

# 上海第二工业大学

## 本科毕业设计（论文）

### 学生手册

题目：基于协同过滤的互联网岗位推荐系统  
的设计与实现

英文题目：DESIGN AND IMPLEMENTATION OF INTERNET  
JOB RECOMMENDATION SYSTEM BASED ON  
COLLABORATIVE FILTERING

学号：20201111012

姓名：王耀彬

班级：20 计科 C2

专业：计算机科学与技术

学部（院）：计算机与信息工程学院

入学时间：2020 级

指导教师：石林祥

开始日期：2023 年 11 月 15 日

完成日期：2024 年 05 月 15 日

(1 起始日期由毕业设计（论文）实际开始日期算起（包括选题与预研阶段）。)

(2 各学部（院）可根据具体情况进行修订后报教务处备案)

## 目 录

- 一、任务书
- 二、开题报告
- 三、中期检查表
- 四、指导记录表
- 五、查重报告（简洁版）
- 六、成绩评定表

毕业设计（论文）任务书

题 目	基于协同过滤的互联网岗位推荐系统的设计与实现				
学部（院）	计算机与信息工程学院	专业	计算机科学与技术 C		
班级	20 计科 C2	姓名	王耀彬	学号	20201111012
一、课题主要研究(设计)内容： <p>随着深度神经网络的不断发展，协同过滤技术在众多领域得到了成功的应用。本系统主要使用基于协同过滤的推荐算法作为岗位的推荐模型。通过对现有互联网岗位招聘网站爬虫并进行数据整理与综合得到岗位数据，利用 Web 作为数据的载体，利用协同过滤技术对用户做个性化岗位推荐，最终实现基于协同过滤的互联网岗位推荐系统。</p> <p>基于协同过滤的互联网招聘岗位推荐系统的设计与实现的主要研究内容如下：</p> <p>1. 课题背景与意义。分析课题研究的背景与意义，结合国内外研究现状，说明课题实现系统的重要性。需求分析，根据安全性需求、性能性需求等需求进行需求分析最终确认系统需实现的全部功能。</p> <p>2. 数据获取爬虫模块。使用 Selenium、Scrapy 和 CrawlSpider 等爬虫工具对现有互联网招聘网站例如牛客、力扣与 BOSS 直聘等进行数据爬取，对数据进行综合处理，得到全面丰富且详细的互联网招聘岗位数据。将数据存入阿里云 RDS MySQL 数据库。</p> <p>3. 岗位数据的显示模块。利用 Web 作为招聘岗位数据的载体，使用 Django + Vue 构建网站前后端，使用本地服务器/阿里云服务器。</p> <p>4. 基于协同过滤的推荐系统模块<sup>0</sup>。基于神经网络采用协同过滤技术实现推荐系统<sup>0</sup>。使用神经协同过滤算法(neural network-based collaborative filtering, NCF)，利用用户基本信息、用户的历史岗位浏览信息与用户收藏岗位信息，给出个性化互联网招聘岗位推荐。</p>					
二、工作进度要求：					
序号	设计（论文）各阶段名称	日期		备注	
1	完成论文课题确定，对课题进行基础分析，分析研究的背景和意义、国内外研究现状，进行需求分析。	2023 年 11 月 1 日- 2024 年 1 月 1 日			
2	完成课题前置基础知识的学习,包括:爬虫基础使用,熟悉 Django 与 Vue，熟悉协同过滤算法。	2024 年 1 月 2 日- 2024 年 1 月 15 日			
3	完成对现有互联网招聘岗位网站的信息爬取,对数据进行综合处理并存入 RDS MySQL 数据库	2024 年 1 月 16 日- 2024 年 2 月 1 日			
4	使用 Django 与 Vue 搭建 Web 前后端, 将其与 MySQL 数据库相连接测试，完成数据显示工作	2024 年 2 月 2 日- 2024 年 3 月 1 日			

5	设计并实现协同过滤推荐模型, 利用已有数据进行模型训练, 并将模型部署至服务器。	2024 年 3 月 2 日 - 2024 年 4 月 1 日	
6	对爬虫模块、Web 模块与推荐算法模块进行测试, 测试其稳定性与效率。撰写毕业论文。	2024 年 4 月 2 日 - 2024 年 5 月 10 日	
<p>三、应查阅的主要参考文献:</p> <p>[1].Jacob Jacoby.Perspectives on Information Overload[J].Journal of Consumer Research,1984(4).</p> <p>[2].Covington P,Adams J,Sargin E.Deep neural networks for youtube recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems.ACM,2016:191-198.</p> <p>[3].孙黎博.基于深度学习和矩阵分解的推荐算法分析[J].电子技术与软件工程,2022(18):218-221.</p> <p>[4].Goldberg D,Nichols D,Oki B M,et al.Using collaborative filtering to weave an information tapestry[J].Communications of the ACM,1992,35(12):61-71.</p> <p>[5].Ghuli P,Ghosh A,Shettar R.A collaborative filtering recommendation engine in a distributed environment[C]//2014 International Conference on Contemporary Computing and Informatics(IC3I).IEEE,2014:568-574.</p> <p>[6].Zhang H,Yang H,Huang T,et al.DBNCF:Personalized courses recommendation system based on DBN in MOOC environment[C]//2017 International Symposium on Educational Technology(ISET).IEEE,2017:106-108.</p> <p>[7].蔡辉虎.基于深度神经网络的协同过滤推荐系统体系结构研究[J].信息记录材料,2023,24(11):236-238+242.DOI:10.16009/j.cnki.cn13-1295/tq.2023.11.011.</p> <p>[8].李建超.基于神经网络的协同过滤推荐算法研究[D].长春工业大学,2023.DOI:10.27805/d.cnki.gccgy.2023.000845.</p> <p>[9].张铁斌,郑行,邢星等.基于深度神经网络融合特征信息的协同过滤模型[J].渤海大学学报(自然科学版),2022,43(02):172-177.DOI:10.13831/j.cnki.issn.1673-0569.2022.02.005.</p> <p>[10].黄立威,江碧涛,吕守业等.基于深度学习的推荐系统研究综述[J].计算机学报,2018,41(07):1619-1647.</p> <p>四、毕业设计(论文)预期成果或结论性观点:</p> <p>实现运行稳定, 推荐高效的基于协同过滤的互联网招聘岗位推荐系统。</p> <p>系统具有良好的运行稳定性, 能过支持一定数目的用户同时访问, 在多用户并发访问时服务器可以较快速的查询数据库给出响应。</p>			

推荐算法高效且精准，区别于随机的推荐，能够依据用户的历史浏览及收藏记录给出针对性的招聘岗位推荐，且模型可以利用用户访问次数、用户指定无需推荐的岗位等信息进行动态调整。模型轻量化且推荐快捷。易于在本地部署。

五、毕业设计（论文）完成提交材料清单：（设计、作品照片、实物、模型、技术文档或论文、含有技术文档或论文的光盘等）

1.爬虫模块。爬虫 Python 源代码、存放数据的数据库映射文件。

2.Web 模块。基于 Django 的 Python 后端代码、基于 Vue 的前端代码、数据库组织存在方式、云数据库账号密码、服务器地址

3.推荐算法模块。算法实现代码，深度学习模型文件。

4 毕业论文，作为以上源代码的技术文档。

六、任务执行日期：

自 2023 年 12 月 27 日起，至 2024 年 5 月 10 日止。

学 生（签字）：王耀楠

指导教师（签字）：王树松

系 主 任（签字）：陈林

注：1、任务书一律由指导教师填写；

2、起始日期由毕业设计（论文）实际开始日期算起（包括选题与预研阶段）。

## 毕业设计（论文）开题报告

题 目	基于协同过滤的互联网岗位推荐系统的设计与实现				
学部（院）	计算机与信息工程学院	专业	计算机科学与技术 C		
班级	20 计科 C2	姓名	王耀彬	学号	20201111012
<p>一、研究目的和意义</p> <p>近些年来，随着计算机和互联网技术的快速发展，互联网已经遍布世界的每一个角落，成为人们日常学习生活中获取信息的最重要方式。在此基础上，互联网行业人才需求急剧增长，岗位招聘信息不断增加导致了“信息过载”问题<sup>0</sup>。求职者难以在浩如烟海的招聘信息中找到最适合自己的岗位，用户很容易因信息过载而迷失方向。又由于互联网求职平台的繁多庞杂，其造成的信息差导致经验不足的求职者如应届生可能难以找到令自己满意的工作。</p> <p>在此背景下，本系统旨在为求职者提供统一的、信息全面丰富且详细的互联网岗位招聘信息平台。针对互联网招聘信息平台的巨大招聘信息带来的“信息过载”问题，本系统在以往平台基于薪资和用人需求的岗位搜索、基于城市和岗位属性的分类以及基于热度的推荐的基础上，使用协同过滤技术根据用户的岗位浏览历史和收藏记录为用户提供个性化、精准化及定制化的理想岗位推荐。</p> <p>通过将用户信息、岗位信息以及用户关注信息等数据相结合使用推荐算法为不同用户提供个性化的推荐，充分利用数据，不仅可以解决用户的求职难题和“信息过载”问题，还可以提高用户的粘性和满意度。为求职者发现适合自己、满足自己需求的岗位打下基础。进一步促进应届生就业。</p>					
<p>二、国内外研究现状及其发展</p> <p>随着近些年来互联网行业的迅猛发展，社会对互联网岗位的需求急剧增加，由于互联网自身万物互联的属性，互联网岗位招聘信息在互联网上开始流通传播，在此基础上各类互联网岗位招聘网站层出不穷，具有代表性的有：牛客/力扣求职板块、BOSS 直聘、职到网和拉勾招聘等。</p> <p>面对“信息过载”问题，早在 20 世纪末期，推荐系统就受到学术界和工业界的关注和研究。随着深度学习技术在图像识别、语音识别和自然语言处理方面取得了良好的进展，许多基于深度神经网络的推荐算法应运而生。如 2016 年 Covington 等人<sup>0</sup>提出一个基于深度神经网络的推荐算法来研究深度学习在 YouTube 视频推荐系统上的影响。最终论文实验表明基于深度学习的推荐算法超越了以往在 YouTube 上使用的矩阵分解方法<sup>0</sup>。协同过滤作为最知名和使用最广泛的推荐技术之一，其通过分析当前用户的行为记录，识别与当前用</p>					

户具有相似品味的其他用户群，通过这些用户群给当前用户推荐感兴趣的物品<sup>0</sup>。近些年来，随着互联网的迅猛发展，在线用户和网络内容急剧增长，单机的协同过滤系统在面对大量数据的处理时会出现性能问题。针对性能问题，一般采用分布式框架与分布式环境，如基于 Hadoop 的大数据推荐系统<sup>0</sup>，具有较高的推荐效率。

传统的协同过滤算法存在稀疏数据问题，无法精确处理在线学习用户的高维属性，导致课程推荐效率低下。为解决这类问题，Zhang 等人<sup>0</sup>提出了一种基于深度信念网络 (Deep Belief Network, DBN) 的课程推荐系统，该系统利用 DBN 在函数逼近、特征提取、预测分类等方面的高性能，使用课程分数作为 DBN 监督学习的课程标签。通过无人监督的预训练和监督反馈微调，来实现基于 DBN 的课程推荐模型训练。实验结果表明，基于 DBN 的协同规律方法比传统的协同过滤方法更有效。

### 三、主要研究内容（提纲）

随着深度神经网络的不断发展，协同过滤在众多领域得到了成功的应用。本系统主要使用基于协同过滤的推荐算法作为岗位的推荐模型。通过对现有互联网岗位招聘网站爬虫并进行数据整理与综合得到岗位数据，利用 Web 作为数据的载体，利用协同过滤算法对用户做个性化岗位推荐，最终实现了基于协同过滤的互联网岗位推荐系统。

基于协同过滤的互联网岗位推荐系统的设计与实现的主要研究内容如下：

1. 课题的背景与意义。分析课题研究的背景与意义，结合国内外研究现状，说明课题实现系统的重要性。需求分析，根据安全性需求、性能性需求等需求进行需求分析最终确认系统需实现的全部功能。

2. 互联网招聘岗位数据的获取。使用 Selenium、Scrapy 和 CrawlSpider 等爬虫工具对现有互联网招聘网站例如牛客、力扣与 BOSS 直聘等进行数据爬取，对数据进行综合处理，得到全面丰富且详细的互联网招聘岗位数据。将数据存入阿里云 RDS MySQL 数据库。

3. 岗位数据的显示。利用 Web 作为招聘岗位数据的载体，使用 Django + Vue 构建网站前后端，使用本地服务器/阿里云服务器。

4. 基于协同过滤的推荐系统的设计与实现<sup>0</sup>。基于神经网络采用协同过滤技术实现推荐系统<sup>0</sup>。使用神经协同过滤算法(neural network-based collaborative filtering, NCF)，利用用户基本信息、用户的历史岗位浏览信息与用户收藏岗位信息，给出个性化互联网招聘岗位推荐。

### 四、拟解决的关键问题

1.爬虫的质量。爬虫得到的数据是否完善，是否具有可操作性，是否无需特殊加工可直接存入数据库并在网站中显示。如何对海量数据进行统一规格化处理。正则表达式的使用。

<p>2.数据的处理。重复数据的删除、过期数据的删除、数据的实时更新、数据的可视化准备及数据的高效存储检索等。如何快速安全的检索数据，采用何种数据库，数据库表如何设计，Web 模块与数据库如何连接。</p> <p>3.推荐模型的质量。能够高质量高效率的为用户提供符合预期的招聘岗位信息，不反复推荐，设计“此后不再推荐”功能，允许用户动态调整推荐功效。对协同过滤技术熟练掌握使用，部署轻量化快捷的模型到本地/服务器。采用何种模型，在具体的模型上针对本系统的需求调整合适的参数。</p>			
<p>五、研究思路和方法</p> <p>以实践为基础，在实践中测试掌握相关知识的熟练程度，并提前经历错误，为系统的最终实现打下基础。在每个模块的正式启用前进行模块单独的测试。</p> <p>1. 爬虫模块。在爬取互联网招聘岗位信息前，尝试性爬取网易新闻、携程酒店信息等作为练习，熟悉爬虫的流程，正则表达式的应用。</p> <p>2. Web 模块。吸取成熟后端经验，采用设计模式与软件工程设计指导方法，吸收借鉴成熟前端美术设计，利用 Django 与 Vue 框架特性，简化前后端设计。</p> <p>3. 推荐算法模块。合理规划训练集与测试集，采用数据可视化等方法对设计实现的模型进行测试，并依据测试结果调整模型参数，以尽可能高效，尽可能精准的推荐作为目标。设计多用户实际测试，检测推荐算法效果。</p>			
<p>六、进度安排</p>			
序号	各阶段安排	起止日期	备注
1	完成论文课题确定，对课题进行基础分析，分析研究的背景和意义、国内外研究现状，进行需求分析。	2023 年 11 月 1 日- 2024 年 1 月 1 日	
2	完成课题前置基础知识的学习，包括：爬虫基础使用，熟悉 Django 与 Vue，熟悉协同过滤技术。	2024 年 1 月 2 日- 2024 年 1 月 15 日	
3	完成对现有互联网招聘岗位网站的信息爬取，对数据进行综合处理并存入 RDS MySQL 数据库	2024 年 1 月 16 日- 2024 年 2 月 1 日	
4	使用 Django 与 Vue 搭建 Web 前后端,将其与 MySQL 数据库相连接测试，完成数据显示工作	2024 年 2 月 2 日- 2024 年 3 月 1 日	
5	设计并实现协同过滤推荐模型，利用已有数据进行模型训练，并将模型部署至服务器。	2024 年 3 月 2 日- 2024 年 4 月 1 日	
6	对爬虫模块、Web 模块与推荐算法模块进行测试，测试其稳定性与效率。撰写毕业论文。	2024 年 4 月 2 日- 2024 年 5 月 10 日	



## 七、参考文献

- [1] Jacob Jacoby.Perspectives on Information Overload[J].Journal of Consumer Research,1984(4).
- [2] Covington P,Adams J,Sargin E.Deep neural networks for youtube recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems.ACM,2016:191-198.
- [3] 孙黎博.基于和矩阵分解的推荐算法分析[J].电子技术与软件工程,2022(18):218-221.
- [4] Goldberg D,Nichols D,Oki B M,et al.Using collaborative filtering to weave an information tapestry[J].Communications of the ACM,1992,35(12):61-71.
- [5] Ghuli P,Ghosh A,Shettar R.A collaborative filtering recommendation engine in a distributed environment[C]//2014 International Conference on Contemporary Computing and Informatics(IC3I).IEEE,2014:568-574.
- [6] Zhang H,Yang H,Huang T,et al.DBNCF:Personalized courses recommendation system based on DBN in MOOC environment[C]//2017 International Symposium on Educational Technology(ISET).IEEE,2017:106-108.
- [7] 蔡辉虎.基于深度神经网络的协同过滤推荐系统体系结构研究[J].信息记录材料,2023,24(11):236-238+242.DOI:10.16009/j.cnki.cn13-1295/tq.2023.11.011.
- [8] 李建超.基于神经网络的协同过滤推荐算法研究[D].长春工业大学,2023.DOI:10.27805/d.cnki.gccgy.2023.000845.
- [9] 张铁斌,郑行,邢星等.基于深度神经网络融合特征信息的协同过滤模型[J].渤海大学学报(自然科学版),2022,43(02):172-177.DOI:10.13831/j.cnki.issn.1673-0569.2022.02.005.
- [10] 黄立威,江碧涛,吕守业等.基于深度学习的推荐系统研究综述[J].计算机学报,2018,41(07):1619-1647.


## 指导教师意见

同意开题。

指导教师 (签字):   
2024 年 1 月 2 日

## 所在系 (所) 意见

同意开题。

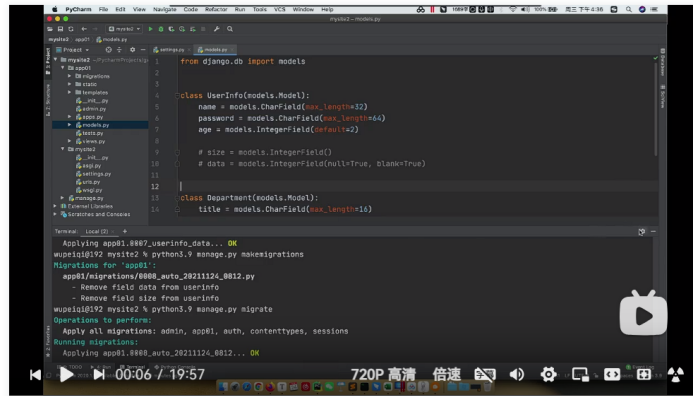
负责人 (签字):   
2024 年 1 月 3 日

## 毕业设计（论文）中期检查表

题 目	基于协同过滤的互联网岗位推荐系统的设计与实现				
学部（院）	计算机与信息工程学院	专业	计算机科学与技术 C		
班级	20 计科 C2	姓名	王耀彬	学号	20201111012
学 生 填 写	<p>工作进度</p> <p style="color: red;">（对照任务书填写）</p> <p>根据任务书进度要求，已经基本完成：</p> <p>（1）导师介绍开题，布置毕业设计（论文）任务，进行开题。</p> <p>（2）熟悉毕业设计任务和目标，对课题进行基础分析，分析研究的背景和意义、国内外研究现状，查阅相应资料。</p> <p>（3）进行需求分析，确定技术路线与目标，熟悉开发工具与平台。Python 端使用 Pycharm、前端采用 WebStorm 软件。</p> <p>（4）开始设计与部署，完成主要模块的搭建与调试，以及模块互联的初步测试，使用 Django 搭建 Web 后端，以 Vue 为框架搭建前端。尝试进行爬虫取得数据。使用阿里云数据库，与其建立连接。</p> <p>（5）学习协同过滤技术，利用协同过滤实现简单的功能。</p> <p>当前本人工作进度与任务书中规定工作进度比较属于正常，已经按任务书规定开始搭建基于深度学习协同过滤的推荐模型大致框架。</p>				
	<p>中期报告：</p> <p style="color: red;">（已完成的调研报告、外文翻译、方案分析、试验数据、设计草图、阶段性成果等）</p> <p>目前毕设课题已经取得了以下阶段性成果：</p> <p>（1）完成相关理论与技术学习</p> <p>通过查找资料，本人对毕设课题相关的爬虫技术、Python Web 技术、Vue 框架、云数据库技术与深度学习技术都有了一定的了解，通过知网查询相关论文了解了国内外研究现状。</p> <p>Web 端技术学习：</p>				

## 最新Python的web开发全家桶 (django+前端+数据库)

106.0万 2.5万 2022-04-05 05:30:23



银角大王-武沛齐-真身 发消息

python开发老炮...

充电

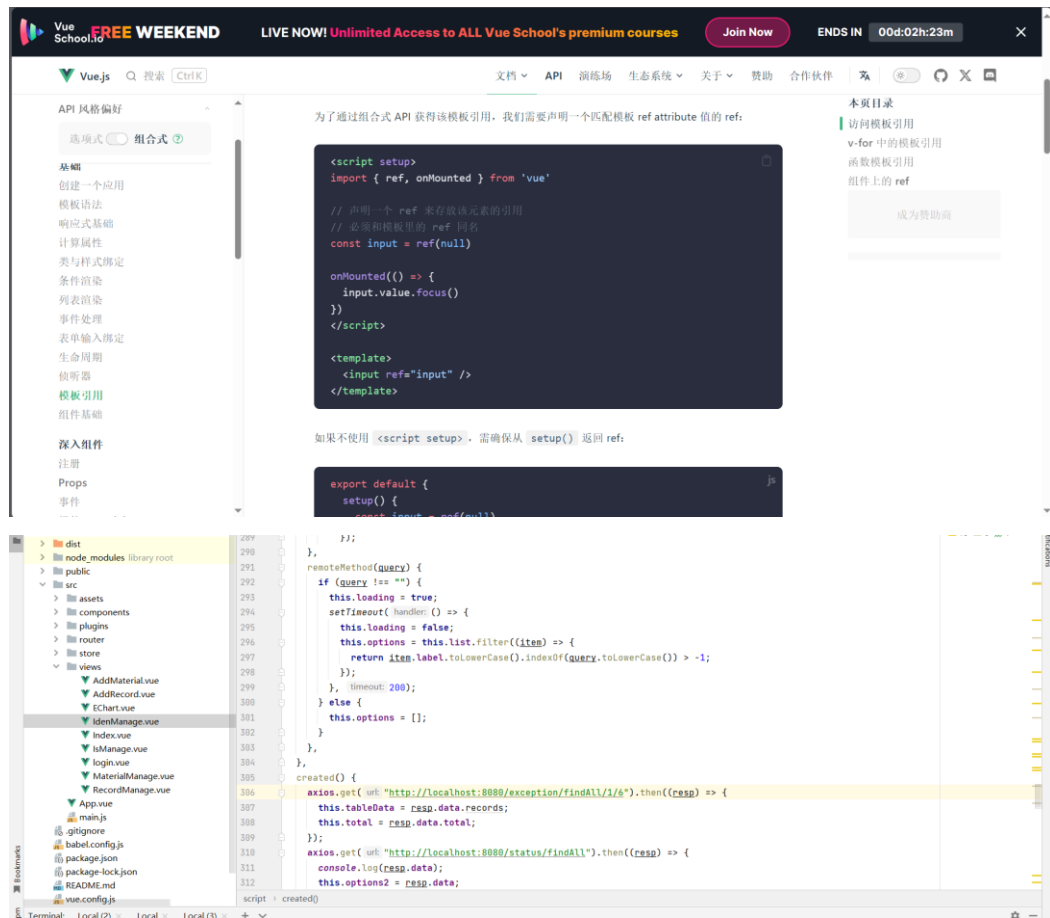
+ 关注 4.0万

弹幕列表

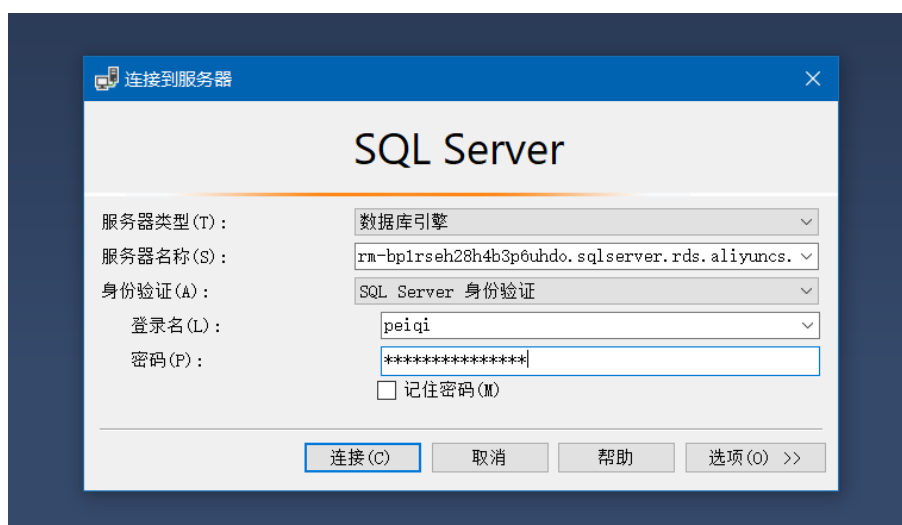
视频选集 (47/100)

自动连播

- P44 1-13 orm-连接mysql的配置 04:15
- P45 1-14 orm-类创建表 10:52
- P46 1-15 orm-创建和修改表结构 09:27
- P47 1-16 orm-增删改查 19:57
- P48 1-17 案例: 用户管理 47:41
- P49 2-1 今日概要 01:26
- P50 2-2 创建项目和app 05:03
- P51 2-3 表结构的创建 35:02
- P52 2-4 生成数据库 06:45
- P53 2-5 部门列表展示 19:39



云数据库技术:



深度学习技术:



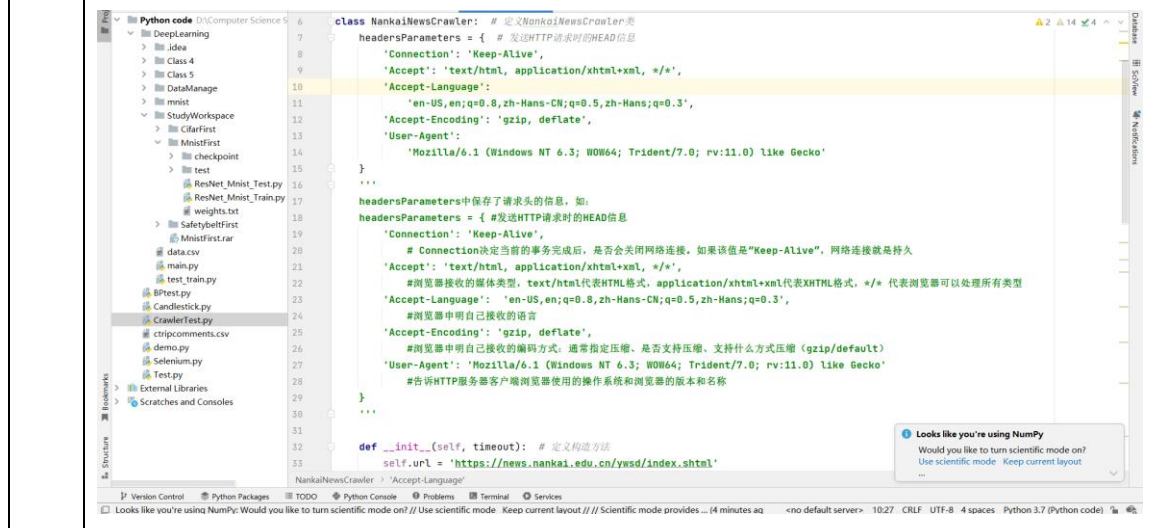


## (2) 完成核心功能模块

系统功能已经基本实现，包括 Python Web 后端与 Vue 前端，云数据库的连接等。

Python Web: Django 是 Python 社区的两大最受欢迎的 Web 框架之一，凭借功能强大的脚手架和诸多开箱即用的组件，用 Django 搭建 Web 应用快速而又省力。在使用原理上，Django 与 Java 语言中的 SpringBoot 的注解开发有类似之处。通过简单的后端接口，并做好相应配置，即可以在前端直接调用后端逻辑。

爬虫:爬虫，即网络数据采集，通过编程向网络服务器请求数据（HTML 表单），然后解析 HTML，提取出自己想要的信息。通过试验，利用 requests 库爬取数据。



```
Python code D:\Computer Science> 31
> DeepLearning
> .idea
> Class 4
> Class 5
> DataManager
> mnist
> StudyWorkspace
> CifarFirst
> MnistFirst
> checkpoint
> ResNet_Mnist_Test.py
> ResNet_Mnist_Train.py
> weights.txt
> SafetybeltFirst
> MnistFirst.rar
> data.csv
> main.py
> test_train.py
> BPtest.py
> Candlestick.py
> CrawlerTest.py
> ctripcomments.csv
> demo.py
> Selenium.py
> Test.py
> External Libraries
> Scratches and Consoles

def __init__(self, timeout): # 定义构造方法
    self.url = 'https://news.nankai.edu.cn/ywzd/index.shtml'
    self.timeout = timeout # 连接超时时间设置(单位:秒)
    self.title = []

def GetHtml(self): # 定义GetHtml方法
    request = requests.get(self.url, timeout=self.timeout, headers=self.headersParameters) # 根据指定网址爬取网页
    request.encoding = 'utf-8'
    self.html = request.text # 获取新闻网页内容

def GetTitles(self): # 定义GetTitles方法
    titles = re.findall(r'<div align="left">{<div>}</div>', self.html) # 匹配新闻标题
    for i in range(len(titles)): # 对于每一个标题
        temp = re.sub(r'<div>*>', '', titles[i]) # 去除所有HTML标记,即<...>
        self.title.append(temp.strip()) # 将标题两边的空白符去掉

def PrintTitles(self): # 定义PrintTitle方法
    no = 1
    for title in self.title: # 显示标题
        print(str(no) + ':' + title)
        no += 1

def ExportToFile(self, filepath): # 定义ExportToFile方法,将标题输出到文件中
    with open(filepath, 'w', encoding='utf-8') as f:
        no = 1
        for title in self.title:
            f.write(str(no) + ':' + title + '\n')
```

深度学习模块:基于测试目的, 使用 AlexNet、LetNet、ResNet 设计了手写数字识别神经网络, 了解了神经网络的运行机制, 熟悉了常用的神经网络模型。

```
Python code D:\Computer Science> 93
> DeepLearning
> .idea
> Class 4
> Class 5
> test
> cf1ar10_alexnet8.py
> cf1ar10_baseline.py
> cf1ar10_datasets.py
> cf1ar10_inception10.py
> cf1ar10_lenet5.py
> cf1ar10_resnet8.py
> cf1ar10_vgg16.py
> mnistPredict.py
> weights.txt
> DataManager
> mnist
> StudyWorkspace
> data.csv
> main.py
> test_train.py
> BPtest.py
> Candlestick.py
> CrawlerTest.py
> ctripcomments.csv
> demo.py
> Selenium.py
> Test.py
> External Libraries
> Scratches and Consoles

for v in model.trainable_variables:
    file.write(str(v.name) + '\n')
    file.write(str(v.shape) + '\n')
    file.write(str(v.numpy()) + '\n')
file.close()

##### show #####

# 显示训练集和验证集的acc和Loss曲线
acc = history.history['sparse_categorical_accuracy']
val_acc = history.history['val_sparse_categorical_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

plt.subplot(1, 2, 1)
plt.plot(acc, label='Training Accuracy')
plt.plot(val_acc, label='Validation Accuracy')
plt.title('Training and Validation Accuracy')
plt.legend()

plt.subplot(1, 2, 2)
plt.plot(loss, label='Training Loss')
plt.plot(val_loss, label='Validation Loss')
plt.title('Training and Validation Loss')
plt.legend()
plt.show()
```

```
Python code D:\Computer Science> 14
> DeepLearning
> .idea
> Class 4
> Class 5
> test
> cf1ar10_alexnet8.py
> cf1ar10_baseline.py
> cf1ar10_datasets.py
> cf1ar10_inception10.py
> cf1ar10_lenet5.py
> cf1ar10_resnet8.py
> cf1ar10_vgg16.py
> mnistPredict.py
> weights.txt
> DataManager
> mnist
> StudyWorkspace
> data.csv
> main.py
> test_train.py
> BPtest.py
> Candlestick.py
> CrawlerTest.py
> ctripcomments.csv
> demo.py
> Selenium.py
> Test.py
> External Libraries
> Scratches and Consoles

x_train, x_test = x_train / 255.0, x_test / 255.0

class ResnetBlock(Model):
    def __init__(self, filters, strides=1, residual_path=False):
        super(ResnetBlock, self).__init__()
        self.filters = filters
        self.strides = strides
        self.residual_path = residual_path


        self.c1 = Conv2D(filters, (3, 3), strides=strides, padding='same', use_bias=False)
        self.b1 = BatchNormalization()
        self.a1 = Activation('relu')

        self.c2 = Conv2D(filters, (3, 3), strides=1, padding='same', use_bias=False)
        self.b2 = BatchNormalization()

        # residual_path为True时,对输入进行下采样,即用1x1的卷积核做卷积操作,保证x能和F(x)维度相同,顺利相加
        if residual_path:
            self.down_c1 = Conv2D(filters, (1, 1), strides=strides, padding='same', use_bias=False)
            self.down_b1 = BatchNormalization()

            self.a2 = Activation('relu')

    def call(self, inputs):
        residual = inputs # residual等于输入值本身,即residual=x
        # 将输入通过卷积、BN层、激活层,计算F(x)
```

指导教师填写	检查项目	评定情况 (√)			备注
		优良	合格	不合格	
	计划执行情况				
	系统设计与实现				
	创新意识				
	自主学习能力				
	工作态度及出勤情况				
	存在问题及建议:				
指导教师 (签字): 					
2024 年 3 月 20 日					

(注：具体检查内容由各专业根据实际情况制定后统一实施并报学部（院）备案)

毕业设计（论文）指导记录表

题目	基于协同过滤的互联网岗位推荐系统的设计与实现				
学部（院）	计算机与信息工程学院	专业	计算机科学与技术		
班级	20 计科 C2	姓名	王耀彬	学号	20201111012
指 导 内 容					
讨论毕设选题，根据自身实际情况和指导老师意见确定选题。					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2023.11.30	
指 导 内 容					
了解学生选题的方向和意图，对毕设进行的时间流程提出要求。					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2023.12.21	
指 导 内 容					
进行开题答辩，开题答辩老师现场进行指导并修改开题报告					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2023.12.27	
指 导 内 容					
指导学生对课题进行具体研究，敲定开发平台与工具，并提供相关文献资料。					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2024.01.01	
指 导 内 容					
了解学生目前查询相关资料文献的情况，对疑问进行解答，帮助梳理课题内容。					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2024.01.06	
指 导 内 容					
布置项目开发任务，引导学生进行外文文献翻译以及课题的开发。					
学生（签字）：		王耀彬		指导老师（签字）： 日期：2024.01.11	
指 导 内 容					
要求学生开始需求分析工作，确定任务整体框架，确定技术路线，进行资料查找。分析所需技术，学习前置知识，确定开发流程，打好框架。					



学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.02.26
指 导 内 容					
要求学生开始对环境进行配置, 并开发初始版本。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.03.01
指 导 内 容					
开始对项目的各个模块进行开发、调试, 并指导学生不断对功能进行优化。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.03.05
指 导 内 容					
指导学生进行程序联调, 对各个页面进行功能测试, 接受中期检查。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.03.15
指 导 内 容					
进一步细化工程项目, 验收结题, 开始着手论文撰写工作。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.03.25
指 导 内 容					
对学生初步提交的论文进行审阅, 并指导进行查重修改, 指导外文翻译等文件编写。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.04.01
指 导 内 容					
确定学生毕业论文最终稿件, 通知其上交相关材料和毕设答辩相关事项, 查看答辩相关准备。					
学生(签字):	王耀彬	指导老师(签字):	Wassil	日期:	2024.04.16