确信息查询与展示 22](#_Toc23720_WPSOffice_Level3)

[4.3.4 精确统计信息查询与展示 23](#_Toc407_WPSOffice_Level3)

[4.4 用户登录注册模块设计与实现 25](#_Toc21052_WPSOffice_Level2)

[4.5 用户个人收藏模块设计与实现 26](#_Toc26811_WPSOffice_Level2)

[4.6 用户个人主页模块设计与实现 26](#_Toc20282_WPSOffice_Level2)

[4.7 后台管理模块设计与实现 27](#_Toc9531_WPSOffice_Level2)

[4.7.1 用户管理 27](#_Toc1557_WPSOffice_Level3)

[4.7.2 招聘统计信息管理 28](#_Toc16108_WPSOffice_Level3)

[4.7.3 爬虫管理 29](#_Toc8418_WPSOffice_Level3)

[5 用户手册 29](#_Toc20474_WPSOffice_Level1)

[5.1 系统功能及运行环境简介 30](#_Toc12395_WPSOffice_Level2)

[5.2 系统运行与操作指南 30](#_Toc14493_WPSOffice_Level2)

[6 系统评价 31](#_Toc14493_WPSOffice_Level1)

[6.1 系统主要的优点和特色介绍 31](#_Toc17970_WPSOffice_Level2)

[6.2 系统存在的不足与改进方案 31](#_Toc11863_WPSOffice_Level2)

[6.2.1 系统不足 31](#_Toc14493_WPSOffice_Level3)

[6.2.2 改进方案 31](#_Toc17970_WPSOffice_Level3)

[7 毕业设计心得与收获 32](#_Toc17970_WPSOffice_Level1)

[结论 32](#_Toc11863_WPSOffice_Level1)

[致谢 32](#_Toc25966_WPSOffice_Level1)

[参考文献 32](#_Toc19203_WPSOffice_Level1)

1. 绪论
   1. 课题研究的意义

由国家教育部与统计局在2017年发布的统计数据可以知道，在中国共有在校大学生2695.8万，应届大学毕业生795万。面对如此庞大的毕业生人群，就业难早已不是一个新鲜的话题了，而很多高校的学子们更是在毕业前就投入求职的浪潮，提前适应就业压力，避免毕业等于失业的情况出现。国家也是出台相关政策，扩大岗位需求，鼓励自主创业，从宏观上把控就业问题。但对于个人而言，若想在某一领域获得一席之地，找到一份不错的工作岗位，自身硬实力始终是第一位，要能够跟上发展的脚步，满足企业的需求，从自身出发解决就业问题。

通过对招聘网站上的信息进行获取并统计分析，可以得知目前某一领域或行业内的岗位缺口量或是热门招聘数，从而更好的对自己的职业发展进行规划，选择符合市场需要也符合自身发展需要的道路，这些信息无论是对于在校学生亦或是在职人员都具有很高的价值，通过切切实实的数据与图表展示出来，可以给用户对于现在市场情况最直观的反映，而不是道听途说，心里没底的规划自身的未来发展道路。并且如今是一个信息爆炸、信息公开的时代，但信息过于繁杂也会让人眼花缭乱，无法获得自己想要的信息，而招聘网站中只是发布岗位信息而已，所以通过获取招聘网站上的信息并展示出来是十分有意义的。

* 1. 发展水平状况和发展趋势

随着计算机技术的飞速发展与智能手机、电脑等硬件设备的普及，招聘网站也随之不断进化，从最初的门户网站、BBS论坛下的招聘版块发展为如今各具特色的招聘网站，例如涉及领域范围广的智联招聘，专注于中高端人才的猎聘网，主打计算机领域的拉勾网等等。由于这些网站与相应的应用广泛存在于PC端和移动端中，其用户数以千万计，不论是对于用户还是企业而言，都是一个很好的平台。

经过多年的发展，如今的招聘网站行业也愈发专业化、成熟化，企业在招聘网站中注册账号发布职位信息，职位信息中包含了企业简讯、工作薪资、工作地点、工作岗位、工作要求等等一系列，既表明了企业对于拥有能力的求职者的诉求，又给求职者指引方向，求职者只需在招聘网站中注册账号然后寻找符合自身条件的岗位投递简历，等待企业联系即可，避免了以往求职者尤其是大学毕业生在城市中像无头苍蝇没有方向的寻找工作，而企业人才难求的状况。

招聘网站如此正规化，但市场上还并未出现针对招聘网站的信息进行统计分析的网站，故而一个对于招聘网站信息获取和展示的网站是很有开发必要与发展前景的，而各种数据挖掘，数据分析技术也会让数据更加准确，让网站更加成熟和规范。

* 1. 可行性分析
     1. 经济可行性

本课题是为了收集招聘网站上的岗位信息并统计出来给广大想要努力奋斗的人们对于市场需要的大致分析，所有开发过程均在单机上进行，通过学校图书馆进行资料查找，通过对招聘网站的使用进行调研分析，无过多额外花费。此外，本系统的开发语言为Python，故而对开发环境和机器配置要求均不算太高，无论是Windows系统还是Linux系统都可以进行本系统的开发，由于是基于B/S架构的系统，主要花费在服务器的租用上，但对于学生机而言，此项成本在可承担范围之内。系统运行起来后，人们只要进入在服务器架设好的展示网站即可查看信息。因此该系统在经济方面没有太大问题，可以开发。

* + 1. 社会可行性

对于广大高校在校生而言，本系统可以让其了解目前市场对于哪类人才的招聘岗位较多，对于自身专业在市场中的分布情况以及发展潜力有所认识，树立自己的职业发展目标。而对于在职人员而言，本系统可以让其了解自身工作行业的人才需求和技能需求，明确学习方向，提高核心竞争力。人们可以在本系统中看到自己想看到的统计信息，而不是用传统的去网上查找，节约了很多精力和时间成本，而且对于招聘网站而言，开放给用户可见的信息是可以对其进行收集的，并且本系统采集到的信息并未进行商业用途。因此，在该系统在社会可行性上不存在问题。

* + 1. 技术可行性

本系统采用常见的B/S开发架构，即浏览器与服务器模式，开发语言使用了时下较为流行的Python语言，版本为3.6.4，网站的搭建使用了基于Python的Flask轻量级框架，Flask拥有良好的扩展性，十分适合开发一些中小型的Web应用程序并且默认支持jinja2的模板引擎，数据库使用MySQL并使用Navicat Premium数据库管理工具对其进行管理，并导入Python第三方模块Flask-SQLAlchemy用于数据表与对象的映射，而前端则使用了bootstrap框架用于美化界面，以便同时支持PC端和移动端，并且使用了Echarts插件来对数据进行图表化。

本系统的网络爬虫采用Python编写，在选取招聘网站爬取招聘信息的问题上，在经过了对智联招聘，前程无忧等多个招聘网站的使用和调查后，考虑到数据真实性以及反爬虫技术，最终决定爬取拉勾网上的职位信息，而考虑到系统内存与服务器内存问题，故而选择仅对计算机相关岗位信息进行爬取并保存到数据库中，这样对于服务器负载也比较轻松，可以实现系统所需的各个功能。因此，在技术方面的问题均可解决并实现，技术可行性通过。

* + 1. 操作可行性

本系统面向用户的主要功能是将从招聘网站中获取到的信息给用户展示出来，界面设计简约直观清晰，只要合理使用查询功能便可以看到自己所想要看到的，操作十分人性化。而对于管理员而言，后台的管理模块功能划分明确，简单易懂，每个模块也有相应的功能提示，因此，该系统在操作方面是可以进行开发的。

* 1. 关键技术介绍

Python网络爬虫：网络爬虫从本质上而言是一个可以自动获取到网页上文本内容的程序，主要应用于类似搜索引擎功能的应用，可以为搜索引擎从万维网上下载网页，现在已经成为搜索引擎不可获缺的重要组成部分。而随着Python语言的兴起，这一跨平台的脚本语言也成为了编写爬虫程序的新宠，相较于传统的Java，Python有更高的拓展性，更丰富的第三方模块以及更易上手的语法，使得越来越多的人使用Python并为其不断拓展第三方模块，使得Python网络爬虫也更易编写，具有更为强大的功能。在本系统中，爬虫模块只是用于获取招聘网站上的数据，利用Python request对象获取响应招聘网站目标URL返回的内容并从中解析出想要的文本数据，保存到本地数据库中。

Flask Web框架：Flask拥有众多为其开发的第三方库，只需要通过import语句导入便可以使用，使其拓展性大大增加，十分便利，且自带可开启debug模式服务器，属于轻量级的Web应用框架，且拥有Jinja2模板引擎，所以对于中小型网站的搭建有很大的优势。

Flask-SQLAlchemy：SQLAlchemy是Python支持的第三方模块，在导入此模块后可以使用SQL工具包来进行SQL操作，且对SQL语句进行了优化，大大加快乐查询的速度，最重要的是，SQLAlchemy提供对象关系映射（ORM）工具,即通过Python对象来创建数据库对象，使得开发更加便捷。

Echarts：Echarts是基于 JavaScript 的第三方可视化库，同样也是百度的开源项目，在其官方网站上有很多中文文档和实例，学习成本较低。通过Echarts编写的前端页面可以流畅的运行在 PC 和移动设备上，并且兼容很多目前常用的浏览器（IE8/9/10/11，Chrome，Firefox，Safari等），其底层依赖ZRender（一个轻量级的矢量图形库），通过调用API，可以实现简单直观，可以个性化定制的数据可视化图表。

1. 需求分析
   1. 需求概述

随着招聘网站，招聘软件的不断普及，这些反映企业普遍需求的网站中的数据与信息从侧面也描绘了市场需求的走势图，但纵观整个招聘网站范围，通过对这些网站的使用可以看到，它们并没有一个能够展示岗位统计信息的功能，也没有一个提供岗位数据的接口，只是仅仅将岗位发布出来而没有对这些岗位信息进行统计，故而本系统主要有两大任务，利用网络爬虫获取招聘岗位信息并将整理清洗好的数据保存到数据库中和将数据库中的信息在后台进行分类统计并通过网站动态展示出来。

通过以上描述可以得知，基于招聘网站信息的数据分析与展示系统的核心为网络爬虫模块与数据展示模块，而网络爬虫应该由系统管理员进行管理，普通用户能够通过搜索关键字查看网站展示出的信息，并且管理员可在网站上对于用户和数据进行管理，如删除不合理数据，更新用户信息等功能。

综上所述，可以列出本系统的功能列表：用户登录与注册，用户个人信息修改，用户收藏信息，招聘网站信息爬取，管理用户，管理招聘信息。

* 1. 功能介绍

根据需求概述，可以列出本系统的功能列表如下：

面向用户：

1. 登录与注册功能：用户可以在本系统中注册账号然后进行登录来保存相关岗位信息与统计信息。
2. 个人信息管理功能：用户可以在登录本系统后查看个人信息情况，并可以对个人账号信息进行适当的修改，如修改昵称，密码等操作。
3. 收藏功能：用户登录后可以对系统中的岗位信息和统计信息进行收藏操作，并且可以在“我的收藏”这一用户界面中查看之前收藏过的相关信息。

面向管理员，管理员应可以使用本系统的所用功能，以便及时发现问题：

1. 普通用户可使用的功能：包含了普通用户可使用的所用功能。
2. 用户管理功能：管理员可以在用户管理界面查看系统当前的用户列表并对用户进行相应的操作，如修改用户信息，删除用户等。
3. 统计信息管理功能：管理员可以统计管理页面查看系统当前数据库中的岗位列表，统计列表，包含了岗位信息和统计信息，可以对这些信息进行修改，也可以删除一些长时间过期冗余数据。
4. 爬虫管理功能：管理员可以在爬虫管理页面输入网络爬虫目标URL地址，并给与一些参数来让爬虫更好的进行数据的爬取。
   1. 数据流分析

根据需求概述，可以确定使用本系统的所有角色，并确定使用者在使用本系统时与系统之间进行交互的信息，还需要分析额外的外部接口与文件接口，下面将使用数据流图分析本系统的数据流。

1. 在本系统中主要有两种角色，普通用户和管理员用户，不同的用户体验不同的功能，基于招聘网站信息的数据分析与展示系统的顶层数据流图如图2.1所示。

图2.1 顶层数据流图

1. 本系统的信息查询与展示功能模块无需用户登录即可使用，使用者可以对感兴趣的关键字进行查询并查看相关统计信息，其数据流图如图2.2所示。

图2.2 信息查询与展示数据流图

1. 无论是普通用户还是管理员在使用本系统时，除了信息查询与展示功能以外需进行登录操作，用户需输入账号和密码传递给后台服务器，后台服务器拿到账号密码后将其与数据库中保存的用户数据进行比对，若是合法用户再判断其权限，显示对应的系统界面，其数据流图如图2.3所示。



图2.3 用户登录数据流图

1. 普通用户在本系统中的1层数据流图如图2.4所示，定义了普通用户在本系统中能使用的功能，并确定了数据的流向。普通用户在进入系统后可以进行对关键字岗位的简单查询，并查看统计信息，也可以对岗位数据进行精确查询并查看统计信息。



图2.4 普通用户功能1层数据流图

1. 管理员在本系统中的1层数据流图如图2.5所示，定义了管理员在本系统中能使用的功能，并确定了数据的流向。管理员用户除了可以进行普通用户能进行的操作以外，还可以对用户、岗位信息、爬虫程序进行管理。



图2.5 管理员功能1层数据流图

* 1. 数据需求

通过上述功能介绍与数据流分析可以分析出整个系统的数据需求，下面将使用数据字典来将数据需求表示出来。

1. 名字：用户注册

描述：用户首次登录时需注册，注册的用户信息将存储在数据库中。注册成功即可登录。

定义：用户注册=账号+密码+昵称

位置：用户表。

1. 名字：用户登录

描述：用户若想进行收藏操作或管理员要使用管理功能时需要登录。用户信息保存在数据库中，登录时判断账号密码匹配则登录成功。

定义：用户注册=账号+密码+用户权限

位置：用户表。

1. 名字：用户信息

描述：用户登录后，可以在个人信息里编辑自己的信息，保存后可修改。

定义：用户信息=用户ID+账号+密码+昵称+用户权限

位置：用户表

1. 名字：岗位信息

描述：利用网络爬虫从招聘网站上获取到的信息片段

定义：岗位信息=岗位ID+岗位名+公司名+工作城市+薪资+学历+工作经验+关键词

位置：岗位信息表

1. 名字：统计信息

描述：将数据库中保存的数据进行统计

定义：统计信息=统计ID+工作地点+薪资+学历+数量+关键词

位置：统计信息表

1. 名字：收藏信息

描述：用户可以将某个查询的结果保存在收藏内，可以收藏岗位，也可以收藏统计结果。

定义：收藏信息=收藏编号+用户编号+统计信息编号/岗位编号+收藏时间

位置：收藏信息表

1. 总体设计
   1. 系统总体功能结构设计

由于本系统属于半开放式展示网站，任何访问者无需注册都可以使用本系统的信息查询与展示功能模块，普通用户可以使用收藏等功能，管理员可以使用全部功能，故而将按照访问者-普通用户-管理员的顺序来构建整个系统功能模块，层层递进，逐渐深入的来描述。系统的功能结构图如图3.1至3.3所示。

访问者：



图3.1 基于招聘网站信息的数据分析与展示系统功能结构图（访问者）

普通用户：



图3.2 基于招聘网站信息的数据分析与展示系统功能结构图（普通用户）

管理员：



图3.3 基于招聘网站信息的数据分析与展示系统功能结构图（管理员）

* 1. 数据结构设计
     1. 数据概要结构设计

通过对数据流的分析以及数据字典的设计可以得知本系统中的实体对象模型，下面将用实体-关系模型（E-R图）来表示基于招聘网站信息的数据分析与展示系统的数据概要结构。

1. 在本系统中，无论是普通用户还是管理员都将归为用户这一对象，其对象模型如图3.4所示。



图3.4 用户对象模型

1. 在本系统中，通过爬虫爬取的岗位信息将以岗位这一对象来保存，其对象模型如图3.5所示。



图3.5 岗位对象模型

1. 在本系统中，为了将数据库中的岗位信息分类统计，便于数据持久化以及便于将来查询，所以建立统计对象，其对象模型如图3.6所示。



图3.6 统计对象模型

1. 通过上述三个对象模型可以简单描绘出基于招聘网站信息的数据分析与展示网站的整体脉络，用户可以查看多个岗位信息与统计信息，故而用户与岗位和分析是一对多的关系，而一个关键词可以查询到多个岗位，也可以通过一个关键词爬取多个岗位信息，所以关键字与岗位是一对多的关系，为了吸引用户并且收集不同时间的岗位信息数据，需要允许用户将感兴趣的岗位信息和统计信息收藏起来，并保存到个人收藏中，用户可以有多个收藏，所以用户与收藏也是一对多的关系，管理员通过关键字对象利用爬虫程序爬取岗位信息。与系统整体E-R 图如图3.7所示。



图3.7 系统整体E-R图

* + 1. 数据逻辑结构设计

1. 在得到了E-R图之后，对E-R图进行分析就可以得到基于招聘网站信息的数据分析与展示系统的系统关系模式：

用户表（用户ID，昵称，用户账号，用户密码，用户权限，年龄，所在城市，所处行业，当前职业，公司名，个性签名）

岗位表（岗位ID，岗位名，公司名，所在城市，薪资，学历要求，工作经验，创建时间，搜索关键词）

统计表（统计ID，城市列表，城市统计数据，薪资列表，薪资统计数据，学历列表，学历统计数据，统计时间，搜索关键字）

关键词表（关键词ID，关键词名，关键词数）

收藏表（收藏ID，用户ID，统计ID，收藏日期，搜索关键字）

1. 通过关系模式的设计，再对关系模式进行优化，可以在数据库中建立基于招聘网站信息的数据分析与展示系统的数据表，数据表详细内容如下所示：
   1. 用户表（user）：用户的数据表user，设置userID为主键，并让其自动递增，user表中保存了所有用户的信息。表的结构如表3.1所示。

表3.1 用户表（user）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 说明 | 数据类型及长度 | 是否关键字 | 是否可以为空 |
| userID | 用户ID | int | 是 | 否 |
| userName | 昵称 | varchar(20) | 否 | 否 |
| userAccount | 账号 | varchar(20) | 否 | 否 |
| userPwd | 密码 | varchar(20) | 否 | 否 |
| Admin | 权限 | int | 否 | 否 |
| userAge | 年龄 | int | 否 | 否 |
| userCity | 城市 | varchar(20) | 否 | 是 |
| userIndustry | 行业 | varchar(20) | 否 | 是 |
| userProfession | 职业 | varchar(20) | 否 | 是 |
| userCompany | 公司 | varchar(20) | 否 | 是 |
| userSign | 签名 | varchar(50) | 否 | 是 |

* 1. 岗位表（position）：从招聘网站上爬取的岗位基本信息将保存在数据表position中，该表的主键设为在招聘网站中该岗位的positionID。表的结构如表3.2所示。

表3.2 岗位表（position）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 说明 | 数据类型及长度 | 是否关键字 | 是否可以为空 |
| positionID | 岗位ID | varchar(20) | 是 | 否 |
| positionName | 岗位名 | varchar(20) | 否 | 否 |
| companyShortName | 公司名 | varchar(20) | 否 | 否 |
| city | 城市 | varchar(20) | 否 | 否 |
| salary | 薪资 | varchar(20) | 否 | 否 |
| education | 学历 | varchar(20) | 否 | 否 |
| workYear | 工作经验 | varchar(20) | 否 | 否 |
| createTime | 创建时间 | date | 否 | 否 |
| skillName | 关键词 | varchar(20) | 否 | 否 |

* 1. 统计表（stat）：关于某一关键词的搜索结果的统计结果将会保存在数据表stat中，该表的主键为statID且为自动递增类型。表的结构如表3.3所示。

表3.3 统计表（stat）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 说明 | 数据类型及长度 | 是否关键字 | 是否可以为空 |
| statID | 统计ID | int | 是 | 否 |
| city | 城市 | text | 否 | 否 |
| cityNum | 城市统计数据 | text | 否 | 否 |
| salary | 薪资 | text | 否 | 否 |
| salaryNum | 薪资统计数据 | text | 否 | 否 |
| education | 学历 | text | 否 | 否 |
| educationNum | 学历统计数据 | text | 否 | 否 |
| queryDate | 统计时间 | date | 否 | 否 |
| skillName | 关键词 | varchar(20) | 否 | 否 |

* 1. 关键字表（skill）：该表用于爬取数据与检索数据，记录岗位按关键词分类初步统计结果，该表的主键设为自动递增类型，字段名为skillID。表的结构如表3.4所示。

表3.4 关键字表（skill）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 说明 | 数据类型及长度 | 是否关键字 | 是否可以为空 |
| skillID | 关键字ID | int | 是 | 否 |
| skillName | 关键字名 | varchar(20) | 否 | 否 |
| skillNum | 关键词数 | int | 否 | 否 |

* 1. 收藏表（collect）：该表用于保存用户的收藏信息，表中有外键与用户表（user）、岗位表（position）和统计表（stat）关联，建立了表与表的映射关系。表的结构如表3.5所示。

表3.5 收藏表（collect）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 说明 | 数据类型及长度 | 是否关键字 | 是否可以为空 |
| collectID | 收藏ID | int | 是 | 否 |
| userID | 用户ID | int | 否 | 否 |
| positionID | 岗位ID | varchar(20) | 否 | 是 |
| statisticsID | 统计ID | int | 否 | 是 |
| collectDate | 收藏时间 | date | 否 | 否 |
| queryName | 关键词 | varchar(20) | 否 | 是 |

* 1. 开发环境与运行环境设计

本系统可以分为前端页面和后台程序两个部分，后台程序采用Flask框架，该框架基于Python且较为轻量，将网站服务器搭建，函数功能映射等操作封装起来，代码复杂度大大降低，前端页面主要由HTML、JavaScript等前端语言开发，并且使用了bootstrap框架，使页面看上去更加干净简洁。

开发环境设计：

1. 操作系统：Windows7及以上
2. 语言环境：Python 3.0及以上
3. IDE: PyCharm 5.0.3
4. 数据库：MySQL5.7
5. 数据库管理工具：Navicat Premium 12
6. 服务器：Flask调试服务器
7. 浏览器：Chrome
8. 版本控制工具：Git
9. 绘图工具：Visio

由于本系统是基于Python编写，Python作为一种跨平台的脚本语言，可以在不同的操作系统环境下运行，兼容性很好，这也是选择Python作为基于招聘网站信息的数据分析与展示系统的原因之一。

运行环境设计：

1. 操作系统：Windows7及以上，Linux内核版本号2.4.0及以上
2. 语言环境：Python 3.0及以上
3. 数据库：MySQL
4. 硬件支持：2G运行内存，2.00GHz主频CPU配置的计算机
5. 详细设计与实现
   1. 系统首页设计与实现

由于本系统的定位为信息展示网站，给用户的第一反应是很重要的，简洁明了、直观大气是首要的要求，故而在系统首页中将展示从招聘网站中爬取到的按职业技能分类的整体分布信息，系统首页设计如图4.1所示。

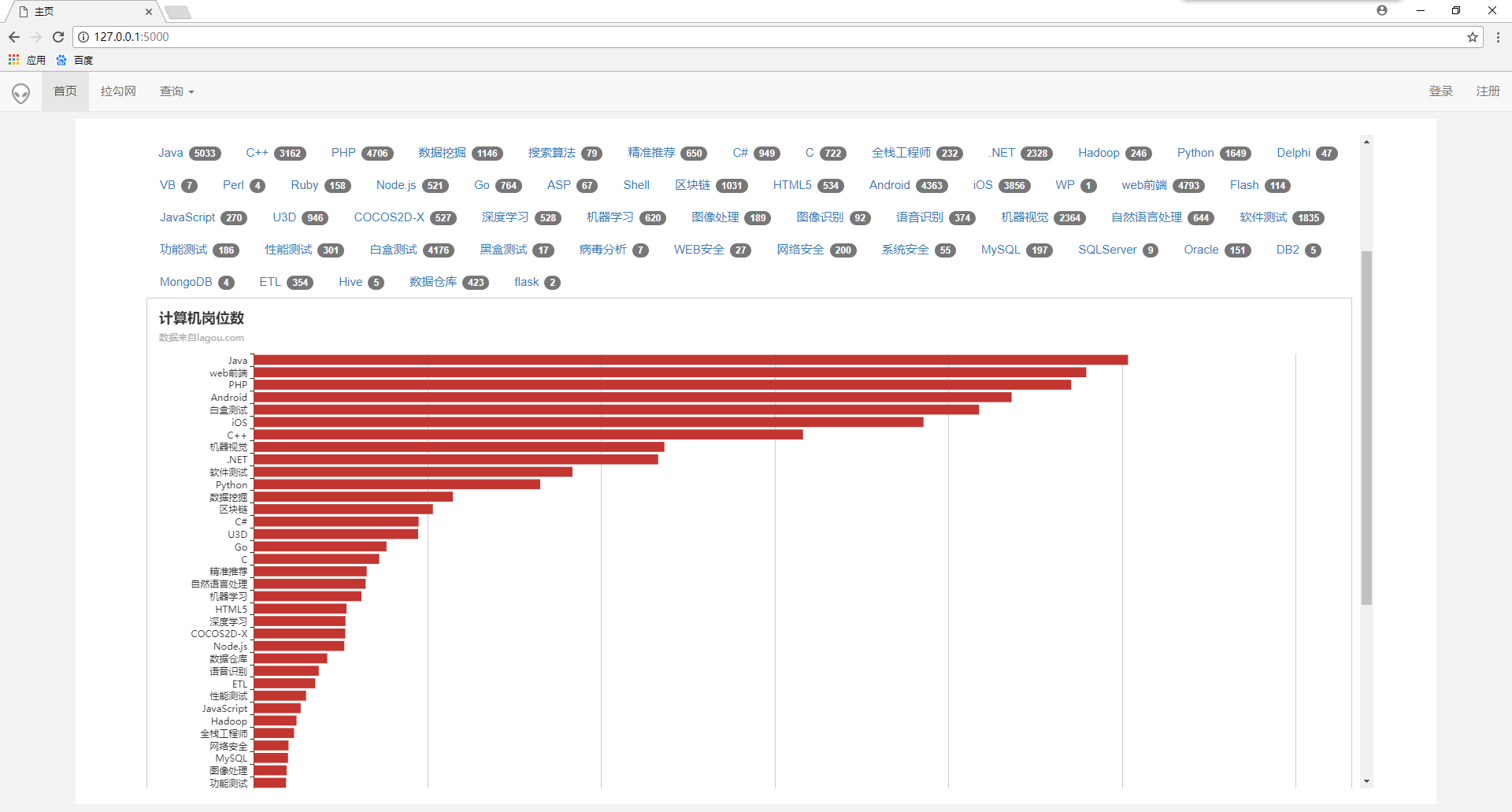


图4.1 系统首页设计

实现过程：在进入首页时，前台页面会向后台发送POST请求，查询数据库中的关键词表，并将关键词名和关键词数这两个字段的以json格式的数据返回给前端，前端获得数据后使用JavaScript插件Echarts进行图形的绘制。

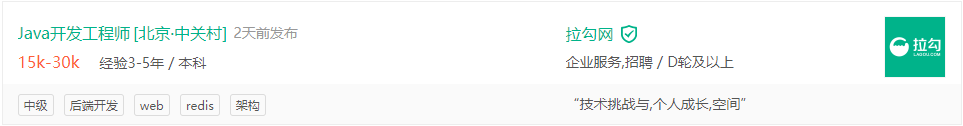
* 1. 拉勾网爬虫模块设计与实现
     1. 反爬虫机制介绍

所谓爬虫程序，从本质上来说就是利用计算机程序浏览网页并保存网页信息的过程。相较于人类浏览网页，爬虫程序拥有更快的访问速度、更高效的批处理功能，可以使数据获取的速度大大提升。但对于目标网站而言，过多的爬虫程序会提高其服务器负载，降低网页访问速度，并且爬虫的速度远快于人类，影响真实用户的体验，并且爬虫程序对给网站不会带来任何收益，所以一般大型网站都会设有反爬虫机制，例如通过验证码反爬虫，通过请求头信息反爬虫等等，而招聘网站也是如此，因此需要对目标网站进行分析，用合适的方法突破反爬虫机制，写出满足条件的爬虫程序。

* + 1. 爬虫实现过程

1. 目标网站分析（以拉勾网为例）

进入拉勾网后，输入Java搜索可以看到类似如下格式多条岗位信息：



在这条岗位信息中就可以看到一个岗位的基本信息也是最重要的信息，如岗位名称，薪资等等。按F12打开浏览器控制台，选择NetWork选项卡，捕获到拉勾网对于搜索条件的请求数据列表如图4.2所示。

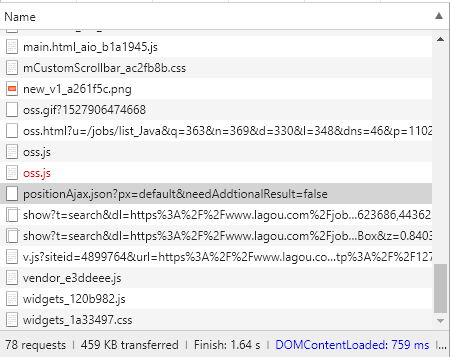


图4.2 拉勾网查询岗位的请求列表

利用搜索功能搜索图4.2中的关键字可以在异步请求的数据中找到如图4.3所示数据。

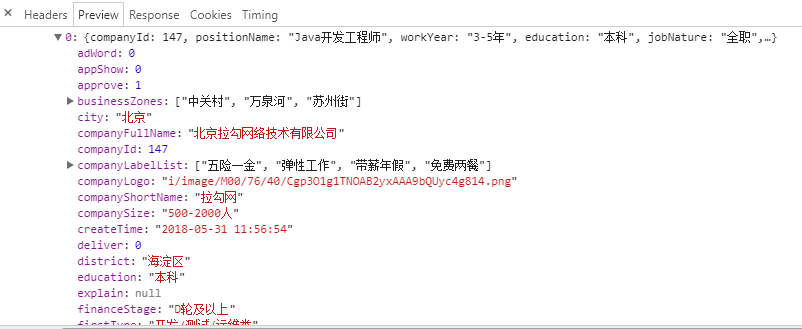


图4.3 包含岗位数据的请求信息

由此可知，此条请求就是拉勾网前端从后台获取数据的请求，查看此条请求的其他内容可以看到页面中的其他岗位的信息，如图4.4所示。

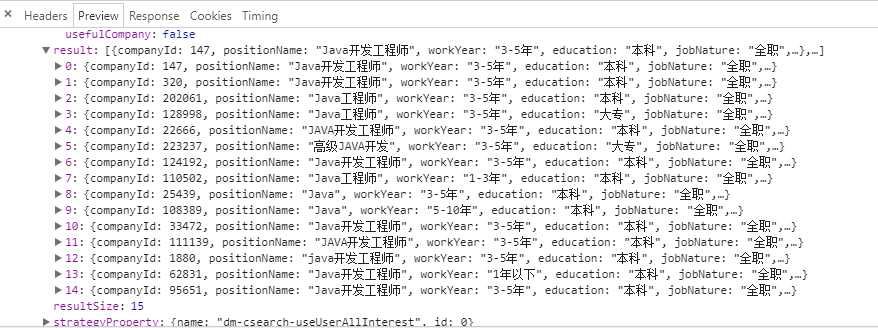
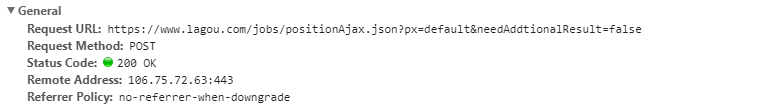


图4.4 包含页面中所有岗位数据的请求信息

由此可以分析，通过这条请求，就可以获取到拉勾网的岗位信息，查看此条信息的Headers部分，具体内容如图4.5所示。





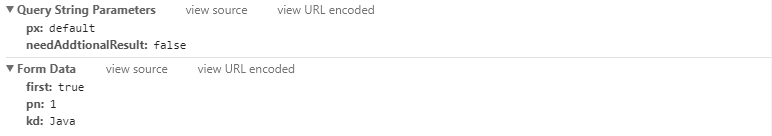


图4.5 请求头部分内容

通过Headers部分中的信息，可以得知请求数据的目标URL（Request URL），请求方式（Request Method），Cookie值，用户代理（User-Agent）以及请求的参数列表（Form Data，其中pn为页码，kd为查询词），至此，目标网站分析结束。

1. Python爬虫编写

网络爬虫的工作流程如图4.6所示。

图4.6 网络爬虫工作流程图

通过对Python官方文档的查阅，可以知道Python内置的requests对象能够以URL为参数向目标URL发送GET或POST请求，并将请求结果保存到Response对象中，再将Response对象中的HTML文本进行解析，就可以得到我们想要的数据。

爬虫伪代码如下：

for 循环多次{

修改请求头；

修改参数；

Response = request.post(目标URL，请求头，参数)；

HTML文本 = Response.text；

Json数据 = 解析函数解析HTML文本；

保存数据；

设置延时；

}

涉及到的函数如表4.7所示：

表4.7 爬虫程序相关函数API表

|  |  |  |
| --- | --- | --- |
| 函数名 | 所属模块 | 说明 |
| post() | requests | 向目标URL发送post请求 |
| get() | requests | 向目标URL发送get请求 |
| loads() | json | 将字符串格式转为json格式 |
| sleep() | time | 进程休眠 |

1. 突破反爬虫机制

由于拉勾网查看岗位不需要用户进行登录，所以爬虫程序不需要进行模拟登录，但经过测试，不对爬虫进行任何伪装会被招聘网站服务器识别为爬虫程序从而返回没有意义的数据，因此我们要将爬虫伪装成一个普通用户。

最常用的方法是通过修改HTTP协议的request请求中headers的内容来欺骗目标服务器，将本地服务器发送post请求时返回的headers内容封装起来，设置用户代理池（UA-pool）和IP代理池（IP-pool）并设置访问延时，通过post请求做为参数发送，达到模拟不同IP地址用户在不同客户端向招聘网站服务器发送请求的目的。

通过以上步骤可以编写出招聘网站爬虫程序，具体运行效果如图4.8所示：

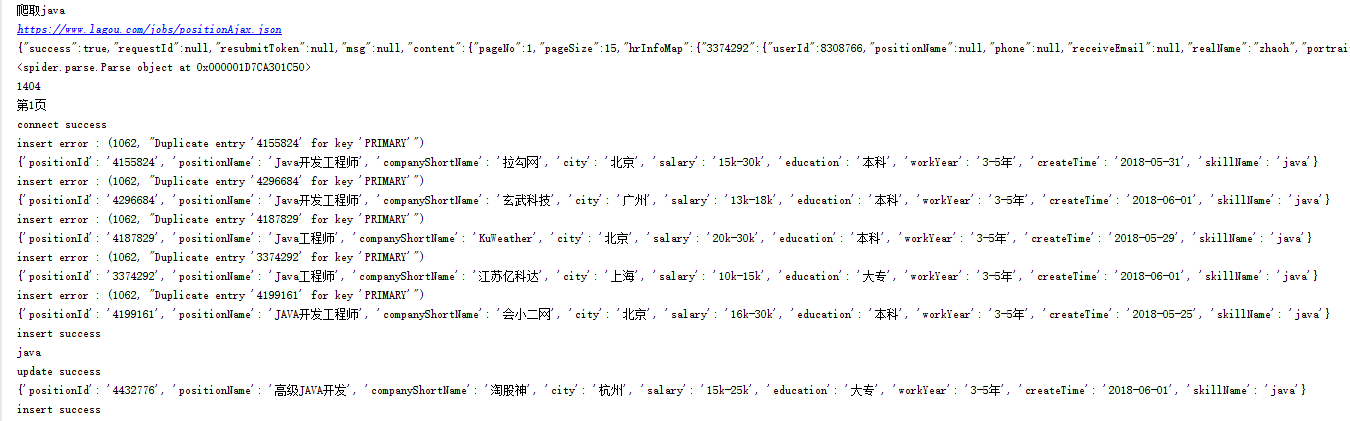


图4.8 招聘网站爬虫程序运行效果图

* 1. 信息查询与展示模块设计与实现
     1. 简单信息查询与展示

本功能适合用户简单针对某一关键字进行查询岗位，不进行其他条件过滤，用户在前台输入要查询的关键词，提交给后台，后台通过关键词对数据库中的岗位信息进行比对匹配，并将查询结果分页返回给前台页面，登录后会在每个岗位后新增收藏按钮，点击岗位编号可以跳转到拉勾网中的对应岗位上。简单信息查询Java关键词的结果界面如图4.9所示。



图4.9 简单信息查询结果界面

* + 1. 简单统计信息查询与展示

本功能将通过用户输入的关键字对岗位信息进行检索，并按照岗位所在的城市，薪资和学历三个比较重要的信息进行分类统计，将统计数据以json格式返回给前台页面，前台页面获取到后通过前端JavaScript插件Echarts进行数据饼图的绘制，来直观的反映岗位分布情况，让用户更好的了解当前市场行业岗位分布。图4.10至4.12为查询机器学习后展示的信息。

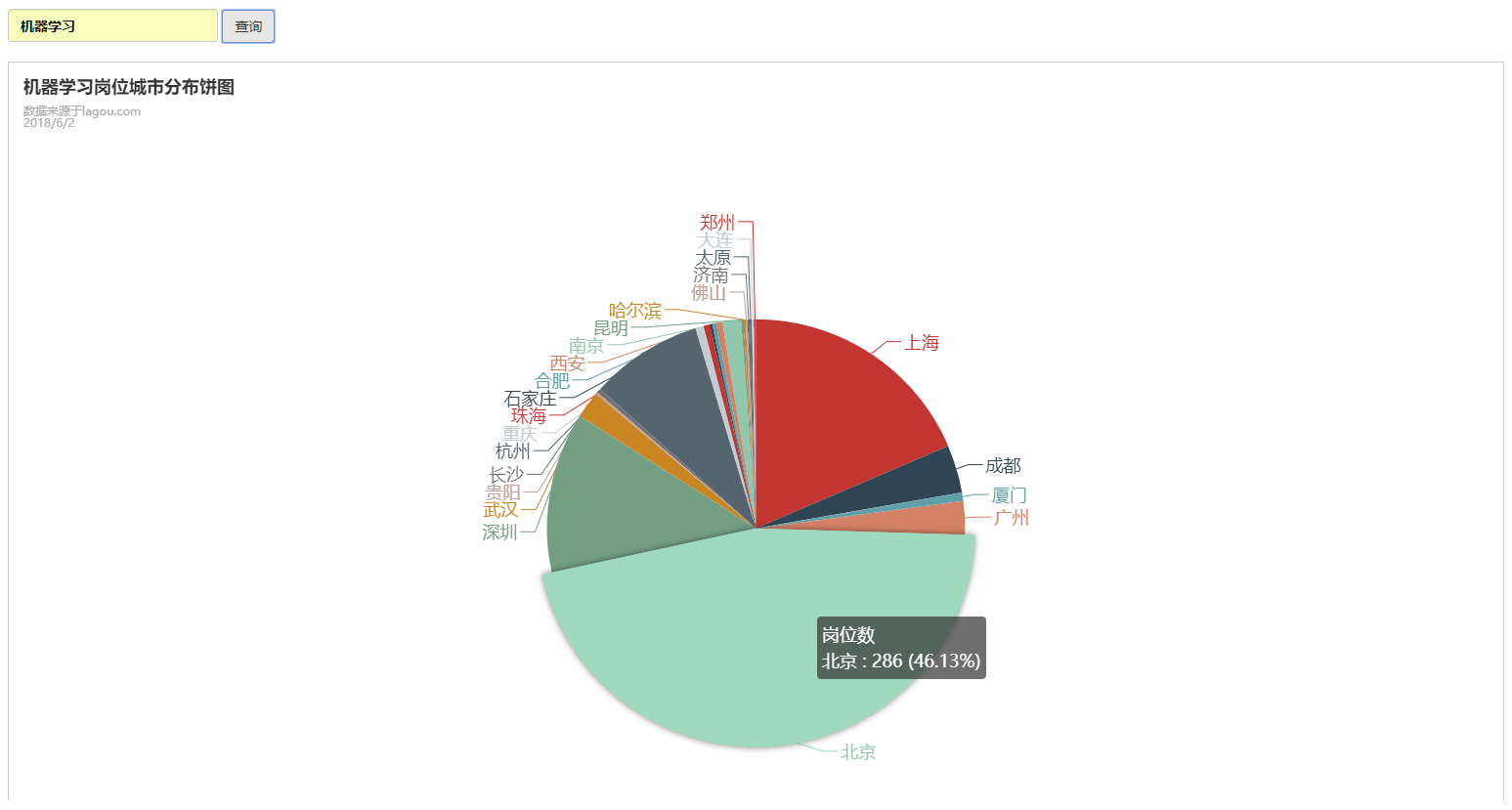


图4.10 机器学习岗位城市分布饼图

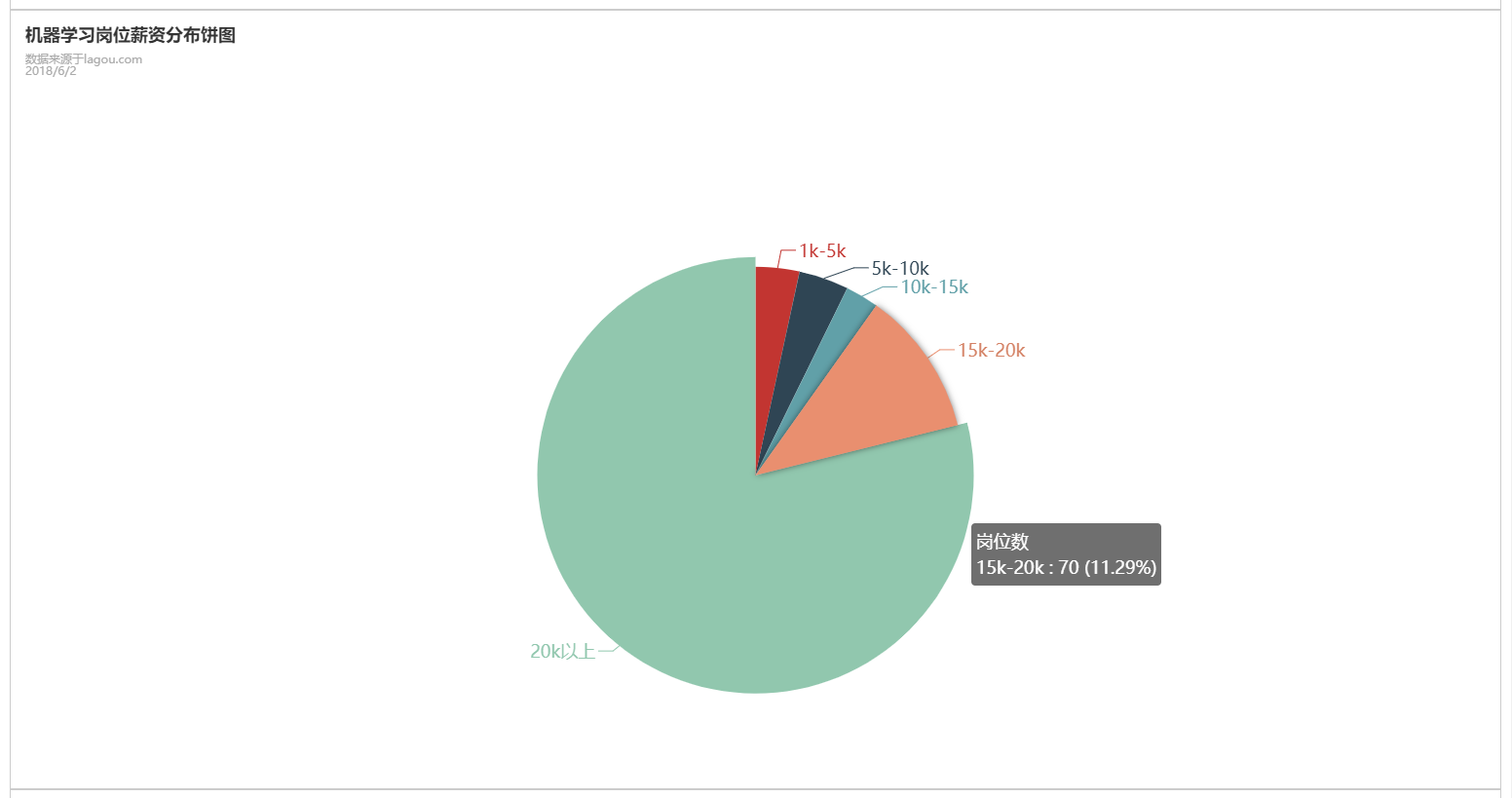


图4.11 机器学习岗位薪资分布饼图

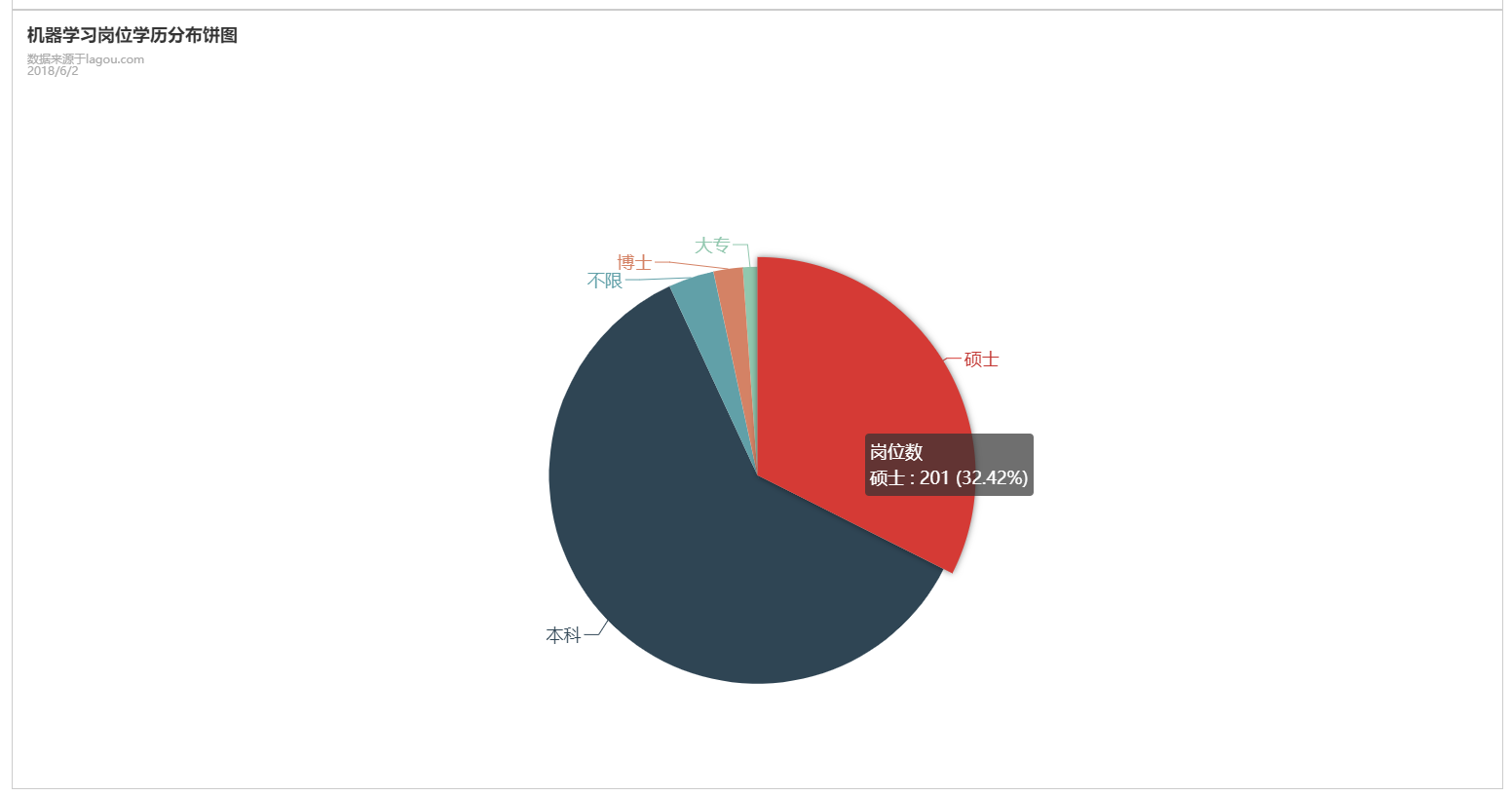


图4.12 机器学习岗位学历分布饼图

* + 1. 精确信息查询与展示

本功能在简单查询的基础上，模仿了拉勾网查询岗位的功能，提供了更多条件查询的选项，用户可以通过城市选择器（如图4.13所示）选择岗位所在的城市，通过时间选择器（如图4.14）选择岗位发布的时间范围和一些其他的选择器，过滤出符合条件的岗位信息，并且可以通过点击岗位编号直接访问拉勾网中的该岗位页面，并且用户登录后也能对每个岗位信息进行收藏，精确信息查询（关键词：机器学习，地点：北京，月薪：不限，学历：本科，工作经验：3-5年，发布日期：2018-03-01至2018-06-01）的查询结果页面如图4.15所示。



图4.13 城市选择器

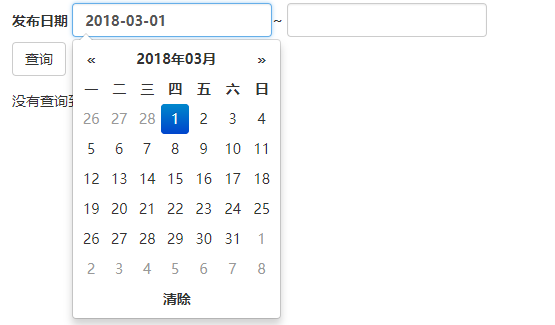


图4.14 日期选择器

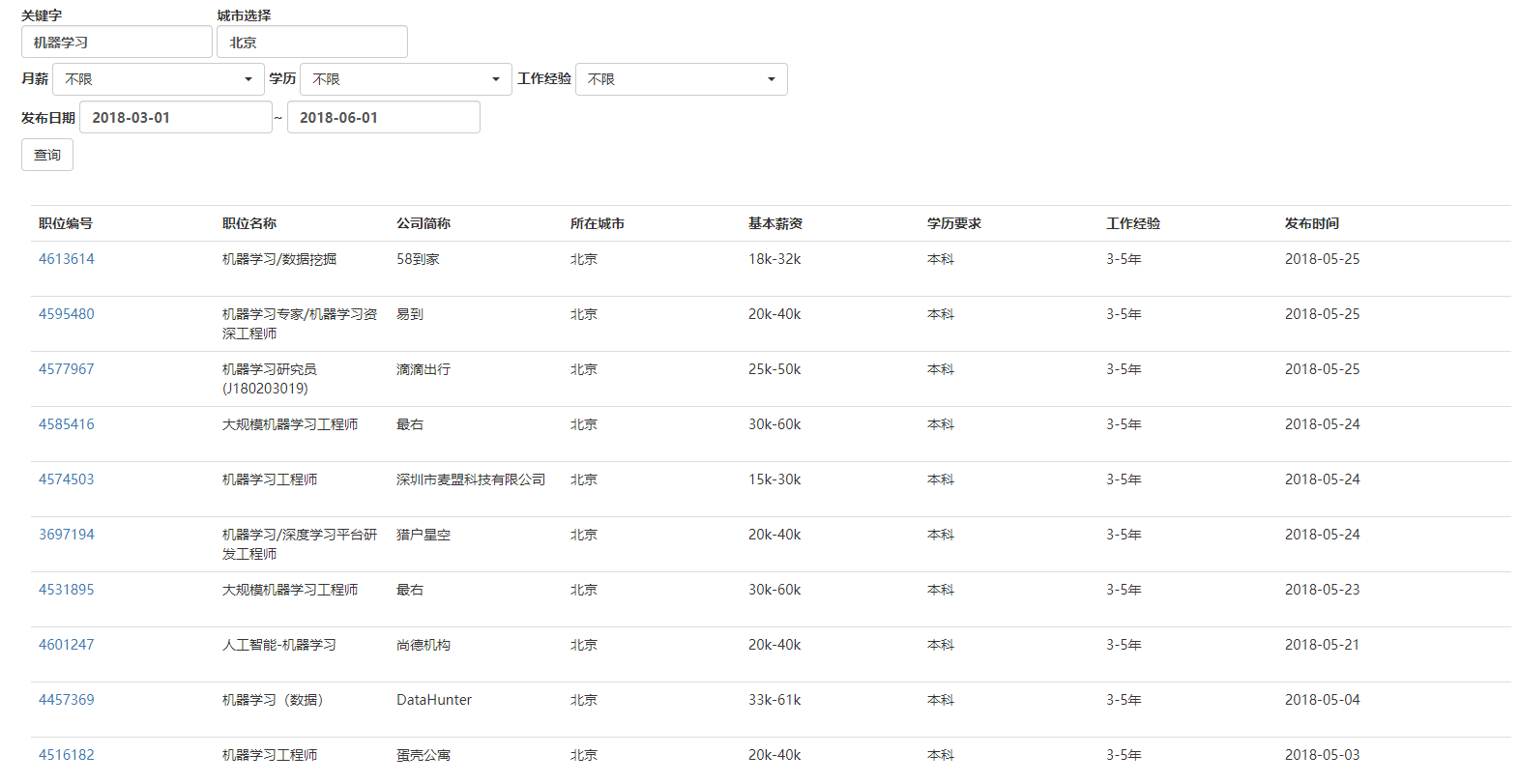


图4.15 精确信息查询结果界面

* + 1. 精确统计信息查询与展示

本功能基于精确信息查询与简单统计信息查询，为用户提供足够的自由度，让其查看自己想看的统计信息结果，提供多条件过滤，可显示不同技能、不同城市、不同薪资、不同学历和不同工作经验的岗位分布图，还提供饼图与柱状图、折线图相互切换观察的功能，充分满足用户的不同需求，更好的查看满足用户条件的岗位情况。精确统计信息查询结果（关键词：机器学习，地点：北京，薪资：不限，学历：不限，工作经验：不限，图表信息：薪资，显示条件外岗位数：否）如图4.16至图4.18所示：

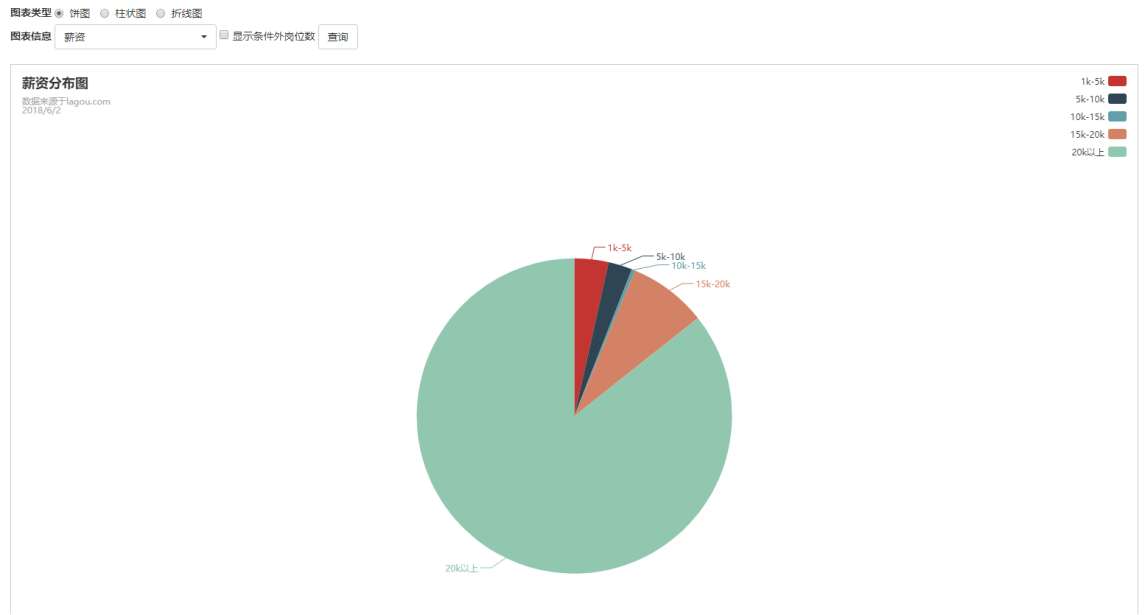


图4.16 精确统计信息查询结果（饼图）

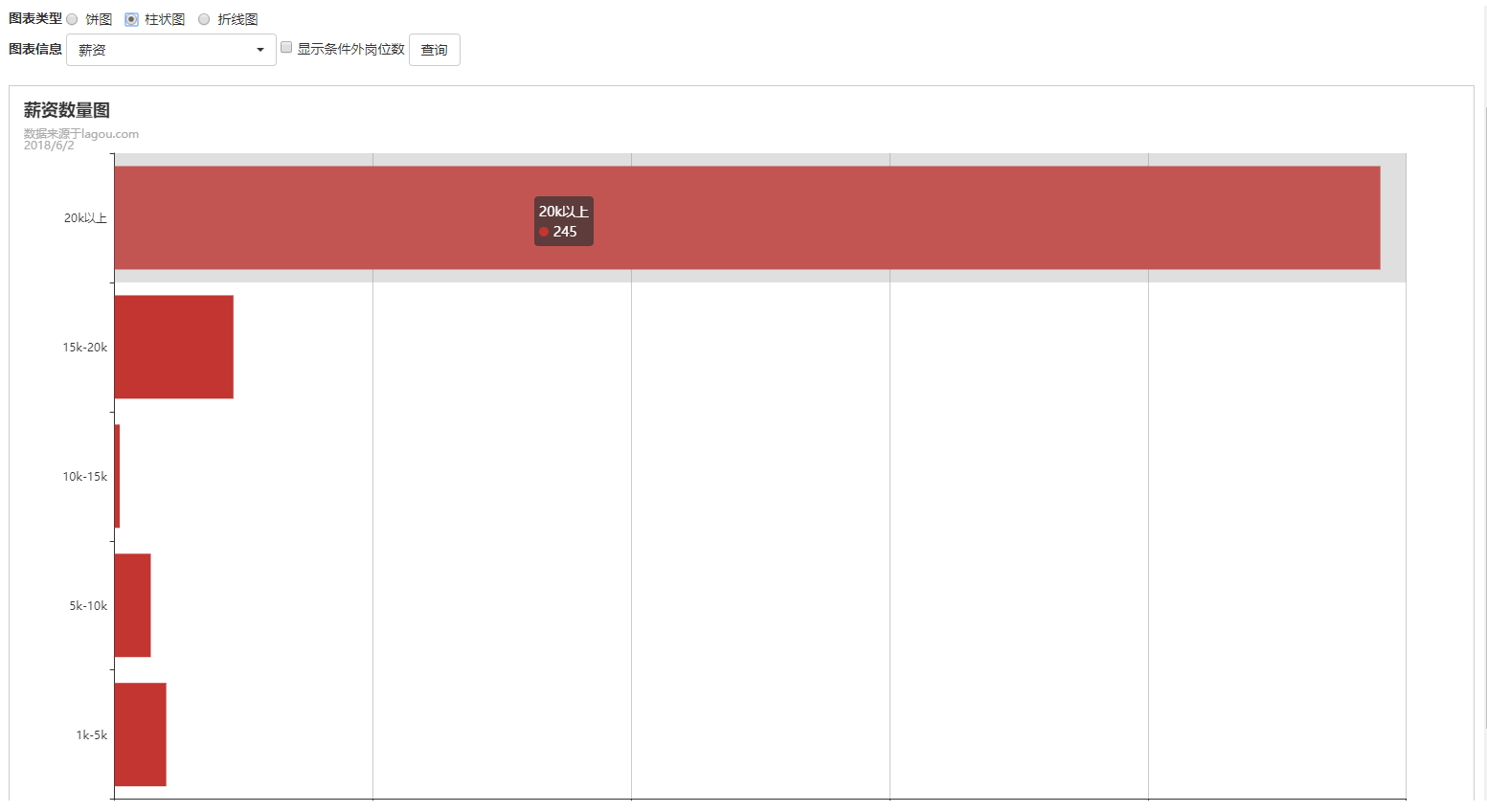


图4.17 精确统计信息查询结果（柱状图）

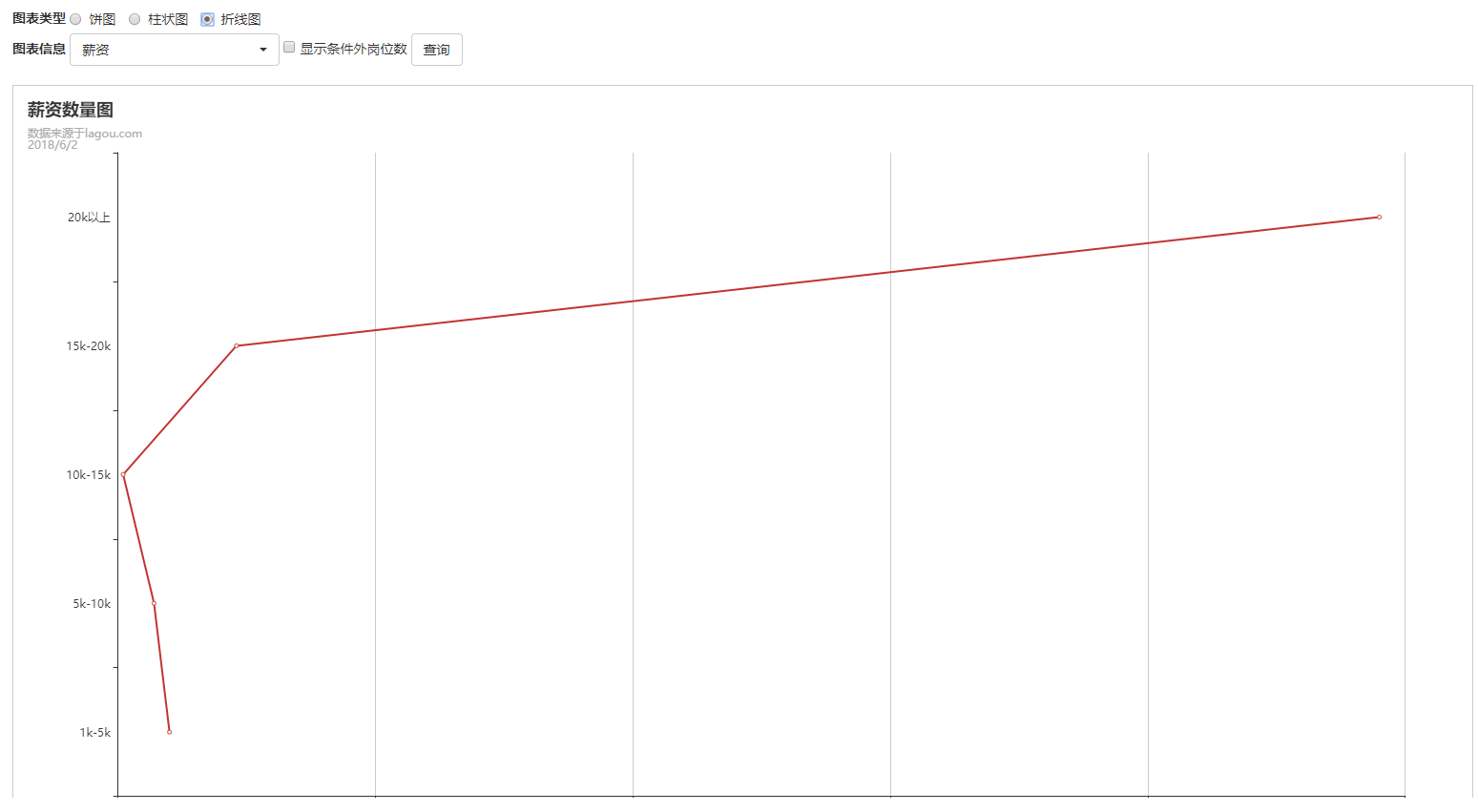


图4.18 精确统计信息查询结果（折线图）

* 1. 用户登录注册模块设计与实现

登录注册功能属于一般系统的常见功能，本系统的登录注册流程如图4.20所示。在本系统中，为了能让用户将感兴趣的岗位和统计信息收藏起来以便日后查看，故而增加了用户登录注册模块，同时为区别普通用户与管理员，设置权限属性，让不同类型的使用者使用不同的系统功能，登录界面与注册界面分别如图4.21与图4.22所示。



图4.20 用户登录注册系统流程图



图4.21 用户登录界面

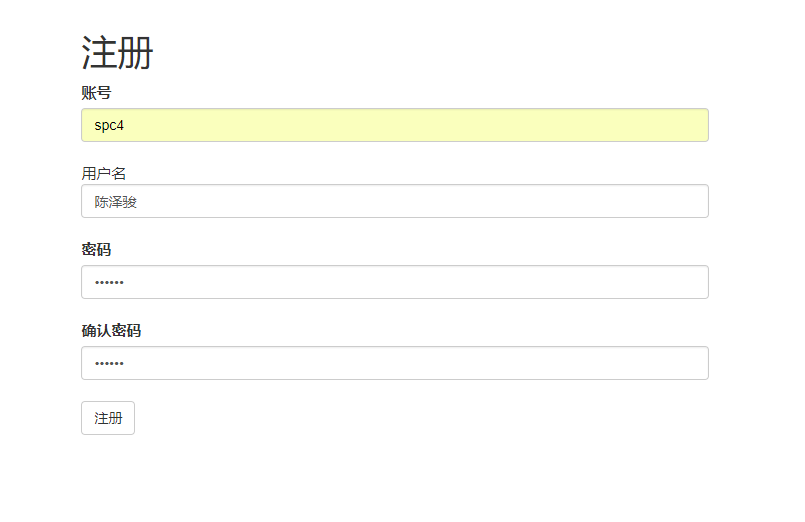


图4.22 用户注册界面

* 1. 用户个人收藏模块设计与实现

用户在登录后，再次查询岗位信息和统计信息时会带有收藏按钮提供给用户收藏此条信息的功能，点击后会将该条信息保存到“我的收藏”中，并弹窗提示收藏成功，如果该条信息已被收藏，则会提示已经收藏过了，收藏操作过程如图4.23所示。在系统界面右上角“功能”选项卡中点击“我的收藏”，即可进入收藏界面，如图4.24所示。在个人收藏中可以对收藏的统计信息进行查看，也可以将收藏过的信息删除。

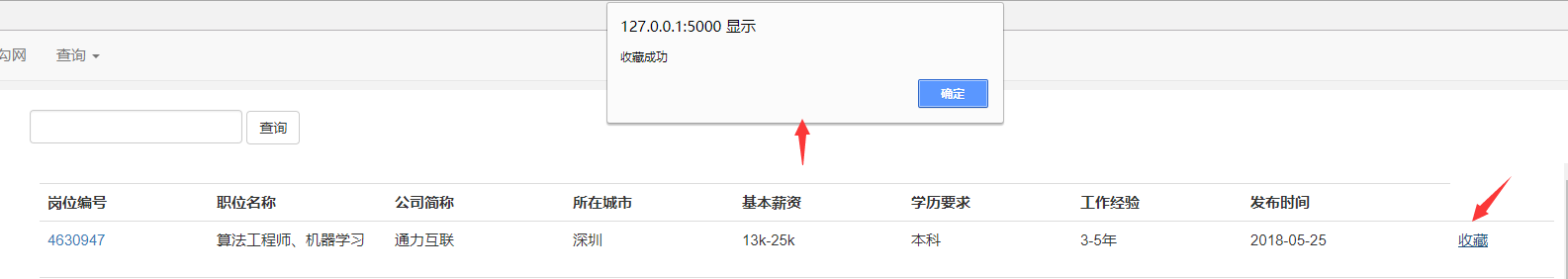


图4.23 收藏过程图

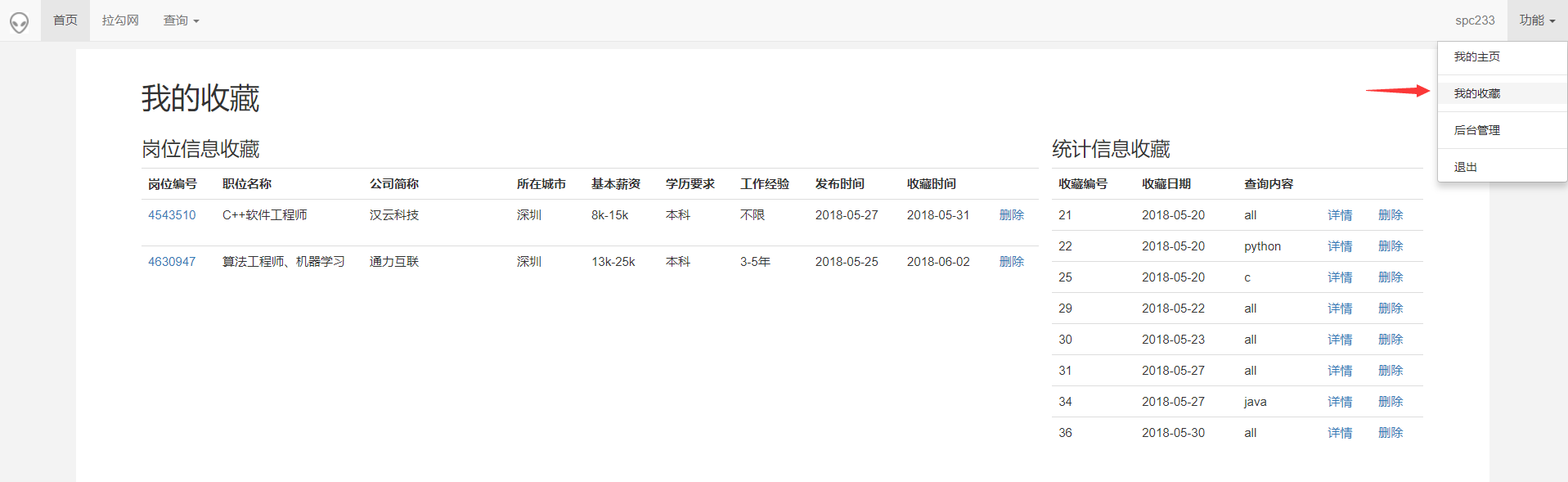


图4.24 我的收藏界面

* 1. 用户个人主页模块设计与实现

本功能为个性化功能，用户可以选择性的对其中内容进行编辑填写，修改个人信息，除了账号密码昵称以外并不强制填写，只是希望能通过此功能系统可以收集到用户的个人信息，对用户年龄，所处行业，职业等特征明显的信息进行采集，以便日后进行统计分析，个人主页界面如图4.25所示。



图4.25 我的主页界面

* 1. 后台管理模块设计与实现
     1. 用户管理

本功能为管理员可见，用于管理普通用户，管理员只可以对用户的用户名和密码进行修改，但无法修改用户的账号和权限等信息，以防系统管理员太多，账号冲突等情况的发生，对于不正当账号可以进行删除，用户管理界面如图4.26所示。



图4.26 用户管理界面（用户列表）

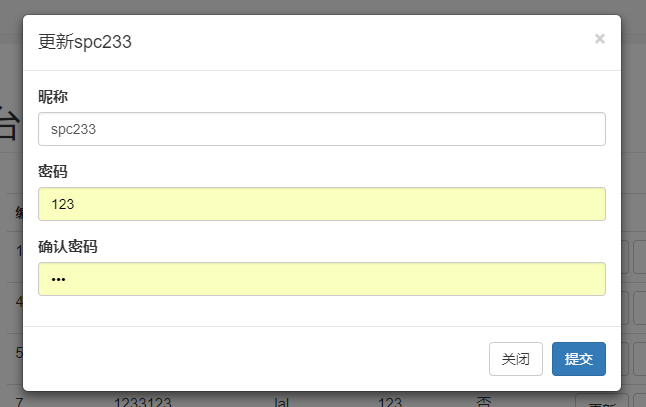


图4.27 用户管理界面（更新信息）

* + 1. 招聘统计信息管理

本功能用于管理数据库中的岗位信息与统计信息，为了确保岗位信息的真实性与准确性，管理员无法通过此功能向数据库中增加岗位信息和随意修改统计信息，只能通过爬虫程序向后台数据库新增岗位信息。但由于招聘信息具有时效性，过期的招聘信息或已经失效的招聘信息需要定期从数据库中清除，以免造成数据库过于臃肿庞大，降低查询效率，所以管理员可以对岗位进行删除操作，并修改不合理的岗位信息。图4.28与图4.29为统计信息管理模块两个部分的页面。



图4.28 统计管理页面（统计列表）

图4.29 统计管理页面（职位列表）

* + 1. 爬虫管理

本功能用于对招聘网站爬虫进行操作与管理，通过前台输入参数传递到后台爬虫程序来对招聘网站信息进行爬取，由于是利用底层代码针对拉勾网编写的爬虫程序，故而可以做到可以控制其爬取范围与爬取页面，这种操作方式更加人性化。由于目前只是针对拉勾网编写了爬虫程序，故而目标URL限定为拉勾网的URL，完善爬虫程序后会加入其它招聘网站的URL获取更多的数据，爬虫管理的界面如图4.30所示，右部为爬取岗位信息的关键词列表，需要通过这些关键词进行数据的爬取。



图4.30 爬虫管理界面（爬取数据）

1. 用户手册
   1. 系统功能及运行环境简介

对于普通用户，本系统有如下四个功能：

1. 主页总览：在系统主页中可以对看到数据库中整体岗位数量的展示，对不同职业技能的岗位数有一个清楚直观的认识。
2. 信息查询：本功能主要是检索数据库中的岗位，并将统计结果以图表的形式展示出来，包括简单信息查询，简单信息统计查询，精确信息查询，精确信息统计查询。
3. 个人收藏：本功能主要是将查询到的岗位信息和展示的统计信息图表保存到个人收藏中，以便日后查看。
4. 个人主页：显示个人信息，并可以在此个人信息进行修改。

对于管理员用户，除了上述功能以外，还可以使用以下功能：

1. 用户管理：此功能需要有系统管理员权限才能访问，主要是对注册用户的账号信息进行管理。
2. 统计管理：此功能是对从招聘网站中爬取的岗位信息以及统计信息进行管理，由于岗位信息均为从招聘网站中爬取得到，故而需要进行一定的数据清洗来保证有效信息，此功能只能由管理员使用。
3. 爬虫管理：此功能可以通过输入的URL和相应的参数爬取目标URL的岗位信息保存到数据库，通过前台页面简化了管理员的操作，通过ajax异步请求使得爬虫程序异步进行。
   1. 系统运行与操作指南
   2. Windows与Linux用户需要拥有Python3.0以上语言环境以及MySQL数据库。打开控制台，进入本系统项目文件夹下，输入以下命令：

pip freeze > requirements.txt

等待系统安装所需的第三方模块。

* 1. 在控制台中按顺序输入以下命令：

python manage.py db init

python manage.py db migrate

python manage.py db upgrade

通过flask-migrate模块完成由对象模型到迁移文件再到数据表的迁移过程，成功后数据库创建完毕。

* 1. 在控制台中进入bishe.py所在目录文件夹下，输入以下命令运行服务器：

python bishe.py

此时服务器已经运行，在浏览器中输入：localhost:5000即可进入基于招聘网站信息的数据分析与展示系统的网站首页并开始使用。

1. 系统评价
   1. 系统主要的优点和特色介绍

（1）本系统经过了详细认真的需求分析，在市场中还没有出现同类型网站，用户定位明确，能够抓住用户的痛点，给他们展示出想要的信息。

（2）本系统实现了针对拉勾网的网络爬虫，确保了本系统中展示的数据是真实有效的，并且可以一直更新数据库中的岗位数据，展示给用户近期最具时效性的信息。

（3）本系统在前端显示数据上进行了前后台的分离，只是给前端提供了一个获取数据的接口，降低了系统的耦合度。

（4）本系统使用了最新的Echarts JavaScript插件，实现了图形的动态绘制，并且更加美观，符合现在的用户审美。

（5）本系统在很多前后台数据交互上使用了ajax异步传输技术，通过异步请求获取数据使得网站加载更顺畅，用户体验更好。

（6）本系统基于Python语言开发，是一门比较新潮的技术，拥有众多第三方模块，使得本系统的可扩展性非常良好，在未来后续开发中能方便的不断加入新的功能模块。

* 1. 系统存在的不足与改进方案
     1. 系统不足

（1）本系统中的爬虫程序还不够完善，只能针对拉勾网，需要添加对其他网站的支持，且爬取信息的效率较低。

（2）在爬虫程序运行一段时间后，数据库中存储的数据量有些庞大，虽然通过sqlalchemy优化了查询速度并且为position表添加了索引，但在要遍历整个表的情况下，还是会有数据加载偏慢的情况出现。

（3）虽然只是信息展示网站，但对普通用户而言功能还是较少。

（4）网站界面不够精美，使用的仅仅是bootstrap提供的官方样例，在界面的美观方面还是需要下点功夫。

* + 1. 改进方案

（1）对其他招聘网站进行同样细致的分析，找出可以获取数据的URL，并编写对应的爬虫程序，并且要对反反爬虫技术进行深入研究，找到更有效的方法伪装爬虫，更高效的爬取数据。

（2）要及时清理数据库，删除冗余数据和失效数据，对数据表的设计进行进一步优化，同时也要对数据返回方式进行优化，让更少的数据传输到前台就能画出同样效果的图形。

（3）多研究研究招聘网站，针对想要使用本系统的用户定制一些其他的功能，比如开辟一个岗位讨论论坛，查看他人收藏岗位等功能，增加用户之间的交互，使系统功能更加多样化。

（4）加强对前端的学习，模仿优秀网站的界面设计。

1. 毕业设计心得与收获

为期三个月的毕业设计以接近尾声，在大学的最后一个学期里，为了完成毕业设计，利用Python写出一个不错的网站，于是我便从零开始自学Python的语法，常用API，Flask框架的用法，学习如何利用Flask搭建一个网站，怎样使用Flask的模板，如何将模板与路由函数映射，如何做到前后台的参数传递等等，我还认识到了平时不够重视的前端语言JavaScript是有多么的强大，可以完成很多在其他网站上常见的网页特效，并且我还深入了解了网络爬虫的机制，对于爬虫的理解真正的由理论上升为实践，这也是毕业设计的目的，将学习的知识与实际使用相结合。

整个毕业设计过程中，网络爬虫的实现花费了我绝大部分时间，从理论知识到初步尝试，一开始并没有了解反爬虫机制，只是简单的通过请求想要获取到招聘网站岗位列表的网页源代码，然后用xpath表达式解析出岗位信息，但结果被招聘网站的反爬虫系统发现，返回一堆没有意义的字符串，并且多次尝试后，我的IP地址也被封了，无法访问该招聘网站。于是我便开始查阅反爬虫的相关资料并学习如何越过这种机制，拿到我想要的数据，经过长时间的不断尝试，终于完成了本次课题的关键爬虫程序。

实现网络爬虫的过程让我受益良多，它让我真真切切认识到，编码能力并不是最重要的，重要的是解决问题的能力和思路。因为在实现网络爬虫的过程中，要一步一步的寻找能够得到数据的URL，以及用合适的方法取到数据，过程很艰难，但一旦解决这些困难，编码实现就相当轻松了。回想一下软件工程理论对于软件开发的一般过程，也让我明白需求分析等前面的工作做好了，后面才能有序顺利进行，才能事半功倍，这对于即将走出校园，走向工作岗位的我无疑是最大收获。、

无论是Flask亦或是爬虫，还是说JavaScript，Python，这些技术或语言都足够深奥，值得我们长时间去研究，而我在本次毕业设计中也只是触碰到其冰山一角，有更多的东西等待着我去挖掘。虽然在这条道路上会不断遇到挫败，但我已体验到了问题解决后的成就感与满足感，今后会继续前行。

结 论

基于招聘网站信息的数据分析与展示系统采用B/S架构模式，基于Python语言开发，其主要目的是为了将招聘网站上的岗位信息收集起来，进行数据统计与分析，将结果展示出来给用户查看。本系统主要针对的用户人群为大学在校生、应届毕业生和需要转岗者，通过近期的招聘网站信息了解整个市场需求，制定自己的学习方向和求职方向，根据自身条件筛选合适的岗位，并了解岗位分布情况，通过本系统的信息能够树立一个正确的目标。

本系统的主要问题在于爬虫的实现，在爬取到了大量岗位信息的基础上，进行展示网站的设计，为了增加用户体验，除了可以输入搜索关键词将结果信息展示以外，还增加了收藏功能，增加用户粘度。为了今后进行进一步统计分析，在本系统中增加了个人主页功能，但此功能还不够完善，这也是本系统的缺陷之一。本系统能够查询到岗位信息并将统计结果以图表的形式展示出来，提供饼图，柱状图，条形图的切换，多种数据图表的展示能从不同的角度描述统计信息。在本系统中，由于所有信息是由爬虫获取，所以管理员是一个很重要的角色，直接影响到网站的数据定位，管理员除了可以对用户和岗位进行管理，还可以设置爬虫爬取的参数，如本系统目前主要针对计算机领域的岗位进行爬取，关键词均与计算机有关，所以本系统主要展示了计算机领域的相关岗位信息。而其他领域可以通过管理员修改关键词来爬取其他行业的岗位信息，展示其他行业的岗位分布情况。

目前互联网上没有的招聘网站信息统计结果可供查看，本系统填补了这一空缺，并针对就业这一尖锐的社会问题提出了解决方案，即将市场的需求展示给人们看，让其清楚自身定位，找到合适的工作，所以本系统具有很高的开发潜力。

致谢

当全身心投入某一件事的时候，时间总是过得很快，如白驹过隙，转瞬即逝，毕业设计临近尾声，大学生活也是如此。通过这次毕业设计，让我的学习能力得到了长足的进步，从对Python的一知半解，到能熟练编写Python程序再到对第三方模块的熟练使用以及利用Python编写网络爬虫。其中，最重要的是我的指导老师，他在我的Python学习道路上起着引路人的作用，让我接触到Python并对我进行耐心的指导，同时在毕业设计的过程中给了我很多可靠的建议，这里我想给纪老师最真挚的感谢，这个系统的完成离不开纪老师的指导。通过自己的努力和老师的指导，才能让我进步的如此之快。

其次，我的大学室友们同样给我提供了很多帮助，我同样对他们抱有感激之情。一个良好的学习环境能够大大提高学习效率和工作效率，在学习道路上，我们一起努力一起奋斗，虽然大家的毕业设计选题互不相同，但大家都尽其所能相互提出建立，进行讨论，研究如何将系统做的更完善，如何将论文语句修改的更专业化，这些点点滴滴积累的帮助潜移默化的影响着我，鼓励着我，最终完成了本次毕业设计的课题。

我还要感谢我们学院给予的帮助和支持。学院在这次毕业设计中给我们学子提供了良好的编码环境，有专门的机房提供给我们编写代码，并让指导老师参与指导，这无疑是对我们最大的帮助。同样支持我的还有我的父母，虽然远在外地读书，但他们会时常带电话给我关怀，在我遇到困难停滞不前时，对我进行开导，疏解我内心的烦躁，让我能坚持下去用心完成毕业设计，感谢他们作为我的后盾与支柱一直在我的后方支持着我，鼓励着我。

大学四年将圆满结束，有太多不舍，太多怀念，感谢这四年的时光，感谢母校对我的培养，让我学到很多人生道理，逐渐由幼稚走向成熟，让我顺利走向社会。

参考文献

[1] Eric Matthes. Python编程从入门到实践[M].北京：人民邮电出版社，2016.

[2] Richard Lawson. 用Python写网络爬虫[M].北京：人民邮电出版社，2016.

[3] Ryan Mitchell. Python网络数据采集[M].北京：人民邮电出版社，2016.

[4] 胡松涛. Python网络爬虫实战[M].北京：清华大学出版社，2017.

[5] Miguel Grinberg. Flask Web开发基于Python的Web应用开发实战[M]. 北京：人民邮电出版社，2015.

[6] 张若愚. Python科学计算[M]. 北京：清华大学出版社，2016.

[7] Luciano Ramalho. 流畅的Python[M]. 北京：人民邮电出版社，2017.

[8] 刘增杰. MySQL 5.7从入门到精通[M]. 北京：清华大学出版社，2016.

[9] 王珊. 数据库系统概论[M]. 北京：高等教育出版社，2014.

[10] 张海藩. 软件工程导论第六版[M]. 北京：清华大学出版社，2013.

[11] Jie Bao. Incorporating twitter-based human activity information in spatial analysis of MARK crashes in urban areas [J].Nanjing：Southeast University，2017：17-19.

[12] W. Gins. Analysis of counting data: Development of the SATLAS Python package [J]. United Kingdom：The University of Manchester，2017：286–294.

[13] 于娟. 主题网络爬虫研究综述[J].福建：福州大学，2015：1007-130X.

[14] 李玲. 基于B/S的毕业设计管理系统设计与实现[J].青岛：青岛理工大学，2014：55-57.