



北京大学

# 挑战杯项目报告

题目： 基于深度学习的大学生阅读推荐交流系

统

A Recommendation and Communication System for College

Students' Reading, based on deep learning

姓 名： 赵皓晨、孟梓墨、张润博、张皓天

学 号： 2100017417、2100017702、

2100017798、2100017733

院 系： 元培学院

专 业： 人工智能、计算机科学与技术、数

据科学与大数据技术

导师姓名： 邹磊 教授

二〇二三年三月



## 版权声明

任何收存和保管本论文各种版本的单位和个人，未经本论文作者同意，不得将本论文转借他人，亦不得随意复制、抄录、拍照或以任何方式传播。否则一旦引起有碍作者著作权之问题，将可能承担法律责任。

## 摘要

本项目旨在研发一款面向大学生的阅读推荐交流系统。我们使用了 Django 作为后端框架，SQLite3 作为数据库，以及 Foundation 作为前端框架完成了系统搭建，实现了阅读推荐、交流和发布写作的功能。我们在推荐算法上进行了广泛调查和实验，最后选择采用了 NN 和 VAE 复合的深度学习模型，达到了相对良好的实际效果。本项目易于部署，可以作为推进大学校园数字化建设的工程投入使用，服务于广大大学生的学习生活。

关键词：软件开发，数据科学，深度学习



## 目 录

<b>第一章 项目概况</b> .....	1
1.1 项目简介 .....	1
1.2 项目架构 .....	1
1.3 功能介绍 .....	3
1.4 社会价值 .....	3
<b>第二章 技术说明</b> .....	5
2.1 后端构建: Django .....	5
2.2 数据库: SQLite3 .....	6
2.3 前端架构: Foundation .....	6
2.4 机器学习模型 .....	7
2.4.1 数据集 .....	7
2.4.2 推荐算法 .....	8
2.4.3 模型对比 .....	10
<b>第三章 效果展示</b> .....	13
3.1 账户功能 .....	13
3.1.1 注册功能 .....	13
3.1.2 登录功能 .....	13
3.1.3 用户主页 .....	14
3.1.4 修改信息功能 .....	14
3.2 推荐功能 .....	15
3.2.1 推荐页面 .....	15
3.2.2 搜索页面 .....	15
3.2.3 图书详情页面 .....	16
3.3 讨论功能 .....	17
3.3.1 讨论组注册功能 .....	17
3.3.2 讨论中心 .....	17
3.3.3 讨论组页面 .....	18
3.4 创作页面 .....	18
3.4.1 创作者页面 .....	18

3.4.2 更新页面 .....	19
3.4.3 作品主页 .....	19
3.4.4 创作中心 .....	20
<b>第四章 开发进程 .....</b>	<b>23</b>
4.1 服务端开发进程 .....	23
4.1.1 第一阶段：2023 年 1 月 7 日——2023 年 1 月 18 日 .....	23
4.1.2 第二阶段：2023 年 1 月 29 日——2023 年 2 月 4 日 .....	24
4.1.3 第三阶段：2023 年 2 月 5 日——2023 年 2 月 18 日 .....	25
4.1.4 第四阶段：2023 年 2 月 19 日——2023 年 3 月 5 日 .....	26
4.2 推荐算法进程 .....	26
4.2.1 第一阶段：2023 年 1 月 7 日——2023 年 1 月 18 日 .....	26
4.2.2 第二阶段：2023 年 1 月 29 日——2023 年 2 月 4 日 .....	27
4.2.3 第三阶段：2023 年 2 月 5 日——2023 年 2 月 18 日 .....	27
4.2.4 第四阶段：2023 年 2 月 19 日——2023 年 3 月 5 日 .....	27
<b>第五章 总结与展望 .....</b>	<b>29</b>
<b>参考文献 .....</b>	<b>31</b>
<b>致谢 .....</b>	<b>33</b>

## 主要符号对照表

$x, y, m, n, t$	标量, 通常为变量
$K, L, D, M, N, T$	标量, 通常为超参数
$x \in \mathbb{R}^D$	D 维列向量
$(x_1, \dots, x_D)$	D 维行向量
$(x_1, \dots, x_D)^T$ or $(x_1; \dots; x_D)^T$	D 维行向量
$A \in \mathbb{R}^{K \times D}$	大小为 $K \times D$ 的矩阵
$x \in \mathbb{R}^{KD}$	$(KD)$ 维的向量
$\mathbb{M}_i$ or $\mathbb{M}_i(x)$	第 $i$ 列为 $\mathbf{1}$ (或者 $x$ ), 其余为 $\mathbf{0}$ 的矩阵
$diag(\mathbf{x})$	对角矩阵, 其对角元素为 $\mathbf{x}$
$I_N$ or $I$	$(N \times N)$ 的单位阵
$diag(A)$	列向量, 其元素为 $A$ 的对角元素
$A \in \mathbb{R}^{D_1 \times D_2 \times \dots \times D_K}$	大小为 $D_1 \times D_2 \times \dots \times D_K$ 的张量
$\{x^{(n)}\}_{n=1}^N$	集合
$\{(x^{(n)}, y^{(n)})\}_{n=1}^N$	数据集
$\mathcal{N}(x; \mu, \Sigma)$	变量 $x$ 服从均值为 $\mu$ , 方差为 $\Sigma$ 的高斯分布

① 本符号对照表内容选自邱锡鹏老师的《神经网络与深度学习》<sup>[1]</sup>一书。



# 第一章 项目概况

本章对该项目的概况进行简要的说明，包括项目简介、项目架构、功能介绍和社会价值。

## 1.1 项目简介

该项目致力于打造一款基于深度学习的大学生阅读推荐交流 Web 应用程序，包含阅读推荐、讨论区和创作区。

用户注册并填写基本信息后，本系统可以通过深度学习算法，基于用户填写的专业方向、兴趣爱好、已读书目等基本信息，生成用户画像。根据该用户的用户画像，系统可以向用户推荐适合用户的书籍名称及简介，并在系统中匹配更可能与之志趣相投的用户和小组，以便有共同话题读者群体进行交流。

在讨论区，读者之间可以通过类似论坛的形式发帖互动，讨论所读书目的内容，分享自己阅读后产生的宝贵想法，在思维火花的碰撞中一起学习，共同进步。另一方面，创作者也可以在讨论区和读者进行直接的交流，方便创作者和读者互相了解对方的想法，从而激发创作者的创作热情，提高读者的阅读体验。

在创作区，创作者可以发布自己的文学作品，定期或不定期更新。读者可以在这里关注特定的创作者和作品，然后及时收到关注作品的最新动态。该系统可以释放大学生丰富的思维和写作欲望，有助于大学生在阅读和写作中健康成长。

(项目代码：<https://github.com/PKU-ChallengeCup2022-2023/RC4CSR/tree/master>)

(报告模版：PKU Undergraduate Thesis Template modified from pkuthss<sup>[2]</sup>)

## 1.2 项目架构

该项目的结构图 1.1 所示：运行代码位于系统目录（System/），运行环境配置位于 freeze.yml 文件。

在系统目录内主要功能被划分为四个功能模块（APP）并分别放入四个子目录，分别是账户模块（System/Account）、讨论模块（System/Discussion）、推荐模块（System/Recommendation）和写作模块（System/Writing）。在系统目录下的其他部分分别是：数据文件夹（System/data）和数据导入程序（default.py, loaddata.py），用于导入获取的书籍数据文件和书籍类型；系统子目录（System/System）和管理文件（manage.py），由 Django 模型自动生成，用于运行架构；数据库文件（db.sqlite3），是

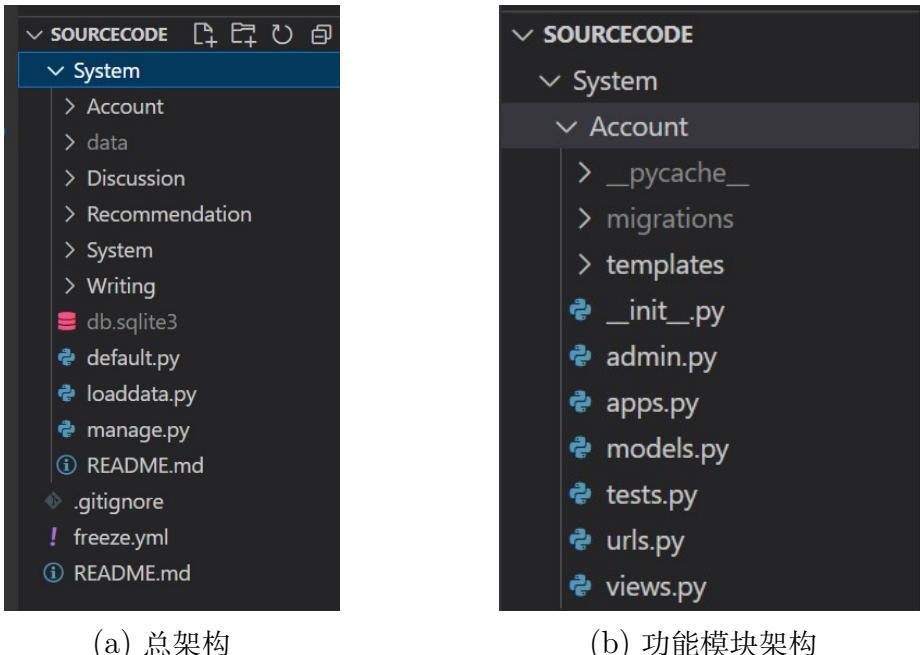


图 1.1 项目架构图示

用于储存全部的系统数据的数据库引擎；以及项目说明 (README.md)，为项目简介和操作使用指南。

下面将对四个功能模块的架构进行简要的介绍。

由于四个功能模块都是由 Django 的 startapp 命令生成，具有 Django 系统下 app 的典型结构，如图 1.1(b) 所示。因此我们选择以账户模块为例进行结构的介绍。

在功能模块下的子目录 `_pycache_` 和 `migrations` 都是配置和运行时自动生成的部分，不需要手动设计实现。子目录 `templates` 内保存了网页前端的 `html` 文件，是该功能模块下所有网页的前端模板。

文件 `_init_.py` 和 `apps.py` 分别在跨模块导入和配置模型上进行声明；文件 `admin.py` 则为超级用户登录站点管理的后台以后进行查看修改等高权限行为的模型进行了准备；文件 `test.py` 内可以部署自动测试（本项目目前没有使用自动测试）。文件 `models.py` 内，我们编写了各种具体功能模型相关的类（class），实现了模型定义和各种功能函数。文件 `urls.py` 规定了用户访问每个功能模块要使用的网址，并将网址和对应的后端视图函数完成匹配。文件 `views.py` 内，我们为每个网页设计了后端视图函数，实现了用户前端操作对后端系统数据的更新修改，从而完成了我们给各个功能模块设计的功能逻辑。

### 1.3 功能介绍

在本小结，我们将相对具体地对我们项目设计的各个功能进行介绍。

在账户模块，我们对用户模型进行了定义。通过编写用户类 (class PlatformUser)，收集了用户的性别、专业、书籍偏好等信息，并实现了用户在本平台上自己设置昵称、密码等安全信息的功能。账户模块实现了用户注册、登录和修改个人信息的功能。

在账户模块的设计实现中，用户在使用该平台提供的服务前必须注册，填写昵称、密码以及其他个人信息。出于安全性的考虑，我们的程序会自动对用户的密码复杂程度进行检查以防止用户自己设置过短过单一的密码后容易被盗；我们在储存密码时也没有直接明文存储用户密码，而是存储用户密码多次哈希后的值，保证了我们平台的用户安全。另外，我们在每个页面的访问权限上也进行了设置，使得用户私密信息得到了进一步的保护。

在讨论模块，我们设计的模型主要是讨论组和讨论记录。用户进入讨论区以后，可以选择自己感兴趣的数据作为主题，创建一个自己的讨论组。在讨论组里，用户们可以自由地发表自己的见解，可以回复跟帖别人的讨论，也可以对自己喜欢的讨论点赞。

在推荐模块，我们实现了书籍推荐、书籍搜索和书籍评分的功能。我们编写了书籍模型、书籍类别模型和搜索记录模型。用户在进入书籍搜索页面以后，通过在搜索栏键入正确的书籍名称，点击搜索后，系统将给出对应书籍的详情页超链接，在用户点击超链接进入详情页后可以查看书籍的详细信息。

我们使用了机器学习的方法来构建书籍推荐的功能。根据我们的设计，在用户使用平台较少时，主要以用户所填写的个人信息作为依据进行推荐；而当用户在平台活动较多后，我们主要使用用户的行为数据进行推荐。在测试了多个模型后，我们发现这种方法可以在避免冷启动问题的同时，更精确地完成用户画像的匹配，有着更好的性能表现。

在写作模块，我们编写了作品类 (class Pencraft) 和章节类 (class Chapter) 作为主要的模型。功能上，我们设计了创作者首页、发布作品和更新作品三个功能。每个用户都有自己的创作者首页，在这里，用户可以查看自己已经发布的作品。在发布作品和更新作品的页面，用户分别可以开始一个新作品的第一个章节，和选择一部已发布作品后更新一个章节。

以上是本项目功能的介绍，实际效果将于第三章以截图形式进行展示。

### 1.4 社会价值

我们组选择大学生阅读推荐交流作为项目的立意，充分考虑到了本项目能带来的社会价值。

我们认为，该项目可以为大学生提供阅读和交流的平台，促进知识交流和学术讨论，提高学术水平和学习效率，增强社交能力和交际能力，以及提高阅读习惯和兴趣，从而有助于提高大学生的综合素质和未来发展。

首先，我们项目成果所提供的平台可以促进大学生的知识交流和学术讨论，对大学生的学术发展非常有益。首先，这种交流可以帮助学生更好地理解和掌握学术知识。通过与其他人分享和讨论，学生可以获得更广泛和深入的见解，了解不同的观点和思路。例如，一个学生可能会发现其他人对某个书籍的某个观点有不同的看法，这可能会促使他重新思考自己的看法，并更深入地了解文献的主题和内容。其次，这种交流还可以激发新的研究方向和思考方式。通过与其他人的交流和互动，学生可能会获得一些新的启示，从而产生新的研究想法。例如，一个学生在阅读一篇文学作品或学术文献时，可能会被其他人的评论或观点所启发，从而想到一些新的研究问题或思路。

其次，它可以帮助大学生更高效地阅读和理解学术文献，从而提高学术水平和学习效率。通过浏览其他人的评论和回复，大学生可以快速了解文献的重要观点和争议点。这可以帮助我们大学生更快地理解和掌握文献的内容，提高学术水平。另一方面，这种工具可以帮助学生更好地组织和管理自己的学习进程。比如说，大学生在我们的系统上可以非常方便地记录自己的阅读笔记和评论，并查看其他人的笔记和评论。这可以帮助我们更好地组织和管理自己的学习进程，避免遗漏重要信息，提高学习效率。

第三，它可以帮助大学生扩展社交网络并增强交际能力。和其它形式的社交软件相比，我们平台具有更加浓厚的学术气息，也有更加集中的讨论主题与交际目的，因此，它在大学生群体当中，可以提供相对优质的社交资源。具体来说，我们的平台可以让大学生更便利地结识志同道合的朋友，尤其是有共同兴趣和研究方向的新朋友，并帮助他们建立联系。因此，在我们的平台上，大学生可以轻松拓展自己的社交网络。此外，通过在我们平台上进行阅读书籍与文献、以评论形式和与他人互动，以及发布自己的文学作品或者学术见解，可以锻炼大学生的阅读表达能力。通过和不同学术背景与文化背景的用户沟通，大学生会学习和锻炼到沟通的使用技巧，也会为之后学术和生活领域的合作打下基础。

最后，从使用角度出发，部署我们的系统也是推进校园智慧化、数字化建设的一个有效方法。我们的项目成果不管是独立部署还是集成在其它智慧校园系统在操作上都十分容易便捷。举例来说，北京大学元培学院的智慧校园系统（YPPF）的“元培书房”，就可以接入我们的服务，在“何善衡图书馆”海量实体藏书的基础上，拓展出更加便捷的信息化运用，为元培同学的阅读交流提供便利。

综上所述，我们相信我们的项目具有十分正面的社会意义，是真正有价值的发明创造。

## 第二章 技术说明

本章对我们挑战杯项目所采用的各种技术进行简要的说明，包括后端建构的 Django 框架、后端数据库的 SQLite3 数据库、前端网页展示的 Foundation 架构，以及我们推荐系统的核心：机器学习模型。

我们的框架和模型都是在开源项目的基础上进行编写的，使用开源项目的优点很多，比如有：可靠性和稳定性：由于开源框架是由一个社区开发和维护的，所以可以期待较高的质量、稳定性和可靠性；社区支持和协作：开源框架具有广泛的社区支持，可以获得用户反馈和修复错误，还可以参与社区贡献代码，共同协作开发；安全性：由于开源框架是开放源代码的，可以被众多开发者审查和发现漏洞，从而提高了安全性。从另一个角度来说，我们的项目代码也是开源的，任何个人和组织在遵守开源协议的前提下都可以自由使用，从而更容易实现我们项目的社会价值。

### 2.1 后端构建：Django

Django<sup>[3]</sup> 是一个由 Python 编写的一个开放源代码的 Web 应用框架。

如果使用 Django 框架，不需要极其笨重繁杂的代码，Python 的程序开发人员就可以相对轻松地完成一个正式网站所需要的大部分内容，并进一步开发出全功能的 Web 服务。Django 本身基于 MVC 模型，即 Model（模型）+ View（视图）+ Controller（控制器）设计模式，MVC 模式使后续对程序的修改和扩展简化，并且使程序某一部分的重复利用成为可能。MVC 模式的典型优势包括：低耦合、开发快捷、部署方便、可重用性高、维护成本低等。<sup>①</sup>

因此，在许多开发者的眼中，Python 加 Django 是快速开发、设计、部署网站的最佳组合。

我们选择 django 框架的原因除了代码的轻便性以外，还有以下四个原因：安全性、可拓展性、文档和社区支持，以及内置的数据库管理。安全性：Django 框架提供了许多内置的安全机制，例如跨站点请求伪造（CSRF）保护、XSS 防护和 SQL 注入防护等，这些机制有助于确保应用程序的安全性。可扩展性：Django 框架具有模块化结构，开发者可以轻松地扩展和重用现有代码。Django 还支持许多第三方插件和库，这些插件和库可以帮助开发者更快地实现特定的功能。良好的文档和社区支持：Django 框架拥有完善的官方文档和庞大的社区支持，这意味着开发者可以轻松地找到相关的资源和解决方案。数据库管理：Django 框架内置了 ORM（对象关系映射）系统，使得数

---

<sup>①</sup> 参考来源：Django 官方网页，<https://www.djangoproject.com/>

数据库操作变得更加方便和易于管理。因此，我们作为开发者在使用 django 框架进行服务端搭建时，可以更专注于业务逻辑和功能实现，不需要花费大量时间“造轮子”，不必亲自实现各种底层支持。

## 2.2 数据库：SQLite3

SQLite3<sup>[4]</sup>是一种轻量级的嵌入式关系型数据库，它是使用 C 语言编写的，可以在多个平台上运行，并且没有任何外部依赖。SQLite3 的数据库是以单个文件的形式存储在磁盘上的，非常适合小型 Web 应用的数据存储需求。<sup>①</sup>

在 Web 应用中使用 SQLite3 作为数据库具有以下好处。轻量级：SQLite3 是一种轻量级的数据库，非常适合小型 Web 应用，尤其是在资源受限的情况下，例如嵌入式设备或移动应用程序。快速：SQLite3 具有快速的读写性能，并且支持多个并发连接，可以提高 Web 应用的响应速度和用户体验。易于使用：SQLite3 是一种易于使用的数据库，它支持标准的 SQL 语言，开发人员可以使用 SQL 命令轻松地执行各种数据库操作。可移植性：SQLite3 可以在多个平台上运行，并且没有任何外部依赖，因此可以轻松地将 Web 应用程序迁移到其他平台上。安全：SQLite3 是一种安全的数据库，它提供了各种安全机制，例如加密、访问控制和安全套接字层（SSL）等。

因此，SQLite3 是一种非常适合小型 Web 应用程序的数据库，它具有轻量级、快速、易于使用、可移植性和安全性等优点，在开发小型 Web 应用程序时使用 SQLite3 是一个非常不错的选择。

我们选择 SQLite3 作为该项目的数据库主要就是看中了它的轻量性和易用性。从另一方面来说，如果我们的服务规模大大增长，使得 SQLite3 无法满足大型 Web 应用的开发需求，在我们的设计当中，我们也完全有能力迅速将数据库迁移，采用其它的关系型数据库，比如 MySQL；或者 NOSQL 数据库，比如 Redis。通常情况下，大型 Web 应用程序往往需要存储大量数据、高并发读写和查询操作，因此需要选择能够满足这些需求的可扩展和高性能数据库，但由于我们项目规模目前比较有限，轻量的 SQLite3 足够实现我们需要的功能。

## 2.3 前端架构：Foundation

Foundation<sup>[5]</sup>是一款流行的响应式前端框架，它是一个基于 CSS 和 JavaScript 的框架，用于快速创建现代化的 Web 应用程序和移动应用程序界面。

使用 Foundation 前端框架有以下优点：响应式设计：Foundation 提供了全面的响应式设计支持，可以轻松地创建适应不同屏幕大小和设备的 Web 应用程序。设计美观：

---

<sup>①</sup> 参考来源：SQLite 官方网页，<https://sqlite.org/index.html>

Foundation 具有现代化的设计和可定制的主题，可以帮助开发人员创建美观的 Web 应用程序和用户界面。易于使用：Foundation 提供了丰富的 HTML、CSS 和 JavaScript 组件和工具，可以让开发人员轻松地创建复杂的 Web 应用程序，举例来说，我们通过非常简单的 HTML 代码就实现了边框、按钮等设计要素。优化性能：Foundation 通过使用最佳实践和性能优化技术来提高 Web 应用程序的性能和速度。支持跨平台：Foundation 支持跨平台开发，可以在多个设备和平台上运行，例如桌面浏览器、移动浏览器、iOS 和 Android 等。<sup>②</sup>

因此，Foundation 前端框架是一个功能强大、易于使用和可定制的框架，适用于创建现代化和响应式的 Web 应用程序和移动应用程序。由于本组组员均为大二本科生，前端工程经验尚浅，所以采用成熟的可定制框架最能满足我们的需求，最后也达到了比较理想的美观效果。

## 2.4 机器学习模型

在本部分中，我们将具体介绍该项目的推荐算法实现与具体训练过程。

### 2.4.1 数据集

我们选择豆瓣上的部分图书作为系统的初始图书库，也同时应用于推荐系统的训练。但由于缺乏所选图书的用户阅读情况，没有合适的数据集用于训练，因此我们结合评分概率分布与偏好假设，采用生成数据集的方式。

我们在豆瓣网站呈现的诸多图书种类中，选取六大类共计 105 个子类别，通过 request 和 BeautifulSoup 两个库，从每个标签下无重复的爬取 50 本书的数据，共计 5250 本。对于每本书，我们爬取的内容包括：封面、书名、类别、豆瓣链接、作者、出版社、出版日期、平均评分，评分分布，10 条书评。部分爬取内容如下图所示。

```
6,文学,小说,https://book.douban.com/subject/4913064/,活着,余华,作家出版社,2012-8-1,9.4,745503,77.2%,20.5%,2.1%,0.1%,0.1%
7,文学,小说,https://book.douban.com/subject/36069426/,大医·破晓篇,马伯庸,上海文艺出版社,2022-9,8.5,3299,48.3%,40.0%,10.2%,1.1%,0.5%
8,文学,小说,https://book.douban.com/subject/35653884/,潮汐图,林棹,上海文艺出版社,2022-1,8.6,2377,45.2%,38.6%,12.7%,2.4%,1.1%
9,文学,小说,https://book.douban.com/subject/3633461/,一句顶一万句,刘震云,长江文艺出版社,2009-3,8.8,65385,51.7%,39.7%,7.8%,0.6%,0.2%
10,文学,小说,https://book.douban.com/subject/36152943/,通俗小说,仁科,四川文艺出版社,2022-12-15,6.8,1893,15.3%,34.6%,35.4%,10.7%,4.0%
11,文学,小说,https://book.douban.com/subject/1322455/,遥远的救世主,豆豆,作家出版社,2005-05-01,8.7,36670,59.2%,29.0%,8.6%,2.0%,1.2%
12,文学,小说,https://book.douban.com/subject/34998019/,秋园,杨本芬,北京联合出版公司,2020-6,9.0,63019,60.3%,34.4%,5.0%,0.3%,0.1%
```

图 2.1 爬取内容

---

<sup>②</sup> 参考来源：Foundation 开发文档，<https://get.foundation/frameworks-docs.html>

进一步地，我们根据每本书的平均评分与评分分布，构建训练数据集。我们给定用户的相关性偏好，从对应类别的书籍中随机选取若干标记为喜欢，同时赋予一个较高分数。此外，从总体图书数据中随机选取若干，根据平均评分与评分分布进行喜好标记与评分赋值。根据此方法，我们模拟出 10000 名用户、累计约 2000000 条的阅读数据，并将其用于之后的训练。

#### 2.4.2 推荐算法

为了充分避免推荐系统“冷启动”问题<sup>[6]</sup>，我们采取了深度神经网络 DNN 与 VAE 生成模型的结合。在用户刚注册不久，由于我们对用户的具体行为不了解，因此 VAE 推荐模型效果欠佳。此时我们主要根据用户在注册时选择的偏好，基于传统神经网络进行推荐。当用户在系统上的互动达到一定程度，进而可以根据用户的评分、书评、搜索等相关信息，构造出用户的行为向量，通过 VAE 算法进行更为精确具体的推荐。下面将对两个模型分别进行介绍。

##### 2.4.2.1 模型：DNN

我们依据此前构造出的用户数据集训练深度神经网络 DNN。我们采用“基于预测评分”的推荐方法，利用 DNN 来预测用户对某本书籍的评分，基于此进行书籍推荐。神经网络架构如下图所示：

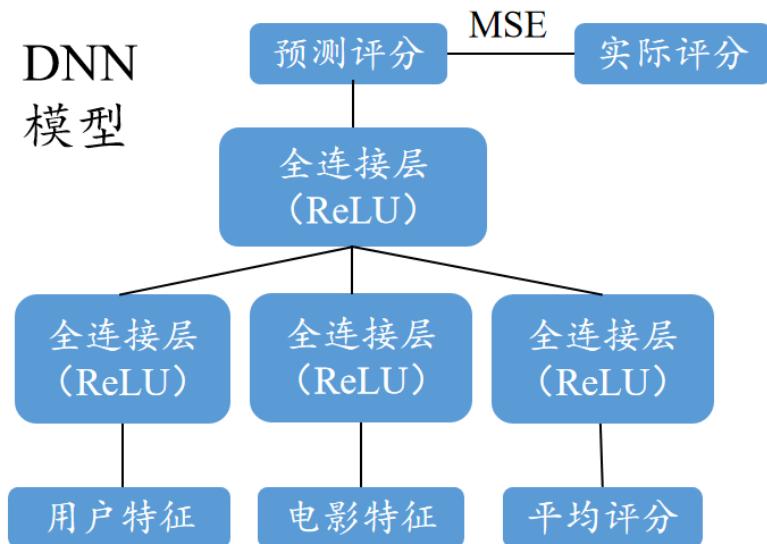


图 2.2 DNN 模型

我们将数据集整理成用户特征，书籍信息，用户评分这样的形式。将用户特征与

书籍信息数字化，作为神经网络的输入。经过上图所示的网络架构，输出预测评分。我们采用均方误差 MSE 作为损失函数，通过梯度下降不断缩小预测评分与真实评分的差别，优化参数，得到最终模型。

对于新注册用户，我们将其注册时填写的特征依次与图书库不同图书的信息整合，经由 DNN 网络得到不同图书的预测评分，选取评分最高的前 k 个推荐给用户。

#### 2.4.2.2 模型：VAE

我们依据此前构造出的用户数据集训练 VAE 推荐模型。VAE 模型原理如下图所示<sup>[7]</sup>：

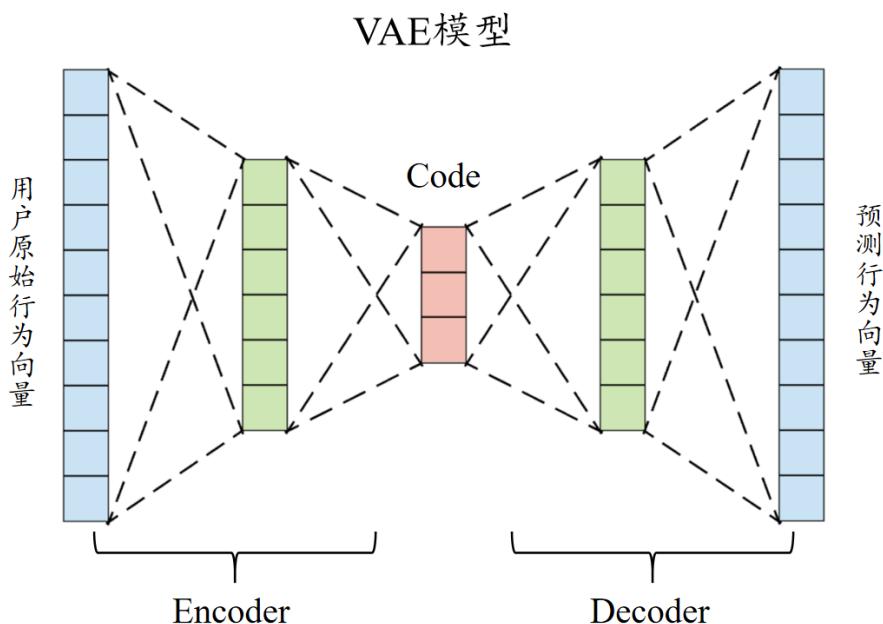


图 2.3 VAE 模型

具体来说，VAE 模型以用户和图书的互动情况作为输入。对于初始数据集，我们将每个用户与图书的互动情况标记为向量形式，喜好图书 i 则在向量对应位置标记为 1，其余为 0，从而得到用户与图书的关联矩阵，关联矩阵经过编码器 (Encoder) 被编码成低维隐空间上的一个概率分布。在推荐时，我们直接从该隐式概率分布上抽样，并且经过一个解码器 (Decoder) 解码成与输入关联矩阵大小相同的一个矩阵。根据每个元素的分数来为用户推荐模型认为最为适合的物品。

在我们的平台部署中，VAE 模型旨在当平台用户互动数达到一定程度之后，进一步利用用户交互数据来进行推荐。我们根据用户的评分、书评、搜索等相关信息构建出用户的特征向量，经过 VAE 模型预测，选取前 k 个未看过的书籍推荐给用户。

### 2.4.3 模型对比

评估一个推荐系统算法的好坏远比简单的分类或者回归问题而复杂。关于推荐效果的评估指标有很多，对于对显式的评分进行预测的模型可以使用均方误差 (MSE) 等常规指标来评判，而对于其他的利用隐式特征进行预测的推荐模型往往采用像召回率 (Recall)，覆盖率 (Coverage) 等指标。不同的指标具有不同的人物场景和不同的特征含义。

我们采取根式均方误差 (RMSE)、TopN 召回率等指标来对我们的模型与传统的推荐模型进行效果对比。我们利用根式均方误差 (RMSE) 和平均绝对值误差 (MAE) 来检测模型对于用户评分预测的准确性，具体定义为

$$RMSE = \sqrt{\frac{1}{MN} \sum_{i=1}^N \sum_{j=1}^M (r_{ij} - \hat{r}_{ij})^2}$$

$$MAE = \frac{1}{MN} \sum_{i=1}^N \sum_{j=1}^M |r_{ij} - \hat{r}_{ij}|$$

其中  $r_{ij}$  表示用户  $i$  对物品  $j$  的真实评分， $\hat{r}_{ij}$  表示模型预测的用户  $i$  对物品  $j$  的评分。TopN 召回率，即最推荐的  $N$  个项目中的召回率，记作  $Recall@N$ ，它的具体定义为

$$Recall@N = \frac{|R_N \cup \hat{R}_N|}{R_N}$$

其中  $R_N$  为数据集中用户实际喜欢的物品， $\hat{R}_N$  为推荐算法推荐的物品。与根式均方误差和平均绝对值误差不同，它直接从推荐的效果上进行了评估。

为了客观评估我们模型的性能和推荐效果，我们在我们关于阅读的数据集上实现了许多现有流行的推荐算法，并将评估结果进行了横向对比。进行横向对比的几个推荐算法为：k 近邻 (kNN) 算法（包括基于用户和基于物品的两种方法）、奇异值分解 (SVD)、和非负矩阵分解 (NMF) 的方法。他们在传统的机器学习算法中脱颖而出，并且被广泛的应用与各种任务之上。

具体来说，在推荐算法这一任务中，k 近邻算法基于用户或者物品之间的相似性（取决于是基于用户还是基于物品），为我们推荐在隐式特征空间中距离更短的相关物品。奇异值分解是一种矩阵分解算法，将一个效用矩阵（用户  $\times$  物品）作为输入，每个值代表用户对某个商品的评价。奇异值分解将这个效用矩阵分解成一个隐特征的矩阵 U，表示每个隐特征强度的矩阵 S 和表示物品和隐特征的相似性的矩阵 V，根据提取出来的隐特征为用户推荐相似的结果。非负矩阵分解与奇异值分解一样，也是矩阵分解的算法，它将效用矩阵分解成两个每个元素都非负的矩阵 M 和 H，分别代表物品和用户与隐特征的相关关系，基于吸引力的权重进行物品推荐。

Algorithm	RMSE	MAE	Recall@20	Recall@50
Item KNN	0.714	0.522	0.054	0.129
User KNN	0.680	0.491	0.054	0.129
SVD	0.694	0.506	0.033	0.092
NMF	0.711	0.545	0.029	0.071
NN	0.602	0.417	-	-
VAE	-	-	0.179	0.188

表 2.1 各种推荐算法的对比结果

表格2.1展示了我们的实验结果。对于评分预测而言，NN 与其他模型相比具有更低的损失，而从推荐效果而言（即召回率），VAE 模型相较其他模型有着更高的召回率。总体而言，NN 与 VAE 的结合与传统推荐算法相比产生了较大的优势。这是因为这些深度学习的模型充分发挥了大量数据和计算资源的优势，利用更多的模型参数学习到了数据背后的隐藏信息。



## 第三章 效果展示

本章对我们挑战杯项目的成果进行简单展示，主要包括账户功能部分、推荐功能部分、讨论功能部分、创作功能部分。

### 3.1 账户功能

#### 3.1.1 注册功能

下图为网站账户的注册页面，用户需要填入用户名、密码、昵称、性别、专业（单选）、书籍类型偏好（多选）等信息。注册成功后会发送提示：“成功注册！请跳转至/login/登录页面进行登录”

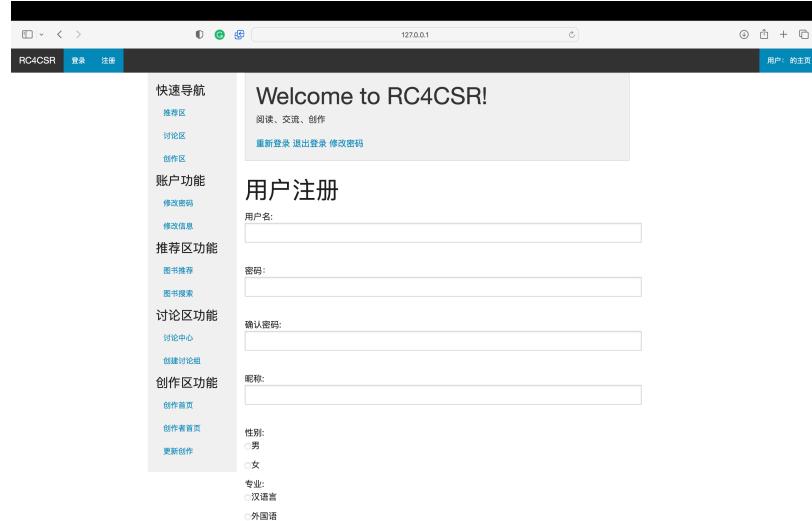


图 3.1 注册页面

#### 3.1.2 登录功能

下图为网站账户的登录页面，用户需要填入用户名和密码进行登录。登录成功后会跳转到用户主页，失败会发送提示：“用户名或密码错误！”

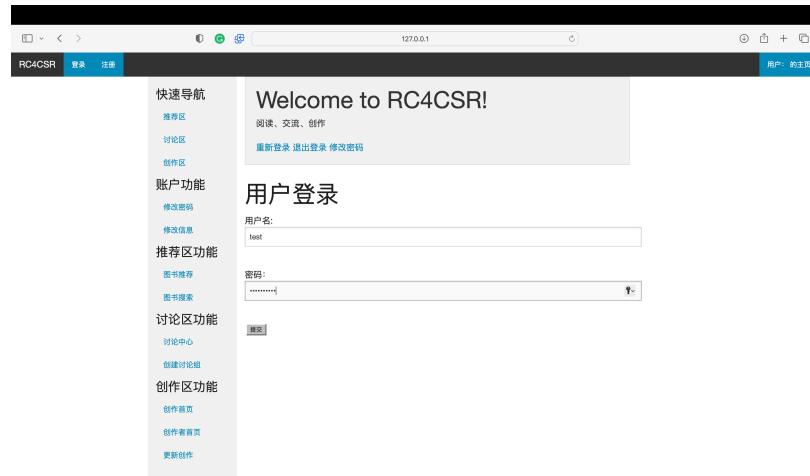


图 3.2 登录页面

### 3.1.3 用户主页

登录后或点击右上角按钮即可进入当前登录用户的主页。主页会展示性别、专业、书籍类型偏好等基本信息。

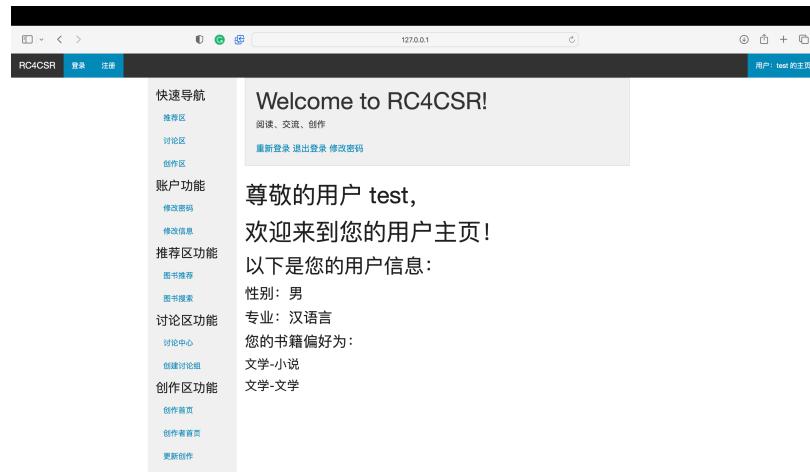


图 3.3 用户主页

### 3.1.4 修改信息功能

下图为网站账户的修改信息页面，与注册页面类似。

### 第三章 效果展示

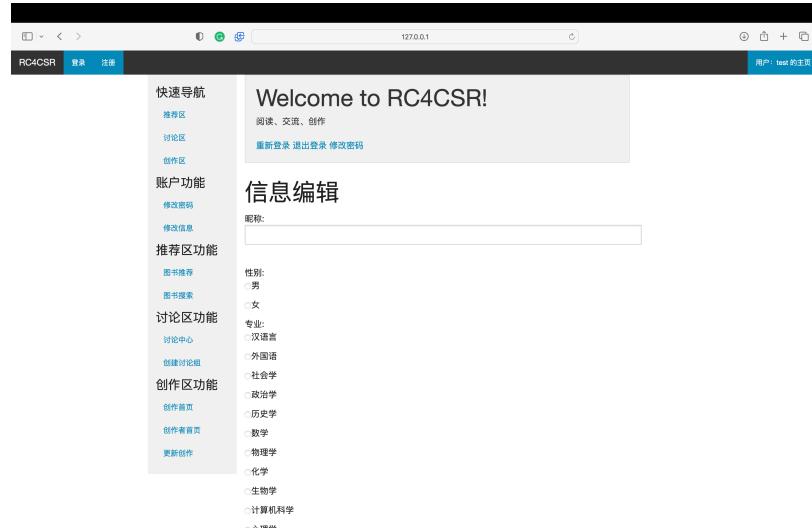


图 3.4 修改信息页面

## 3.2 推荐功能

推荐功能的原理在上一章节有详细介绍，本节主要展示推荐页面。

### 3.2.1 推荐页面

推荐页面会给出根据用户行为和偏好的图书推荐。点击图书名字可以进入图书详情页面

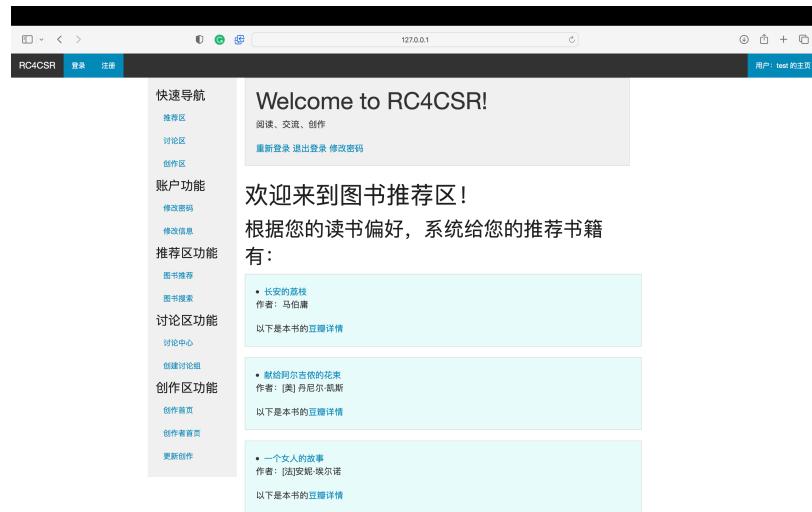


图 3.5 推荐页面

### 3.2.2 搜索页面

用户可以通过在搜索框内键入图书名字进行搜索。

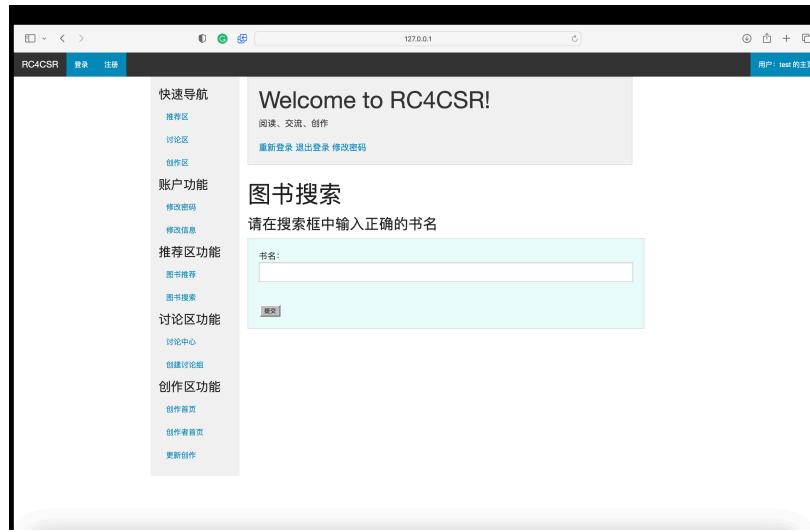


图 3.6 搜索页面

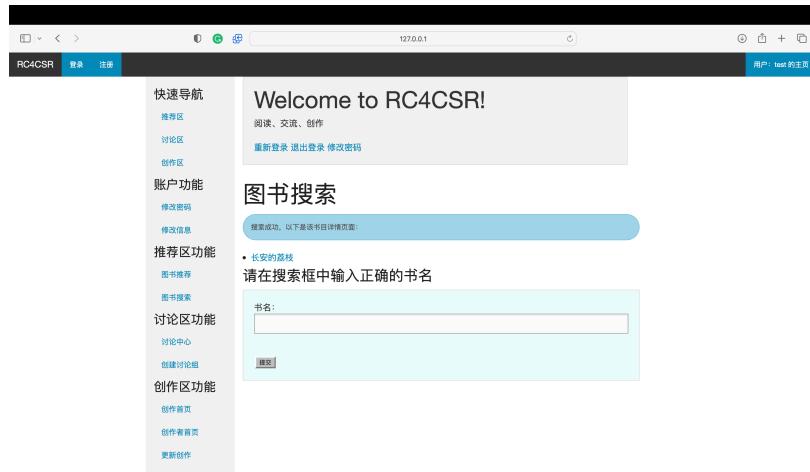


图 3.7 搜索成功

### 3.2.3 图书详情页面

单击图书名字可以进入图书详情页面，其中会展示图书名字、作者、评分等信息。

### 第三章 效果展示

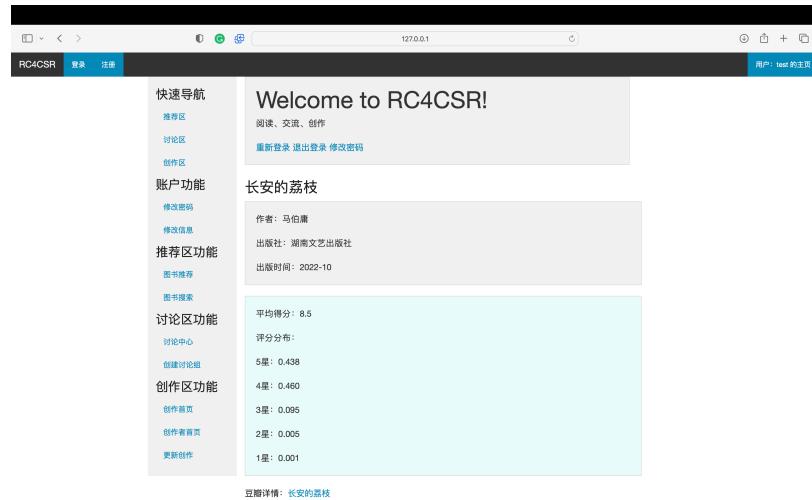


图 3.8 图书详情

## 3.3 讨论功能

讨论区主要以讨论组为组织形式。

### 3.3.1 讨论组注册功能

下图为讨论组注册页面，用户需要填入组名、讨论书目、发起者、成员（选填）等信息。

A screenshot of a web browser showing the '讨论组注册' (Discussion Group Registration) page. The left sidebar has the same navigation as in Figure 3.8. The main form consists of several input fields: '组名:' (Group Name), '中心书目:' (Central Book), '发起者:' (Initiator), and five optional fields for '成员一 (选填)' through '成员五 (选填)' (Members 1-5). There is also a '描述:' (Description) field at the bottom.

图 3.9 讨论组注册页面

### 3.3.2 讨论中心

下图为网站的讨论中心，其中包括了目前所有的讨论组。

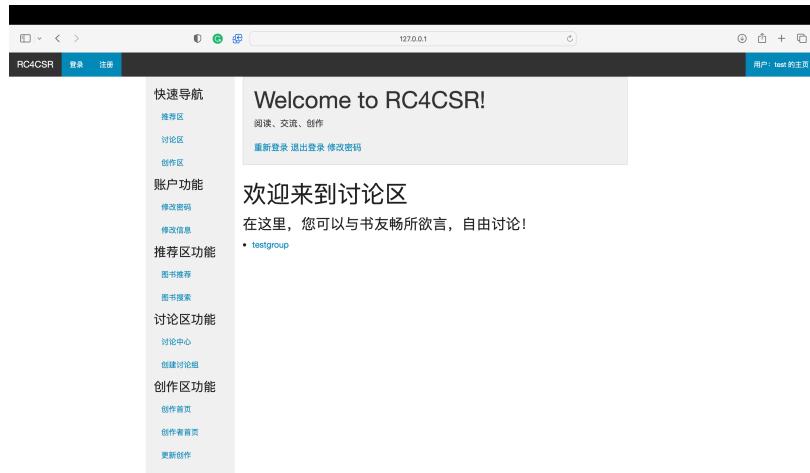


图 3.10 讨论中心

### 3.3.3 讨论组页面

下图为讨论组的主页，用户可以在其中发表评论并进行点赞等操作。

每个评论有自己的 id，除了讨论组的第一条评论，对某个评论进行回复时需要填入被回复评论的 id。

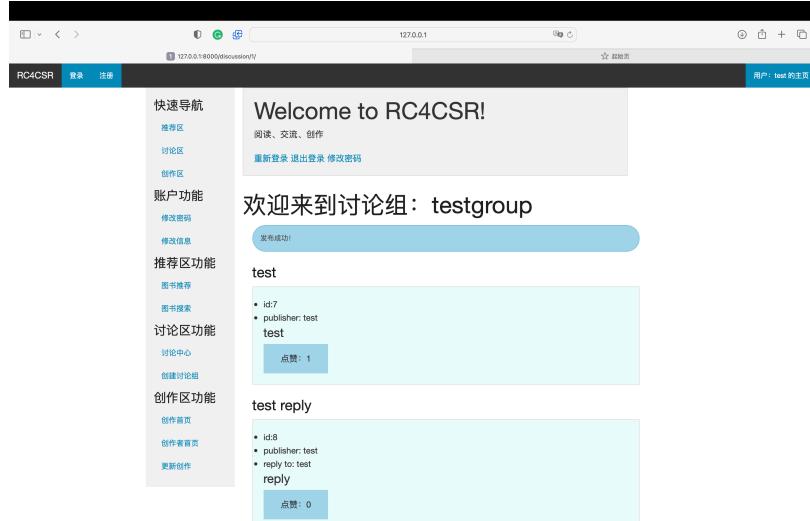


图 3.11 讨论组主页

## 3.4 创作页面

### 3.4.1 创作者页面

下图为创作者的创作页面，创作者可以在这个页面开始新的作品并查看自己已有创作。

### 第三章 效果展示



图 3.12 创作者页面

#### 3.4.2 更新页面

下图为作品的更新页面，创作者可以在这个页面为作品添加新的章节。



图 3.13 更新页面

#### 3.4.3 作品主页

单击作品名字会进入作品的主页，其中列出了各章节的名字。

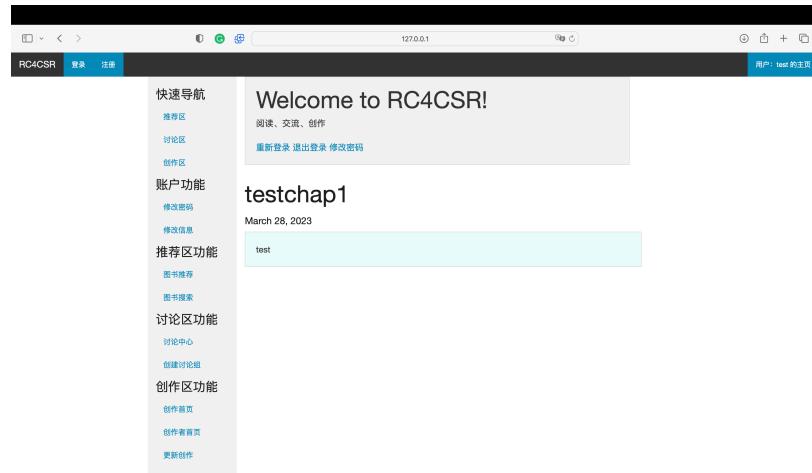


图 3.15 章节主页

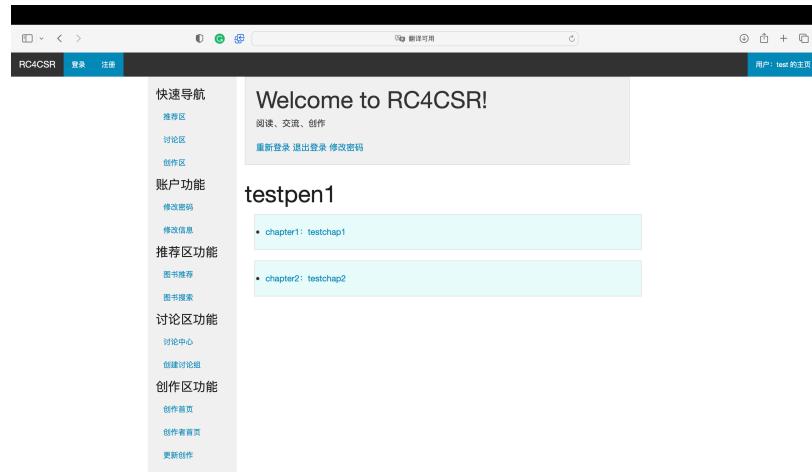


图 3.14 作品主页

单击章节名字会进入章节主页，其中列出了章节名字和章节正文。

#### 3.4.4 创作中心

下图为网站的创作中心，用户可以在创作中心浏览并对喜欢的作品点赞。

### 第三章 效果展示

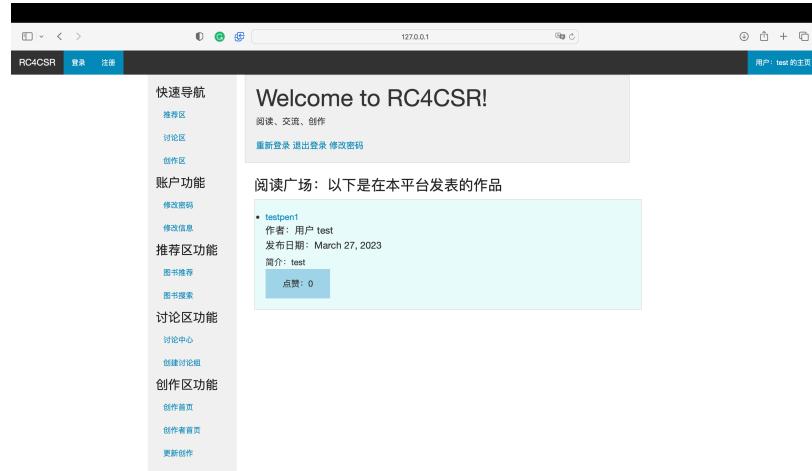


图 3.16 创作中心



## 第四章 开发进程

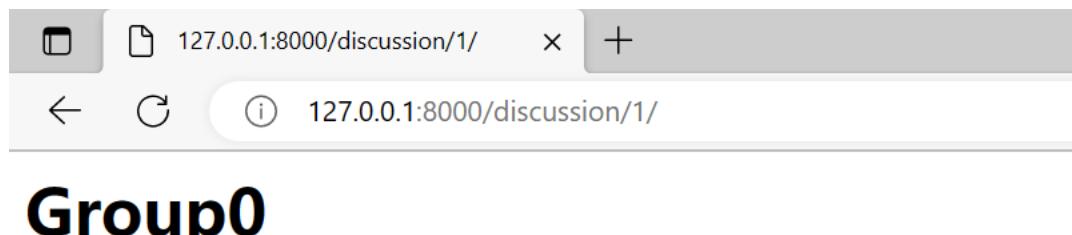
在本章，我们将简要回顾该项目的开发历程，以及项目产品已有的迭代升级过程。

### 4.1 服务端开发进程

#### 4.1.1 第一阶段：2023 年 1 月 7 日——2023 年 1 月 18 日

在项目的开始阶段，我们复习了 Django 的官方文档，初步完成了项目代码和文件的组织架构，并细化了分工。在我们初步搭建系统的阶段，我们将功能模块分为：系统（后优化为“账户”）模块、推荐模块、交流模块和写作模块。

具体来说，我们用非常短的时间，完成了讨论组、讨论记录、作品和章节等重要类（class）的编写和数据库创建，也完成了用户注册的简单实现。在前端，我们完成了风格简朴的前端页面编写，功能上打通了前后端的数据传输，用户可以在浏览器读取所有已经更新的内容。



### 随便讨论

- publisher: ZHC
- **这是第一条讨论**
- likes: 0

图 4.1 讨论区前端 v1.0

#### 4.1.2 第二阶段：2023年1月29日——2023年2月4日

在春节假期后，我们迅速投入工作，很快基本完成了系统组全部功能模块的初步实现。

具体来说，在图书推荐模块，我们完成了图书详情页的初步编写，并实现了图书搜索的功能。在我们的设计中，我们特别保留了用户的每一条搜索记录，作为用户行为数据来丰富我们的用户画像，从而做到更匹配的推荐。

在讨论区，我们更新了用户写入的功能，用户此时不仅可以查看已有评论，也可以发表自己的评论、创建评论组、以及点赞其它用户的评论。

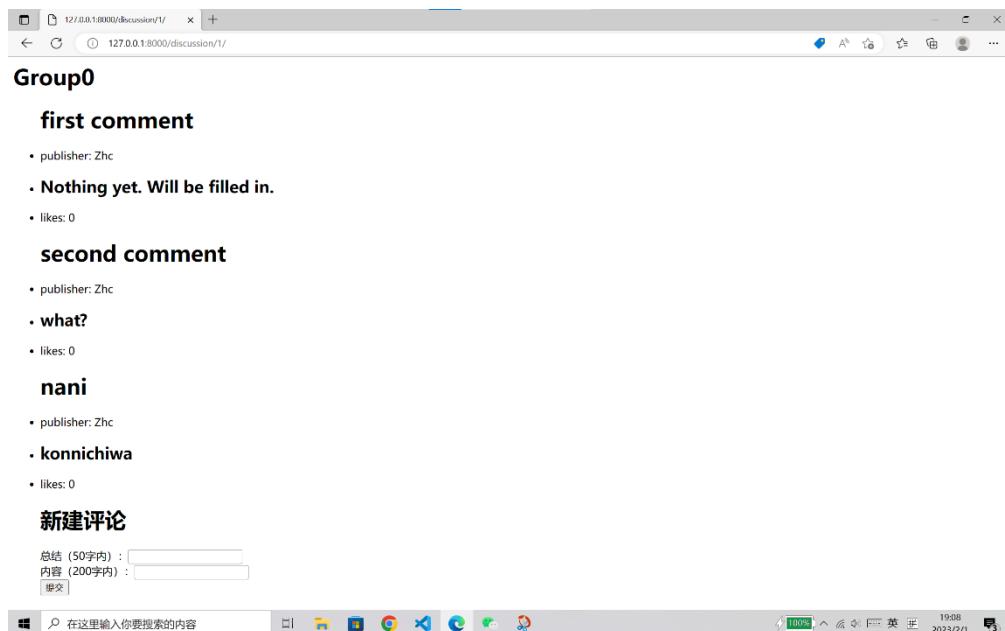


图 4.2 讨论区前端 v2.0

在创作区，我们设计了创作者首页，以及创建新作品和更新已有作品新章节的不同机制。经过调整，用户更新在操作上的便利程度和原先相比大大增加。

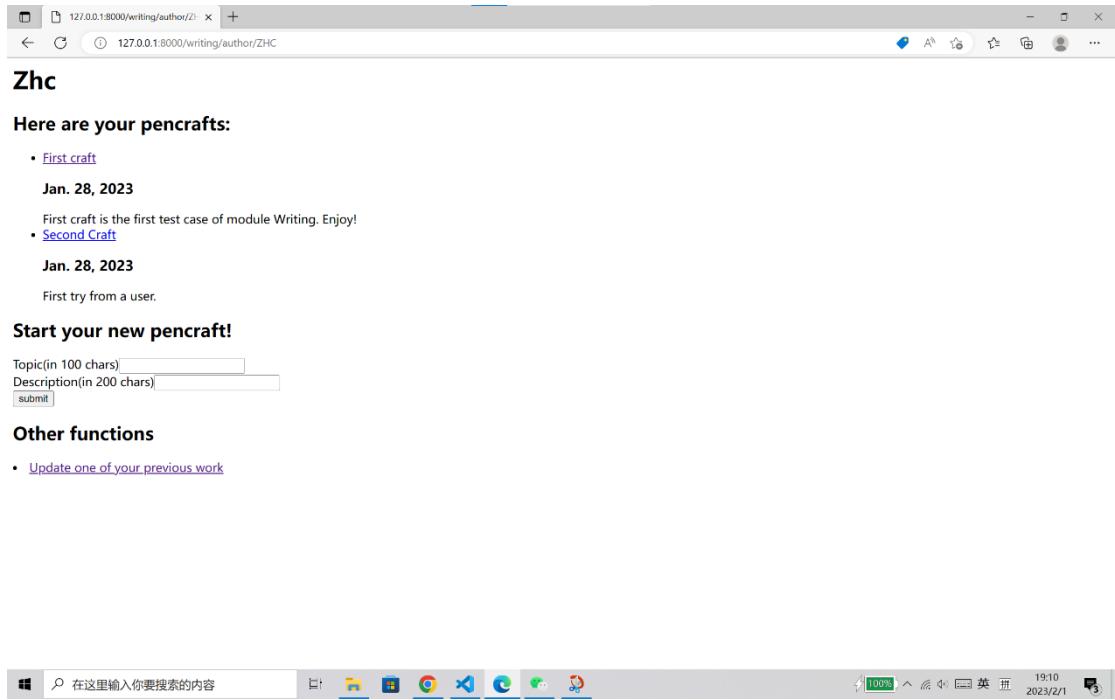


图 4.3 创作者主页前端 v1.2

另一方面，我们将原先“系统”模块中的功能拆分，迁移其中重要功能到“账户”模块。我们进一步完成了账户的注册与登入登出机制，在用户尝试越权访问时会进行强制跳转，并详细计划了其它的权限隔离措施。

#### 4.1.3 第三阶段：2023 年 2 月 5 日——2023 年 2 月 18 日

完成了全部基本功能后，我们主要进行了大范围的优化，并和机器学习组进行对接，将深度学习算法接入我们的推荐模块。

在安全性的优化上，我们对每个页面的访问权限进行了细致的界定，并对不同权限的用户访问内容做了区分。我们还实现了用户信息的修改，以及密码复杂度检查等小功能，进一步提高了安全性。在美观性的优化上，我们尝试了不同复杂度的前端实



图 4.4 用户主页前端 v2.0



图 4.5 登录界面前端 v3.0

现方案。从先前的白板前端网页，先加入了颜色与区域划分，然后加入了 Foundation

架构的按钮、下拉框等各种控件。网页风格从“古典互联网”论坛风格逐渐演化为现代前端页面风格，美观性和操作性得到飞跃式的提升。在功能上，我们也做了少量的进一步更新，比如增加了讨论区的跟帖评论功能，方便交流。

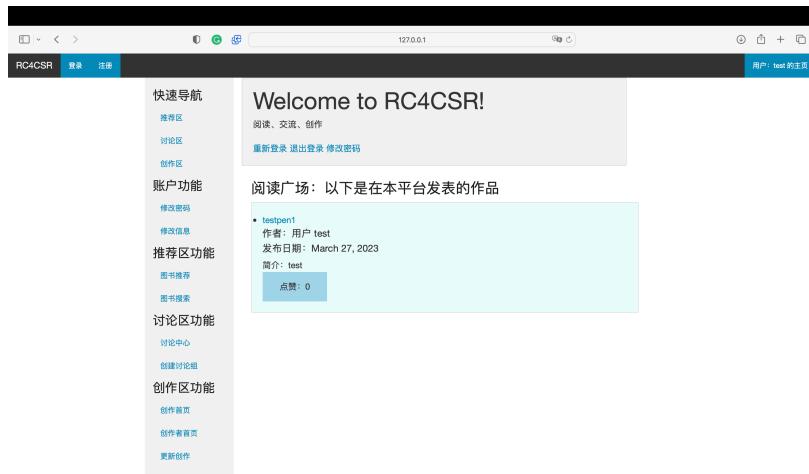


图 4.6 创作中心前端 v3.5 (最终版)

#### 4.1.4 第四阶段：2023 年 2 月 19 日——2023 年 3 月 5 日

在开学后，我们的主要工作在于更多地运行调试我们的系统，修复可能遇到的 bug，并将代码风格进行一定程度的优化，比如移除硬编码，从而在日后有功能扩展需求时方便接入新功能。

## 4.2 推荐算法进程

### 4.2.1 第一阶段：2023 年 1 月 7 日——2023 年 1 月 18 日

在第一周的时间内，我们回顾了机器学习、推荐算法相关的知识，并结合实际情况与系统组确定项目的组织架构以及需要收集的用于推荐的信息。我们探讨了 KNN、协同过滤、矩阵分解等诸多推荐算法，最后将本项目的主要算法定为 VAE 生成模型。

在第二周的时间内，我们主要为收集数据集做准备。我们需要搭建一个自己的图书库，但从无到有收集图书信息十分困难，因此我们从豆瓣网站上爬取了一定量不同类别的图书信息，作为自己的初始库。但此后我们发现一个更为棘手的问题——缺乏真实的基于我们收集图书信息的用户行为信息。几经尝试后，我们通过进行不同的用户偏好性假设，根据图书的平均评分与评分分布，采用一定的数学与概率方法模拟生成二百万条用户行为数据，并将其用于推荐训练。

#### 4.2.2 第二阶段：2023 年 1 月 29 日——2023 年 2 月 4 日

在第二阶段，我们主要进行算法层面的研究。用将近一周的时间初步搭建好 VAE 框架，并作效果评估。根据指标不断调整模型架构与数据集生成方法。但是我们意识到，在实际推荐过程中，无法收集到足够的新用户行为信息进行推荐，面临“冷启动”的问题。因此，我们加入 DNN 网络，根据用户自身的偏好等特征而这些信息是注册时收集到的，以此来克服 VAE 的冷启动问题。

#### 4.2.3 第三阶段：2023 年 2 月 5 日——2023 年 2 月 18 日

第三阶段中，我们主要与系统组进行对接，将算法部分接入系统框架中。与此同时，编写诸多信息收集接口。并实现模型自我更新功能，待用户数量与用户行为达到一定程度时，自动进行网络参数的优化。但因为受实际情况的限制，这一功能并未得到检验。

此外，我们对比分析了 KNN、SVD、NMF 等不同推荐模型，检验了本项目使用算法的优越性。

#### 4.2.4 第四阶段：2023 年 2 月 19 日——2023 年 3 月 5 日

开学后，我们的主要工作在于梳理代码结构，进行相应标注，使整个项目更为体系化。同时撰写项目报告。



## 第五章 总结与展望

本工程实践项目旨在开发一款基于深度学习的 Web 应用，提供多项便捷服务，包括推荐书籍文献、评论与互动、发布自己的作品等。我们的系统可以帮助大学生阅读交流，扩大社交圈子，提高学术能力，丰富校园生活，具有十分积极的社会意义。

通过本次工程实践项目的实施和完成，我们在以下几个方面做出了积极的贡献：

首先，我们独立完成了整个项目的开发和设计，实现了一个可用的、功能齐全的 Web 应用程序。在产品功能和性能方面，我们运用了多种技术手段，如数据采集和处理、UI 设计和交互设计等，提升了应用程序的实用性和易用性。

其次，我们注重团队合作和沟通，充分发挥了每个成员的专业优势和协作能力，实现了项目开发过程中的顺利推进。在工程实践过程中，我们也意识到了团队合作的重要性，并学习到了如何更好地协作和沟通，为以后的工作打下了坚实的基础。

最后，在未来的发展方向上，我们认为在训练数据、算法和应用性能层面还有很大的提升空间。

本项目推荐系统的训练是基于生成的数据集，尽管通过诸多数学概率方法使其尽可能合理化，但这依旧不是社会中真实存在的数据，因此在测试效果与推荐效果上差强人意。我们期待当用户在系统上通过互动产生更多行为信息时，用新收集到的数据不断迭代训练，或许会产生更好的效果。

在算法层面，本项目使用的 DNN 与 VAE 模型存在一个转换点，即用户在互动情况达到一定水平（阈值）时由 DNN 推荐转换为 VAE 推荐。但关于这一最优阈值仍缺乏实验验证。同时，我们也考虑采用更改权值的方式实现二者间平滑的过度，但受于训练数据的限制，这些需要等到平台用户达到一定规模方能进一步去调整。

此外，我们也将考虑扩展应用程序的服务范围和功能，加入更多有益的服务和功能模块。如果将来我们的项目成果能得到实际部署，我们将注重用户体验，通过用户调研和反馈，不断优化产品功能和性能，使其更加符合用户需求和使用习惯。我们相信，通过不断的改进和创新，这个应用程序将成为校园服务领域中的佼佼者，为广大学生带来更多的便利和服务。



## 参考文献

- [1] 邱锡鹏. 神经网络与深度学习[M/OL]. 北京: 机械工业出版社, 2020. <https://nndl.github.io/>.
- [2] VECTOR C T. Pkuthss: LaTeX template for dissertations in Peking University[EB/OL]. 2011 [2011-06-26]. <https://gitea.com/CasperVector/pkuthss>.
- [3] The web framework for perfectionists with deadlines | Django[EB/OL]. <https://www.djangoproject.com>.
- [4] SQLite Home Page[EB/OL]. <https://sqlite.org/index.html>.
- [5] Foundation Documentation | Foundation[EB/OL]. <https://get.foundation/frameworks-docs.html>.
- [6] BOGAARDS N, SCHUT F. Content-Based Book Recommendations: Personalised and Explainable Recommendations without the Cold-Start Problem[C/OL]//RecSys '21: Proceedings of the 15th ACM Conference on Recommender Systems. Amsterdam, Netherlands: Association for Computing Machinery, 2021: 545-547. <https://doi.org/10.1145/3460231.3474603>. DOI: 10.1145/3460231.3474603.
- [7] VANCURA V, KORDÍK P. Deep Variational Autoencoder with Shallow Parallel Path for Top-N Recommendation (VASP)[J/OL]. CoRR, 2021, abs/2102.05774. arXiv: 2102.05774. <https://arxiv.org/abs/2102.05774>.



## 致谢

在此，我们希望对本项目组的导师和开源社区表示深深的感谢，同时对北京大学和元培学院提供的资源和支持表示衷心的感激。

首先，我们要感谢本项目组的导师邹磊教授。在整个项目的研究和开发过程中，他给予了我们很多的指导和建议，他的严谨治学和科学的工作态度，让我们深受启发。通过与邹磊老师的交流，我们对科研界开发算法的通常流程，以及科研报告的撰写得到了更为深入的认识。没有他的帮助和支持，这个项目就难以取得今天的成果。

同时，我们要感谢开源社区。开源社区为项目提供了丰富的代码和资源，使得我们能够更快地开发出高质量的软件。在项目中，我们使用了许多开源软件和框架，这些软件和框架为项目的开发和运行提供了强有力的支持。

最后，我们要感谢北京大学和元培学院提供的资源和支持。作为“挑战杯”比赛的主办方，学校在我们进行比赛项目的过程中，给出了强有力的支持。

总之，在此我们要对导师、开源社区、北京大学和元培学院表示最真诚的感谢。他们的支持和帮助，使得我们能够在这个项目中学到很多知识和技能，收获了宝贵的经验。我们全组的四名成员都是大二本科生，对在高校进行科研并不能说是十分了解，通过参加本次“挑战杯”赛事，我们锻炼了我们的开发大型项目的代码能力、学术写作能力，对深度学习模型的分析设计能力，以及开发软件产品需要的设计能力。感谢“挑战杯”赛事提供的宝贵机会，相信这次锻炼能帮助我们在将来的学术道路上走得更远。