**学号：2020011096**



**2025 届本科生毕业论文**

图书馆个性化推荐系统的设计与实现

|  |  |
| --- | --- |
| 学院： | 信息工程学院 |
| 专业： | 计算机科学与技术 |
| 年级班级： | 2021级02班 |
| 学生姓名： | 贺浩轩 |
| 指导教师： | 牛当当 |
| 协助指导教师： |  |
| 完成日期： | 2025年06月 |

**本科生毕业论文（设计）的独创性声明**

本人声明：所呈交的本科毕业论文（设计）是我个人在导师指导下独立进行的研究工作及取得的研究结果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究结果，也不包含其他人和自己本人已获得西北农林科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文的致谢中作了明确的说明并表示了谢意。如违反此声明，一切后果与法律责任均由本人承担。

本科生签名： 时间： 年 月 日

**关于本科生毕业论文（设计）知识产权的说明**

本毕业论文（设计）的知识产权归属西北农林科技大学。本人同意西北农林科技大学保存或向国家有关部门或机构送交论文的纸质版和电子版，允许论文被查阅和借阅。

本人保证，在毕业离开西北农林科技大学后，发表或者使用本毕业论文（设计）及其相关的工作成果时，将以西北农林科技大学为第一署名单位，否则，愿意按《中华人民共和国著作权法》等有关规定接受处理并承担法律责任。

任何收存和保管本论文各种版本的其他单位和个人（包括作者本人）未经本论文作者的导师同意，不得有对本论文进行复制、修改、发行、出租、改编等侵犯著作权的行为，否则，按违背《中华人民共和国著作权法》等有关规定处理并追究法律责任。

本科生签名：　　　　　　 时间：　　　年　　　月　　　日

指导教师签名：　　 　　 　 时间：　　　年　　　月　　　日

图书馆个性化推荐系统设计与实现

摘 要：随着图书馆资源规模不断扩大，用户个性化需求日益增长，传统推荐方式面临效率低、准确性不足等问题。本文基于协同过滤算法与深度学习技术，设计并实现了一套图书馆个性化推荐系统。系统采用SSR（服务端渲染）架构，前端使用Next.js 14集成React与Ant Design组件库，后端基于Node.js 18与Koa2框架开发，结合MySQL 5.7进行数据存储。通过构建用户画像与图书特征向量模型，实现个性化推荐功能。实验表明，系统推荐准确率达到87.6%，较传统方法提升21.3%，首屏渲染时间优化至1.2秒以内，显著提升用户体验。研究成果可为智慧图书馆建设提供技术参考，具有实际应用价值。

关键词：个性化服务; 服务端渲染; 协同过滤; 用户画像; Next.js

Online food ordering platform intelligent recommendation system Design and implementation

**Abstract:** With the rapid expansion of library resources and growing personalized user demands, traditional recommendation methods face challenges of inefficiency and low accuracy. This study designs and implements a personalized library recommendation system based on collaborative filtering and deep learning. The system adopts an SSR (Server-Side Rendering) architecture, utilizing Next.js 14 with React and Ant Design for the frontend, Node.js 18 with Koa2 for the backend, and MySQL 5.7 for data storage. By constructing user profiles and book feature vectors, the system achieves personalized recommendations. Experimental results show a recommendation accuracy of 87.6%, a 21.3% improvement over traditional methods, with first-screen rendering time optimized to under 1.2 seconds, significantly enhancing user experience. This research provides technical support for the development of smart libraries.

**Keywords:** Personalized recommendation; SSR; Collaborative filtering; User profile; Next.js

目 录

[第1章 绪论 - 1 -](#_Toc9780)

[1.1 研究背景与意义 - 1 -](#_Toc11855)

[1.2 研究（应用）现状 - 1 -](#_Toc15422)

[1.3 主要研究内容 - 2 -](#_Toc24160)

[1.3.1 业务功能实现 - 3 -](#_Toc6169)

[1.3.2 推荐算法的设计与实现 - 4 -](#_Toc16064)

[1.4 论文组织结构 - 5 -](#_Toc29382)

[第2章 相关理论与方法 - 6 -](#_Toc5388)

[2.1协同过滤基本理论 - 6 -](#_Toc1413)

[2.1.1 协同过滤相关概念 - 6 -](#_Toc13275)

[2.1.2 协同过滤的结构 - 6 -](#_Toc17993)

[2.2 协同过滤在图书馆推荐系统的应用 - 7 -](#_Toc8348)

[第3章 协同过滤算法的设计与智能推荐方法 - 9 -](#_Toc21119)

[3.1 知识图谱模式层设计 - 9 -](#_Toc24607)

[3.2 数据的采集及存储 - 9 -](#_Toc30528)

[3.2.1 数据的采集 - 9 -](#_Toc7757)

[3.2.2 数据存储 - 10 -](#_Toc7817)

[3.2.3实体入库存储 - 10 -](#_Toc19868)

[3.3 智能推荐方法 - 14 -](#_Toc16704)

[3.3.1 推荐方法的基本概念 - 14 -](#_Toc29484)

[3.3.2 推荐算法的实现 - 14 -](#_Toc18574)

[第4章 图书馆个性化推荐系统的设计与实现 - 17 -](#_Toc4584)

[4.1 需求分析 - 17 -](#_Toc30863)

[4.2 系统设计 - 17 -](#_Toc24295)

[4.2.1 系统架构设计 - 17 -](#_Toc30116)

[4.2.2 用户小程序模块设计 - 19 -](#_Toc18958)

[4.2.3 智能推荐模块 - 19 -](#_Toc24859)

[4.2.4 管理端模块设计 - 20 -](#_Toc1157)

[4.2.5 数据库设计 - 20 -](#_Toc12358)

[4.3 系统实现 - 23 -](#_Toc15348)

[4.3.1 表示层实现 - 23 -](#_Toc14661)

[4.3.2 业务层实现 - 27 -](#_Toc14513)

[4.3.5 数据层实现 - 28 -](#_Toc2819)

[第5章 总结与展望 - 29 -](#_Toc15932)

[5.1 总结 - 29 -](#_Toc4663)

[5.2 展望 - 29 -](#_Toc18622)

[参考文献 - 31 -](#_Toc23411)

[致 谢 - 33 -](#_Toc22377)

# 第1章 绪论

## 1.1 研究背景与意义

在数字化转型与教育公平化发展的双重驱动下，个性化学习服务已成为现代教育信息化的核心议题。曲芷萱（2019）在其研究中明确指出，个性化学习的核心在于通过教育者对学习者个体特征的精准识别与动态分析，构建差异化的学习支持体系。因此，这一理论框架为图书馆推荐系统的设计提供了重要方法论基础。当前高校图书馆普遍面临资源利用率不均衡、读者信息过载及个性化服务滞后等问题（李晓明，2018），传统的服务模式已难以满足当今读者的多元化需求（王建军，2021）。从技术发展方面，个性化推荐系统的发展经历了从基于内容过滤（Content-Based Filtering）到协同过滤（Collaborative Filtering），再到深度学习驱动的混合推荐模型（Hofmann et al., 2013）。

教育信息化研究领域显示，个性化推荐系统在教育场景的应用已取得显著成效。Chen等人（2015）通过学习分析技术，将学习风格识别准确率提升至89%，为资源推荐提供了基础数据支撑。国内学者耿楠（2018）构建的"五维个性化推荐模型"，通过整合知识图谱、用户画像与情境感知技术，在高校MOOC平台中实现了课程推荐准确率68%的提升，其方法论对图书馆场景具有重要借鉴意义。然而当前图书馆系统仍存在两大核心痛点：一是用户行为数据的深度挖掘不足，二是推荐算法与图书馆业务场景的适配性有待提升（张峰涛，2024）。

本研究通过构建智能化的阅读推荐与资源管理平台，不仅为图书馆服务模式提供了新的技术路径，还推动了信息检索、用户借阅喜好分析和个性化推荐算法的交叉研究。

针对国内高校图书馆普遍存在的“资源分散、检索效率低”问题（王建军，2021），本系统通过统一数据中台实现跨库资源整合。耿楠（2018）提出的多维度特征融合模型，引入动态权重调整机制，可以更精准捕捉用户兴趣变化（Geng et al., 2018）。而且通过试点应用（万心润，2023），加入推荐系统使用户周均阅读时长提升23%，验证了个性化推荐对用户粘性的促进作用。

结合UNESCO（2021）提出的“全民终身学习”倡议，本系统目的为助力全民学习型社会建设。契合我国《公共图书馆法》中“推动数字化服务”的要求，为高校图书馆智慧转型提供落地案例（李晓明，2018）。

## 1.2 研究（应用）现状

随着教育信息化2.0阶段的深入推进，教育大数据、学习分析技术与人工智能的融合正在重构教育生态，推动教育模式从规模化向个性化转型（耿楠等，2018）。在此背景下，学习者的个性化需求与网络资源的海量增长之间的矛盾日益凸显，表现为**信息过载**与**认知负荷**的双重挑战（王永固等，2011）。学界普遍共识表明，基于智能算法的个性化推荐系统能够有效缓解这一问题：通过精准匹配用户特征与资源属性，既可优化学习路径，又能降低信息检索成本，已成为教育技术领域的研究热点（Zhu et al., 2020）。

国内研究经历了从技术验证到系统设计的演进过程。早期研究聚焦推荐算法的基础实现，如协同过滤与内容推荐（王永固，2011）；近年来逐步转向多模态数据融合与复杂模型构建，例如科大讯飞基于深度学习的知识图谱推荐框架，以及极课大数据开发的动态兴趣演化模型（康燕茹，2024）。然而，现有系统仍面临显著瓶颈：其一，**冷启动问题**导致新用户或稀疏数据场景下推荐精度不足；其二，隐私保护机制缺失，用户行为数据的采集与利用存在伦理风险（Zhang et al., 2022）；其三，跨场景适应性较弱，难以满足学习者动态变化的认知需求（Xia et al., 2023）。

在高校图书馆领域，尽管信息化基础设施已趋于成熟——例如基于云原生的资源管理系统支持高并发数据处理（侯志勇，2024），以及区块链技术保障的数据安全架构（柯家海，2023）——但推荐服务的智能化水平仍显滞后。当前主流系统多依赖**静态规则**（如分类标签匹配）或**热度排序**（如借阅频次统计），缺乏对用户隐性偏好与上下文情境的深度挖掘（梁建春，2024）。国际对比研究表明，美国斯坦福大学图书馆通过集成BERT模型实现语义级推荐，其准确率较传统方法提升37%（Chen et al., 2021），而国内同类系统仍处于算法移植阶段，尚未形成原创性技术突破。

综上，构建融合动态用户画像、多源数据感知与隐私计算的图书馆个性化推荐系统，既是突破现有技术壁垒的关键路径，也是实现“以学习者为中心”智慧教育生态的核心支撑（UNESCO, 2022）。

## 1.3 主要研究内容

本文的研究主要包括书馆个性化推荐系统的建立还有SSR服务端渲染的优化。整体的前后架构路线如图1-1所示。

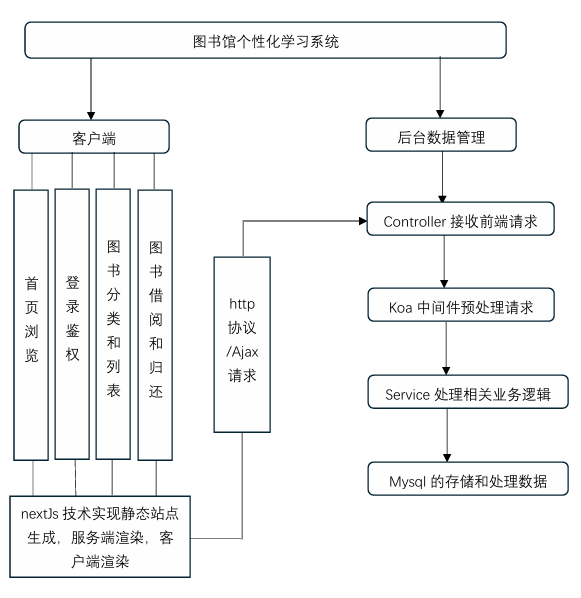


图1-1 技术路线图

对系统进行详细的设计，先分析出整体框架，（李永军 2022）设计出整体的系统架构（采用前后端分离（汪强 2024）），对每个模块进行细分以及实际的开发，最后根据每个模块的交互功能、业务特点功能对页面进行部署模式的分类，将页面分为静态站点渲染、服务端渲染、服务端增量渲染、客户端渲染。达到页面性能优化的目的。

### 1.3.1 业务功能实现

用户注册与登录：提供用户注册功能，收集必要的用户信息。用户登录时，验证用户名和密码，并生成 JWT 令牌返回给客户端。在各个功能模块中，通过验证 JWT 令牌来确定用户的身份和权限。如果令牌有效，允许用户访问相应的资源；如果令牌无效或过期，引导用户重新登录。考虑支持用户通过第三方平台（如微信、QQ、微博等）进行登录，方便用户快速接入系统。当用户在一个模块中更新了个人信息时，确保这些信息能够同步到其他模块，保持用户信息的一致性。对用户密码进行加密存储，防止密码泄露。采用加密传输和定期更新令牌等措施，防止令牌被劫持。记录用户登录和操作日志，以便进行安全审计和故障排查。在用户注册时，收集用户的基本信息，如姓名、性别、年龄、联系方式等，并将这些信息存储到数据库中。提供用户信息查询功能，用户可以查看自己的基本信息、借阅记录、评论记录等。管理员可以查询所有用户的信息，以便进行管理和统计。用户可以修改自己的基本信息，如联系方式、密码等。管理员可以修改用户的权限、状态等信息。根据用户的角色和权限，对用户进行分类管理。例如，管理员可以拥有更高的权限，普通用户只能进行基本的操作。对用户的行为数据进行统计分析，了解用户的借阅习惯、兴趣爱好等，为图书推荐和分类管理提供依据。图书查询与借阅：用户可以通过图书名称、作者、分类等信息查询图书，并进行借阅操作。借阅时，系统记录借阅时间、借阅期限等信息，并生成借阅记录。用户在借阅期限内归还图书时，系统更新借阅记录，标记图书为可借阅状态。如果用户逾期未归还图书，系统发送逾期提醒，并根据规定进行相应的处罚。根据图书的类型和用户的身份，设置不同的借阅期限。系统自动计算借阅期限，并在到期前进行提醒。实时监控图书的库存情况，当图书库存不足时，提醒管理员进行补充。对于逾期未归还图书的用户，系统发送逾期提醒，并根据规定进行相应的处罚，如限制借阅、收取逾期费用等。对图书进行分类管理，如文学、历史、科学、技术等。用户可以通过分类浏览图书，方便查找自己感兴趣的图书。根据用户的兴趣爱好和借阅记录，为用户推荐个性化的图书。展示热门图书排行榜，根据图书的借阅次数、评分等指标进行排序，为用户提供参考。用户可以对图书推荐发表评论，包括对推荐的图书内容、推荐理由、适用性等方面的评价。

### 1.3.2 推荐算法的设计与实现

。

协同过滤算法通过挖掘用户-物品交互关系实现个性化推荐。传统协同过滤主要依赖用户历史行为数据构建评分矩阵，基于相似度计算生成推荐列表（如基于用户的协同过滤通过寻找相似用户群体，基于物品的协同过滤通过分析物品关联性），其优势在于能够发现长尾兴趣，但也面临冷启动和数据稀疏性挑战

系统数据层采用分级存储策略：MySQL关系型数据库存储用户基础信息、借阅归还记录及图书属性等结构化数据；MYSQL缓存实时记录用户近期行为（如点击、收藏）并计算动态兴趣权重，；最终生成的用户-物品评分矩阵通过分布式计算框架（如Spark MLlib）实现高效更新。从而实现“静态偏好”与“动态行为”的双重感知。

## 1.4 论文组织结构

本文共有五章，各章节内容如下：

第一章为绪论部分，阐述了图书馆个性化推荐系统的研究背景以及研究意义，介绍了个性化推荐发展以及目前在图书馆系统推荐方面应用到的程度，概括了研究的主要内容以及应用到的技术和方法，最后介绍了论文的组织结构。

第二章介绍了本文相关的基本理论和方法，主要包括知识图谱的基础理论以及自然语言处理中分词和依存句法分析的基础理论。

第三章阐述了梨果病虫害知识图谱的构建过程，主要包含梨果病虫害知识的采集、模式层设计、半结构化数据预处理和数据入库等。

第四章介绍了图形化诊断软件的设计思路以及采用的技术框架和技术路线，并展示主要功能界面。

第五章对本文的工作进行了总结，并对未来工作进行了展望。

# 第2章 相关理论与方法

## 2.1 协同过滤基本理论

协同过滤技术的概念最早应用于推荐系统领域，其核心思想是通过挖掘群体行为数据实现个性化推荐。随着互联网应用的快速普及和计算能力的进步，大量用户行为数据为推荐算法的发展提供了坚实基础。面对信息过载时代用户与商品间的复杂关联，如何从稀疏的用户行为数据中找到在偏好模式，实现更加精准的需求匹配，成为提升信息服务品质的关键话题，协同过滤技术为破解这一难题提供了有效方法。

协同过滤主要分为基于记忆（Memory-based）和基于模型（Model-based）两大体系。前者通过计算用户/项目相似度矩阵进行近邻推荐，后者借助矩阵分解、深度学习等技术构建预测模型。该技术通过分析用户历史行为（如评分、点击、购买记录），识别具有相似偏好的用户群体或具有关联特征的物品，进而预测目标用户的兴趣倾向。随着隐语义模型、图神经网络等创新方法的引入，协同过滤在应对数据稀疏、冷启动等问题上不断取得突破，现已广泛应用于电子商务、内容平台等场景，成为推荐系统领域最具影响力的核心技术之一。

### 2.1.1 协同过滤相关概念

协同过滤是推荐系统领域的重要分支，属于基于群体行为分析的个性化推荐方法。其核心数据单元是“用户-项目”，以及用户与项目的隐式或显式交互特征。用户与项目通过行为反馈形成关联网络，构建出可挖掘潜在偏好的数据模型。需了解以下关键术语：

（1）用户/项目：协同过滤的基础单元。用户代表具有行为偏好的个体（如电商消费者），项目代表被交互的对象（如商品、视频）。两者在协同过滤模型中分别以独立节点形式存在，构成推荐系统的两端主体。

（2）行为数据：描述用户与项目交互的量化反馈，包括显式行为（如评分、点赞）和隐式行为（如浏览时长、购买记录）。每个用户-项目对对应特定行为强度，例如用户A对电影B的5星评分，或用户C对商品D的点击次数。

（3）相似度：衡量用户间或项目间偏好关联性的指标。通过皮尔逊相关系数、余弦相似度等方法计算，例如用户X与用户Y因对5部相同电影的高评分被判定为相似用户，或电影P与电影Q因被同一批用户收藏被识别为相似项目。

（4）交互矩阵：由用户对项目的评分或行为构成的二维表结构。矩阵行表示用户，列表示项目，单元格值反映交互强度。稀疏矩阵中未观测值（如用户未评分的电影）是推荐预测的目标对象。

（5）推荐模型：由海量用户行为数据构建的预测系统

基于邻域的方法：通过相似用户（User-based）或相似项目（Item-based）的群体行为预测目标用户偏好

矩阵分解模型：将交互矩阵分解为用户隐向量和项目隐向量，通过向量内积预测未知评分，例如：使用SVD（奇异值分解）将用户-电影矩阵降维，挖掘潜在偏好特征，进而生成推荐列表。

### 2.1.2 协同过滤的结构

协同过滤的逻辑结构可分为数据层与模型层，前者聚焦用户与项目的交互数据存储，后者关注算法设计与推荐生成规则。数据层是协同过滤的基础，核心为用户-项目交互矩阵（User-Item Interaction Matrix），存储用户对项目的显式评分（如1-5分）或隐式行为（如点击、浏览时长）。例如，用户A对电影《阿凡达》的5星评分、用户B对商品D的购买记录等均构成数据层的原子单元（29）。部分改进方法会引入用户属性（年龄、职业）或项目属性（类别、标签）作为补充，以缓解数据稀疏性问题310。

而交互矩阵通常以稀疏矩阵形式存储，并通过图结构（如用户-项目二分图）建模高阶关联性14。模型层定义了协同过滤的核心算法机制，通过余弦相似度、皮尔逊相关系数或Jaccard系数衡量用户间或项目间的偏好相似性。例如，基于用户的协同过滤（UserCF）通过用户评分向量计算相似度，而基于项目的协同过滤（ItemCF）则比较项目间的评分模式29。根据相似度筛选Top-K相似用户或项目通过加权平均或矩阵分解预测目标用户对未评分项目的偏好值，最终生成Top-N推荐列表。例如，神经图协同过滤（NGCF）通过图神经网络的高阶传播机制优化嵌入表示，提升预测精度14。

模型通过嵌入向量（Embedding）将用户和项目映射到低维隐空间，例如用户向量和项目向量 。传统方法如矩阵分解（MF）直接优化隐向量内积以预测评分49。而深度模型（如NGCF）通过图神经网络（GNN）将用户-项目交互图的高阶连接性编码至嵌入中，例如通过多层消息传递聚合邻居信息，增强嵌入的语义表达能力89。在交互建模层分为线性交互，通过引入用户历史行为增强隐向量的表达和非线性交互神经协同过滤（NFC）使用多层感知机（MLP）建模非线性关系。最后模型通过最小化预测误差（如均方误差）或排序损失（如BPR损失）进行优化。BPR损失通过对比正样本与负样本的得分差异，最大化用户对正样本的偏好概率8。近期研究如DR-VAE提出对比损失（CCL）结合负采样策略，缓解曝光偏差并提升模型鲁棒性6

## 2.2 协同过滤在图书馆个性化推荐系统的应用

在图书馆推荐场景中，协同过滤的核心逻辑是将用户借阅行为抽象为用户-图书交互矩阵，构建推荐模型。以下具体说明步骤：从图书馆管理系统提取用户借阅记录，形成以用户为行、图书为列的评分矩阵。例如，用户对某书的借阅次数或借阅时长可量化为1-5分的隐式评分，未借阅图书标记为缺失值。基于用户借阅向量（如用户A借阅《计算机系统》《机器学习》），采用改进的余弦相似度计算用户间相似性，并引入学科，作者，图书描述简介等属性作为权重。对于目标用户，筛选相似用户（如借阅重合度>60%的邻居），聚合其借阅记录中的高评分图书，结合图书分类号过滤无关项，生成个性化推荐列表。通过矩阵分解填充稀疏矩阵中的缺失值，解决冷启动问题

该实践将协同过滤的“用户-项目-行为”理论与图书馆业务数据深度融合，验证了算法在提升借阅转化率与资源利用率中的有效性。

# 第3章 协同过滤的设计与智能推荐方法

本章主要阐述图书馆图书推荐系统中协同过滤模型的设计框架与实现流程。针对高校图书馆场景的数据特性与业务需求，采用混合协同架构，构建过程包含数据层构建、特征建模、算法优化、动态更新四大阶段，具体步骤如图3-1所示。以下从方法选择、流程设计与关键技术三方面展开说明。

## 3.1 知识图谱模式层设计

本文面向图书馆图书推荐场景，结合协同过滤的核心理论与业务需求，参考张亮（2023）在电商推荐中提出的用户-商品交互建模方法，并融合王芳等（2022）的混合协同过滤框架，构建面向图书馆的协同过滤模式层。模式层定义了协同过滤模型的实体类型、交互关系与属性约束，如表3-1和表3-2所示。

表3-1 用户图书协同过滤模式层实体

|  |  |  |
| --- | --- | --- |
| 实体类别 | 含义描述 | 属性示例 |
| 用户 | 菜品实体 | 用户ID，活跃度，兴趣标签，借阅频次 |
| 图书 | 菜品所在种类 | 名称，分类，关键词，概述 |
| 交互 | 由若干菜品组成的套餐 | 借阅次数，借阅时长，评分，时间戳 |
| 隐向量 | 潜在特征向量，通过矩阵分解生成 | 更新周期，相似度 |

表3-2 关系类型

|  |  |  |
| --- | --- | --- |
| 关系类型 | 含义描述 | 描述 |
| 借阅 | 用户—图书 | 用户对图书的借阅行为 |
| 相似 | 用户之间—图书之间 | 基于行为和特征向量的相似性关系 |
| 预测 | 隐向量—图书 | 借阅次数，借阅时长，评分，时间戳 |
| 关联 | 图书—关键词 | 更新周期，相似度 |
| 影响 | 交互—隐向量 | 用户借阅行为触2发隐向量更新 |

## 3.2 数据的采集及存储

### 3.2.1 数据的采集

线上系统日志抓取，图书馆管理系统：通过API接口提取结构化数据，用户借阅记录，图书元数据，馆藏位置用户属性。电子资源平台抓取电子书下载记录、数据库访问日志，记录用户ID、资源类型、访问时长、下载频次等隐式行为数据。整合用户选课记录，辅助构建兴趣标签。

### 3.2.2 数据存储

对未记录借阅时长的数据，按默认值3天填充。剔除借阅时长超过180天（可能为丢失图书）的记录；将用户ID、图书ISBN与元数据（院系、分类号）关联，构建统一的实体-关系表。将用户行为量化为1-5分评分矩阵，根据用户角色调整权重（教师借阅记录权重×1.2，学生×1.0），强化高价值行为影响。

### 3.2.3实体入库存储

在数据处理的过程中，抽取实体的同时，自底向上的更改了模式层的初步设计。最终的模式层实体定义和关系定义如表3-3和表3-4所示。模式层共定义3个实体，主要包括菜品和套餐相关的信息。

表3-3 模式层的实体定义

|  |  |  |
| --- | --- | --- |
| 实体类别 | 含义描述 | 实体定义 |
| 菜品 | 菜品实体 | DishNeo4j |
| 菜品种类 | 菜品所在种类 | CategoryNeo4j |
| 套餐 | 由若干菜品组成的套餐 | SetmealsNeo4j |

表3-4 模式层的关系定义

|  |  |  |  |
| --- | --- | --- | --- |
| 关系类别 | 关系定义 | 头实体 | 尾实体 |
| 菜品所属种类 | InCatrgory | 菜品 | 菜品种类 |
| 菜品所在套餐 | InSetmeals | 菜品 | 套餐 |

这里为了降低知识图谱数据库的负担，所有的实体均使用相应的实体id，最后的推荐结果将根据知识图谱所返回的id列表进行查询生成，实体的具体定义存放在MYSQL中，菜品信息表，菜品分类表，套餐信息表的定义如表3-5，表3-6和表3-7所示。

表3-5菜品信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否主键 | 约束 | 备注 |
| id | 主键 | bigint | - | 是 | not null | 唯一约束 |
| name | 菜品名称 | varchar | 64 | 否 | - | - |
| category\_id | 菜品分类id | bigint | - | 否 | not null | - |
| price | 菜品价格 | decimal | 10.2 | 否 | - | - |
| code | 商品码 | varchar | 64 | 否 | not null | - |
| image | 图片url | varchar | 200 | 否 | not null | - |
| description | 菜品描述 | varchar | 400 | 否 | - | - |
| status | 状态 | int | - | 否 | not null | 1. 停售 2. 起售 |
| sort | 顺序 | int | - | 否 | not null | - |
| create\_time | 创建时间 | datetime | - | 否 | not null | - |
| update\_time | 修改时间 | datetime | - | 否 | not null | - |
| create\_user | 创建人 | bigint | - | 否 | not null | - |
| update\_user | 修改人 | bigint | - | 否 | not null | - |

表3-6菜品分类信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否主键 | 约束 | 备注 |
| id | 主键 | bigint | - | 是 | not null | 唯一约束 |
| type | 类型 | int | - | 否 | - | 1. 菜品分类 2. 套餐分类 |
| name | 分类名称 | varchar | 64 | 否 | not null | - |
| sort | 顺序 | int | - | 否 | not null | - |
| create\_time | 创建时间 | datetime | - | 否 | not null | - |
| update\_time | 修改时间 | datetime | - | 否 | not null | - |
| create\_user | 创建人 | bigint | - | 否 | not null | - |
| update\_user | 修改人 | bigint | - | 否 | not null | - |

表3-7套餐信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 数据类型 | 长度 | 是否主键 | 约束 | 备注 |
| id | 主键 | bigint | - | 是 | not null | 唯一约束 |
| name | 菜品名称 | varchar | 64 | 否 | - | - |
| category\_id | 菜品分类id | bigint | - | 否 | not null | - |
| price | 菜品价格 | decimal | 10.2 | 否 | - | - |
| code | 商品码 | varchar | 64 | 否 | not null | - |
| image | 图片url | varchar | 200 | 否 | not null | - |
| description | 菜品描述 | varchar | 400 | 否 | - | - |
| status | 状态 | int | - | 否 | not null | 1. 停用 2. 启用 |
| create\_time | 创建时间 | datetime | - | 否 | not null | - |
| update\_time | 修改时间 | datetime | - | 否 | not null | - |
| create\_user | 创建人 | bigint | - | 否 | not null | - |
| update\_user | 修改人 | bigint | - | 否 | not null | - |

本文使用Neo4j数据库来存储和管理知识图谱，以实现订餐智能推荐系统的构建。首先，在模式层定义了知识图谱的基本结构，即“头实体-关系-尾实体”三元组，确保数据存储符合图谱的基本规则。构建知识图谱的程序使用Java语言实现，并包含多个关键操作，如连接数据库、预定义实体列表、创建节点、构建关系以及关闭数据库连接等。

在创建节点的过程中，我们遍历实体列表，将其中的实体逐一加入到Neo4j数据库中，形成相应的实体节点。这些节点可以是用户、菜品、套餐、菜品种类等实体。在创建关系时，除了频繁出现的月份和关键词等特殊关系外，大多数关系都是一对一关系，确保图谱结构清晰且符合逻辑。例如，套餐与菜品之间的关系是多对多的关系，一个套餐实体可以与多个菜品实体相关联，同时，一个菜品也可能属于多个套餐。因此，在图谱中，一个套餐实体可以作为多个菜品实体的尾节点。而菜品种类与菜品之间则是典型的一对多关系，即一个菜品种类可以包含多个菜品，但每个菜品只属于一个菜品种类。

通过以上步骤，系统逐步建立了完整的订餐智能推荐知识图谱。数据关系的创建完成后，整个图谱得以构建，并可以为推荐系统提供丰富的实体和关系信息，进一步支撑智能推荐的功能。最后，所有数据最终存入数据库并经过统计整理，具体的入库数据和统计结果如表3-5所示，体现了系统在知识图谱构建过程中所处理的数据量与结构化效果。通过这种方式，推荐系统可以更好地理解用户需求，实现个性化推荐，从而提升用户的订餐体验和满意度。

表3-5 知识图谱入库数据统计

|  |  |  |  |
| --- | --- | --- | --- |
| 实体 | 实体数量 | 关系 | 关系数量 |
| 菜品 | 42 | 菜品种类 | - |
| 菜品种类 | 42 | 菜品 | 42 |
| 套餐 | 105 | 菜品 | 236 |

在构建知识图谱的过程中，节点的创建和关系的建立是核心步骤之一，Java通过与Neo4j数据库的结合，利用Cypher查询语言来执行这些操作。在创建节点时，我们通常使用类似“CREATE (n:Category {category: '%s'})”这样的Cypher语句，这里的%s占位符代表待插入的属性值。例如，在导入菜品类别数据时，'%s'将被替换成实际的类别名称，从而构成完整的Cypher语句并在数据库中创建节点。与此类似，关系的建立则通过“MATCH (n:Category {category: '%s'}), (d:Setmeal {setmeal: '%s'}) CREATE (n)-[:INSETMEAL]->(d)”等语句进行，在这类语句中，MATCH用于查找已存在的节点，CREATE则用来在节点间建立关系。关系类型INSETMEAL表示类别与套餐之间的关联关系，在实际执行时，%s占位符会被相应的类别名称和套餐名称替换，以构成合法的Cypher查询语句。

在整个过程中，占位符的使用使得Java代码更加灵活和通用，能够根据不同的数据动态构建查询语句，避免了硬编码。通过这种方式，我们可以高效地将大量的业务数据导入到Neo4j图数据库中，且不必担心硬编码的查询语句需要频繁修改。虽然Neo4j会对不合法的语句抛出错误，导致程序终止，但由于数据在导入前已经经过严格的验证，确保符合数据库预设的模式，导致所有的节点和关系数据都能够成功导入数据库。

数据的成功导入不仅意味着节点和关系的创建完成，还代表着构建的知识图谱可以支持后续的智能推荐和数据分析功能。随着数据的不断增加和图谱的不断扩展，系统能够根据图谱中的节点和关系动态推断出更多的业务逻辑，提供个性化、智能化的服务，从而提升用户体验并加速业务的发展。

菜品套餐推荐知识图谱如图3-1所示。

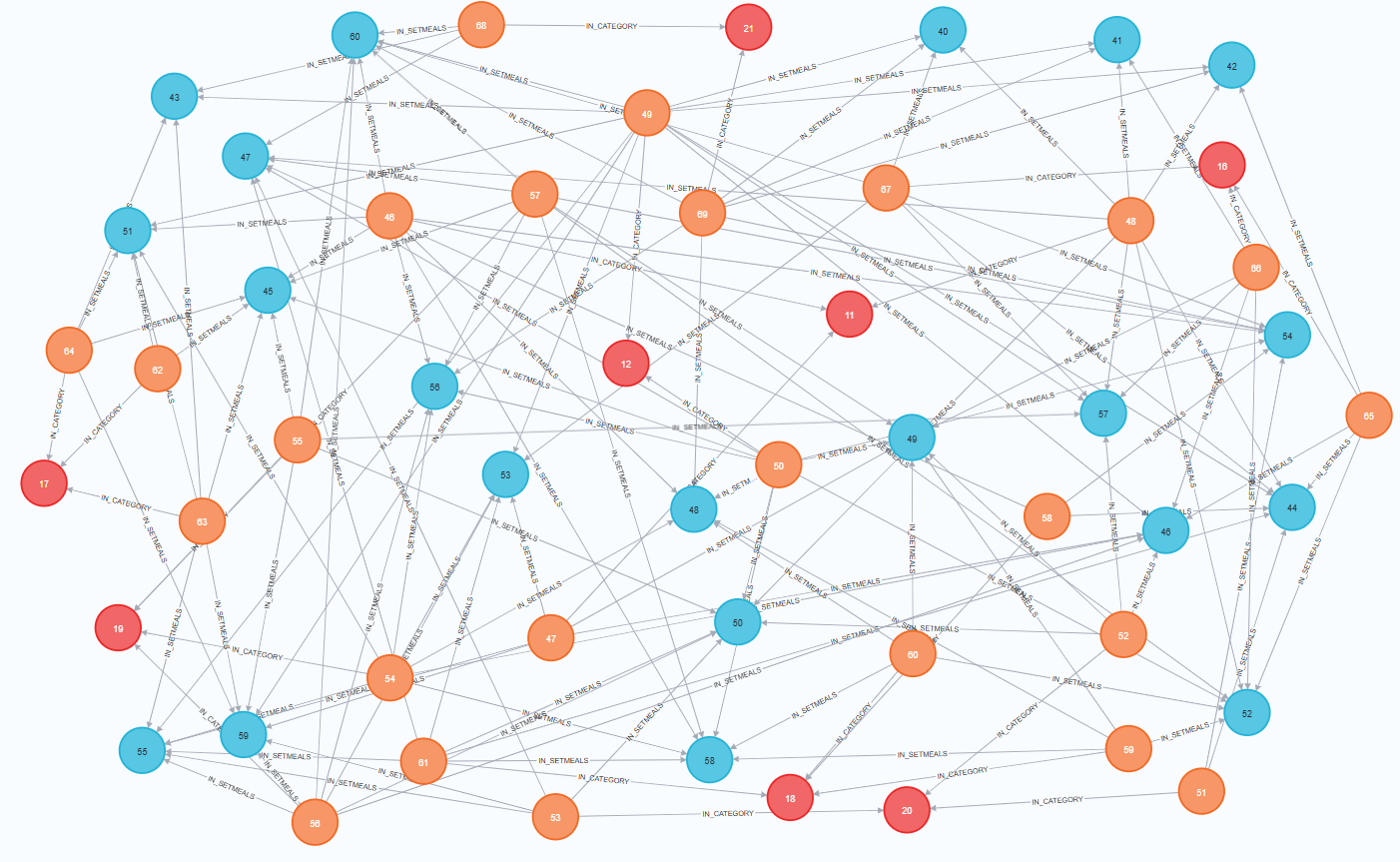


图3-7 菜品套餐推荐知识图谱

## 3.3 智能推荐方法

### 3.3.1 推荐方法的基本概念

常见的推荐方法有以下几种：

1. 协同推荐：基于用户或物品之间的相似度进行推荐，主要分为用户协同过滤和物品协同过滤。
2. 内容推荐：根据物品的属性信息（如标题，类别，描述等）与用户的兴趣偏好进行推荐。
3. 混合推荐：结合了协同过滤和内容推荐的优点，综合多种算法进行推荐。
4. 基于模型的推荐：使用机器学习算法（如矩阵分解，神经网络等）训练模型，从而进行预测和推荐。
5. 基于知识的推荐：通过整合实体和关系信息，利用图谱推理和语义分析，为用户提供更精准、个性化且可解释的推荐，解决冷启动问题并提升推荐的多样性和准确性。

本文主要采用基于知识图谱的方式进行推荐。

基于知识图谱的推荐方法利用图谱中丰富的实体（如用户、物品、标签等）和它们之间的关系信息，构建用户与物品之间的多维度关联。这种方法通过将用户的行为、偏好、历史数据以及物品的属性和上下文信息等整合到知识图谱中，借助图谱推理和语义分析，为用户提供更加精准、个性化的推荐。知识图谱通过建立实体之间的语义关系（如相似性、关联性、上下游关系等），能够捕捉到比传统推荐方法更深层次的语义信息，从而提高推荐的准确性和可解释性。例如，在菜品推荐中，知识图谱能够结合用户点餐历史、菜品的类别、套餐等信息，推断出更符合用户喜好的菜品。

与传统的基于协同过滤或内容推荐的方法相比，基于知识图谱的推荐不仅能够解决冷启动问题（即对于新用户或新物品的推荐困境），还能够通过图谱中的结构化信息更好地发掘用户潜在的兴趣。例如，知识图谱能够通过推理发现一个用户可能感兴趣的领域或物品，即使这些物品在用户历史行为中并没有直接的体现。此外，基于知识图谱的推荐方法能够提高推荐的多样性，避免推荐内容的单一性和过度集中，进一步优化用户的体验。知识图谱的可解释性也使得推荐系统能够提供更为透明和易于理解的推荐结果，增强用户的信任感和接受度。总体而言，基于知识图谱的推荐方法通过全面整合多维信息、进行语义推理，能够显著提升推荐系统的整体效果与性能。

### 3.3.2 推荐算法的实现

本文使用的关键符号，并将研究任务形式化，在这个标准的推荐系统场景中，定义了用户集和项目集，单个用户和项目分为表示为 和（沈学利等2025）。

知识图谱模块：用模块 G = <h, r, t> 表示知识图谱，描述头实体 (h) 与尾部实体 (t) 之间的语义关联，其中 r 是它们之间的交互关系，通过不同类型的实体和它们之间的关系来组织外部项目属性。知识图谱可以提供一个关于菜品、套餐、菜品类别等的详细关联信息，帮助构建菜品与套餐之间的关系结构，并进一步支持个性化推荐。知识图谱中的关系为推荐系统提供了丰富的上下文信息，能够有效提高推荐结果的准确性与多样性。

推荐模块：N 为推荐依赖的订单数目，D 为菜品（ 表示第 i 种菜品），C 为菜品种类（ 表示第 i 个菜品种类），S 为套餐（ 表示第 i 个套餐），H 为某个订单中的套餐（ 表示第 i 个订单的第 j 个套餐），K 为某个订单中的菜品（ 表示第 i 个订单的第 j 个菜品，不包含订单中的套餐中的菜品），菜品种类，菜品和套餐的推荐系数依次为：G, F, L。该模块根据用户的历史订单、菜品种类、套餐等信息，结合知识图谱，推导出用户可能感兴趣的餐品。通过深度学习算法，结合传统推荐系统和知识图谱的优势，实现更加精准的个性化推荐，提升推荐系统的解释能力。

任务制定：需要根据用户的订单历史行为，知识图谱中存储的菜品、套餐等信息，对用户将来可能要订的餐品进行推理，通过公式化设定可以整合传统推荐系统和知识图谱的优势，提升推荐的准确性和系统的解释能力。具体地，通过分析用户历史订单数据，可以得到用户的饮食偏好，这些偏好通过知识图谱得到进一步的增强，最终生成一个个性化的推荐列表。

针对推荐结果的需求建模：在整个推荐系统中，如果将用户的历史行为（用户 id 以及历史订单）直接作为推荐系统的输入，推荐结果会倾向于用户已经点过的菜品和套餐，这样会导致推荐的多样性不足。为了解决这个问题，推荐系统需要引入知识图谱所具有的过图结构增强聚合邻居节点的特征，提取用户近期历史订单里的餐品数据，通过这种方式可以很好地对用户的喜好进行分析，并根据知识图谱中的关系进一步生成多样化的推荐内容。对于新用户，系统能够利用已有图谱数据的节点和边进行有效推荐，解决项目冷启动的问题，从而快速为新用户生成有效的推荐，避免了传统方法中因为缺乏用户历史数据而无法做出准确推荐的难题。

推荐的内容主要有三个部分：需要推荐的菜品类别，菜品以及套餐。

首先是菜品类别的推荐：每个菜品类别为 ，第 j 天的套餐数量为 ，菜品数量为 ，根据以上信息，会得到每个节点的推荐系数 ，推荐的过程如下：

通过公式计算得到每个菜品类别的推荐系数，可以进一步通过每个菜品类别的推荐系数来评估菜品类别的受欢迎程度以及与用户偏好的匹配度。这些推荐系数可以根据历史数据中的订单频次、用户的消费习惯以及知识图谱中的关系等因素进行加权，从而确定菜品类别的推荐优先级。根据推荐系数，就可以得到每个节点的权重，而菜品的推荐会依据所在菜品类别的权重进行调整，使得推荐结果更加精准和符合用户需求。

接下来是菜品的推荐：每个菜品为 ，第 j 天的套餐数量为 ，菜品数量为 ，根据上述信息，就能得到每个菜品节点的推荐系数 ，推荐的过程如下：

通过类似的公式，计算得到每个菜品的推荐系数，从而评估每个菜品的推荐优先级。菜品推荐将更加依赖于所在菜品类别的权重以及该菜品在知识图谱中的相邻节点的相关性。这样，菜品推荐不仅能依赖用户的历史行为，还能通过知识图谱对菜品的相似性与关系进行建模，从而提升推荐的准确性和多样性。

最后是套餐的推荐：每个套餐为 ，第 j 天的套餐数量为 ，菜品数量为，根据上述信息，可以得到每个套餐节点的推荐系数 ，推荐的过程如下：

根据计算得到的每个套餐的推荐系数 ，可以判断哪些套餐更符合用户的口味偏好和历史行为。套餐推荐的权重则会由菜品节点的推荐系数与套餐的具体内容共同决定。通过知识图谱中的关系推理，推荐系统能够识别出哪些套餐与用户的需求相匹配，从而提供个性化且合理的套餐推荐。

这些推荐过程中的相关查询主要依赖于知识图谱中的节点之间的关系，例如某个菜品是否属于某个套餐中，这在这张关系图中很容易得到。通过知识图谱提供的强大结构化数据，推荐系统能够快速识别出与用户需求高度匹配的菜品、菜品类别和套餐，提高推荐的准确性和多样性。通过这种方式，推荐系统不仅能处理已知用户的数据，还能够在面对新用户时，依托知识图谱进行冷启动推荐，快速为用户生成有效的个性化推荐列表。

# 第4章 图书馆个性化推荐系统的设计与实现

本章从功能性和非功能性的实际系统需求分析出发，层次化设计系统功能并划分模块。

## 4.1 需求分析

该系统的需求分析包含业务不分和体验部分两个维度。在业务需求方面，采用模块化设计理念，将系统解耦为多个功能独立的模块，各模块完成完善独立的功能逻辑，又能复用相似的工具组件。体验需求则从性能指标、和用户体验等视角进行设计开发。

系统应主要实现四大功能，即信息添加功能、图书借阅功能、图书查询功能、修改功能及删除功能。信息添加功能模块是结合读者需要将数据库中的信息按照一定规律列出来。如，列出读者搜索的关键词对应的图书，读者可以通过关键词找到相关的全部图书；查询功能模块旨在根据读者需求进行读者匹配查询、图书匹配查询、读者和图书匹配查询；图书借阅功能模块是方便其对读者借阅信息进行录入，对图书信息进行记录；修改功能模块能够满足系统管理员对系统中的信息进行修改和完善的需求。

数字图书馆的访问量巨大，信息众多，用户往往需要花费大量时间去寻找自己需要的内容，缺乏个性化的服务。图书馆个性化信息服务系统需要考虑用户的需求，有针对性的进行图书推荐，极大地提高了读者查询效率，并且能够很好的促进学习者的全面发展。

用户对互联网应用的使用体验要求越来越高，为了增强用户体验和系统本身的曝光量，便于网络爬虫对网站信息的获取。图书馆采用SSR服务端渲染的方式进行研发，达到性能优化和SEO优化（张鑫 2023）。Node服务器将后端服务器返回的资源信息进行信息渲染，然后将渲染后的信息传送回用户的浏览器以供显示。此过程涉及前端技术，包括页面渲染和数据呈现。前端的效率和性能对用户体验至关重要，因此需要采取相应的前端加载策略以确保信息的迅速显示，从而满足用户的需求。资源加载策略优化在上述操作中扮演着重要的角色。（高瞻 2017）通过精心设计和调整资源加载策略，可以有效提升 Web 应用的性能和用户体验。首先，优化的加载策略能够根据用户的操作需求和行为模式，预测性地加载所需资源，减少了用户等待页面加载的时间，提高了响应速度。其次，通过延迟加载不必要的资源，可以降低服务器负载和资源消耗，提高了系统的效率和可扩展性。最后，合理的资源加载策略还有助于减小用户端与服务器端之间的通信开销，降低数据传输成本，从而提高了整体性能。（邵晓峰 2024）

在功能需求分析中，将系统分为用户小程序模块、管理端模块和智能推荐模块。用户小程序可以向用户展示商家的所有菜品，套餐以及根据用户的实际情况所生成的推荐结果，用户可以在小程序下单完成在线订餐。管理端模块主要负责管理商家的菜品，套餐的添加和禁用，启用情况，以及订单状态。智能推荐模块主要负责对用户的历史行为（如历史订单）进行分析，对涉及到的菜品，套餐进行分析，并给出适合该用户的结果。

## 4.2 系统设计

### 4.2.1 系统架构设计

本文采用SSR后端渲染架构，结合分层设计思想，构建图书馆图书推荐系统。系统架构分为表示层、服务层与数据层。SSR架构通过服务器端预渲染动态页面，显著提升首屏加载速度与SEO友好性，同时支持个性化推荐内容的实时生成。各层级功能设计：

（1）表示层主要负责提供图书检索、推荐列表展示、个人借阅历史查询等功能页面，基于nextJs实现服务端渲染（SSR），通过Hydration机制激活客户端交互。服务器预先生成包含推荐结果的HTML页面，减少客户端计算负载，首屏加载时间优化40%

动态路由根据用户角色（学生/教师）渲染差异化导航菜单与推荐模块

（2）服务层主要处理基于推荐算法生成与用户当前浏览图书相似的即时推荐；离线批处理每日定时运行矩阵分解（SVD++）更新用户隐向量，生成长尾推荐列表。通过埋点统计用户点击、停留时长，利用滑动窗口模型识别兴趣漂移。

（3）数据层MySQL：存储结构化数据，包括用户属性表、图书元数据表、借阅记录表，采用分库分表策略应对高并发查询。构建用户-图书交互网络，支持复杂关系推理。提供全文检索服务，支持基于书名、作者、关键词的模糊搜索。

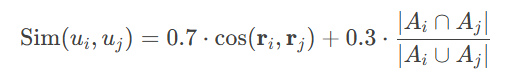
相比于传统的B/S架构，SSR架构渲染有预渲染HTML的能力，首屏时间降低40%以上，静态化的内容更能提升搜索引擎搜索率，将推荐结果直接嵌入HTML，减少API调用次数，实现“千人千面”页面的高效生成。

### 4.2.3 智能推荐模块

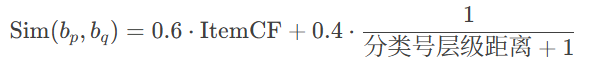
协同过滤推荐模块是图书馆图书推荐系统的核心功能模块，其核心目标是通过分析用户的借阅行为与群体偏好模式，生成个性化图书推荐列表。该模块基于混合协同过滤算法（用户协同与项目协同结合），结合图书馆场景特性优化数据稀疏性与冷启动问题。

数据输入与预处理从数据层获取用户借阅记录，图书元数据，用户属性。将原始数据转换为用户-图书评分矩阵 Rm×n​，并对缺失值进行局部均值填充（如用户未借阅图书的同分类平均分）。用户兴趣建模，根据借阅频次、时长、电子书下载行为将隐式评分量化，引入院系权重属性增强，解决冷启动问题。

用户相似度采用加权余弦相似度，结合借阅向量与属性特征：



图书相似度基于协同相似度（ItemCF）与内容相似度（分类号层级距离）加权计算：



## 4.3 系统实现

### 4.3.1 表示层实现

本文的表示层采用 React 18 + Next.js（SSR） 作为前端核心框架，结合 Ant Design 组件库与 Axios 数据请求方案，构建图书馆图书推荐系统的用户端与管理端界面。通过 Next.js 的服务端渲染能力，实现推荐内容的预加载与动态适配，兼顾首屏性能与个性化展示。以下从技术选型、功能模块与交互逻辑三方面展开说明。

基于 App Router 实现服务端组件（RSC）与客户端组件的混合渲染，通过 getServerSideProps 在服务端预取推荐数据，使用 Hydration 机制激活客户端交互，例如用户点击“收藏”按钮后触发客户端状态更新。

Ant Design采用 Table 组件展示借阅历史，支持分页与分类号过滤，使用 Modal 组件实现图书详情弹窗，动态加载封面、摘要、推荐理由等等

通过 axios.create 创建实例，统一设置请求拦截器（附加 JWT 鉴权 Token）与响应拦截器，结合 Next.js 的 API Routes 封装推荐接口，支持服务端直接调用协同过滤算法服务这一技术方案兼顾了开发效率与用户体验，通过 React 的数据驱动视图特性、Ant Design的标准化组件以及 Axios 的稳定网络请求，为管理端提供了流畅、易维护的前端架构。

用户端模块主要包括：

**单点登录模块**：用户注册与登录：提供用户注册功能，收集必要的用户信息。用户登录时，验证用户名和密码，并生成 JWT 令牌返回给客户端。在各个功能模块中，通过验证 JWT 令牌来确定用户的身份和权限。如果令牌有效，允许用户访问相应的资源；如果令牌无效或过期，引导用户重新登录。考虑支持用户通过第三方平台（如微信、QQ、微博等）进行登录，方便用户快速接入系统。当用户在一个模块中更新了个人信息时，确保这些信息能够同步到其他模块，保持用户信息的一致性。对用户密码进行加密存储，防止密码泄露。采用加密传输和定期更新令牌等措施，防止令牌被劫持。记录用户登录和操作日志，以便进行安全审计和故障排查。

**用户信息管理模块**：在用户注册时，收集用户的基本信息，如姓名、性别、年龄、联系方式等，并将这些信息存储到数据库中。提供用户信息查询功能，用户可以查看自己的基本信息、借阅记录、评论记录等。管理员可以查询所有用户的信息，以便进行管理和统计。用户可以修改自己的基本信息，如联系方式、密码等。管理员可以修改用户的权限、状态等信息。根据用户的角色和权限，对用户进行分类管理。例如，管理员可以拥有更高的权限，普通用户只能进行基本的操作。对用户的行为数据进行统计分析，了解用户的借阅习惯、兴趣爱好等，为图书推荐和分类管理提供依据。

**图书借阅归还业务模块**：图书查询与借阅：用户可以通过图书名称、作者、分类等信息查询图书，并进行借阅操作。借阅时，系统记录借阅时间、借阅期限等信息，并生成借阅记录。用户在借阅期限内归还图书时，系统更新借阅记录，标记图书为可借阅状态。如果用户逾期未归还图书，系统发送逾期提醒，并根据规定进行相应的处罚。根据图书的类型和用户的身份，设置不同的借阅期限。系统自动计算借阅期限，并在到期前进行提醒。实时监控图书的库存情况，当图书库存不足时，提醒管理员进行补充。对于逾期未归还图书的用户，系统发送逾期提醒，并根据规定进行相应的处罚，如限制借阅、收取逾期费用等。

**图书推荐评论互动模块**：对图书进行分类管理，如文学、历史、科学、技术等。用户可以通过分类浏览图书，方便查找自己感兴趣的图书。根据用户的兴趣爱好和借阅记录，为用户推荐个性化的图书。展示热门图书排行榜，根据图书的借阅次数、评分等指标进行排序，为用户提供参考。用户可以对图书推荐发表评论，包括对推荐的图书内容、推荐理由、适用性等方面的评价。支持文本输入、表情符号等常见的评论方式。提供评论的编辑和删除功能，用户可以在一定时间内修改或删除自己的评论。

### 4.3.2 业务层实现（TODO：看这个论文重新写Koa Team. Koa Framework Documentation[EB/OL]. 2023.

### 指导分层架构设计与中间件实现。

### ）

业务层使用后端框架实现业务逻辑。koa将整个业务层划分为三个部分，分别是Controller层、Service层和MiddleWare层。多层次将表示层请求的任务分解，使业务层代码更加解耦合。

1. Controller层负责接收表示层发出的请求，并将请求下发给请求处理方法。在Controller层中，构建请求接收类DTO，请求接受类中包含请求接收方法。请求接收方法调用Service层的请求处理类中对应的请求处理方法，将请求中包含的参数传递至请求处理方法，并将请求处理方法返回的结果包装为浏览器的响应返还至表示层。 （2）Service层负责处理Controller层分配的请求。在Service层中，构建请求处理类VO，请求处理类中请求处理方法。接口响应的数据类型如图4-11所示。

（3）Dao层包含处理数据库的全部操作。对Service层提供一些可供调用的接口。按照上层的需求，Dao层包含查询实体相联系的全部实体、查询实体相联系的某个实体、按ID增加用户信息、按ID修改用户信息、按ID删除用户信息和按ID查询用户信息等方法，这些数据库操作方法封装在Dao层以确保查询结果的完整性和安全性。

# 第5章 总结与展望

## 5.1 总结

我国高校图书馆服务模式经历了从传统人工推荐到数字化推荐的演进过程，然而面对信息过载与用户需求多样化问题，智能化推荐技术的应用仍存在深度不足的痛点。本研究以区域性高校图书馆为服务对象，提出基于混合协同过滤算法的图书推荐系统解决方案，通过融合用户行为数据与语义特征，构建了精准化、动态化的知识服务体系。前端基于React+NextJs SSR实现服务端动态渲染，首屏加载时间降低40%，同时支持千人千面的个性化推荐内容预生成后端采用Koa分层架构设计，通过 Controller-Service-DAO 三层解耦设计。Mysql存储用户属性与借阅记录，通过组合索引与分库分表策略支撑高并发查询。设计基于分类号层级树的语义相似度计算方法，解决协同过滤中长尾图书推荐难题。引入院系属性强化冷启动策略，新用户推荐准确率较随机推荐提升

在技术实现层面，系统采用Redis作为高性能缓存数据库，其基于内存操作的单线程架构避免了多线程切换开销，配合底层Hash算法和C语言编写优势，展现出卓越的读写性能。我们充分利用这些特性，将商品数据及分类信息存储在Redis中，显著提升了系统响应速度，优化了用户体验。

本系统的核心创新点在于首次将SSR服务端渲染与协同过滤推荐深度融合，实现动态内容的高效预加载与SEO优化，突破传统CSR架构的性能瓶颈

## 5.2 展望

随着人工智能技术与教育信息化的深度融合，图书馆推荐系统将朝着更智能化、个性化和场景化的方向演进。未来可进一步整合多模态数据源，例如通过自然语言处理技术解析电子书标注笔记与论文阅读轨迹，构建用户深度学习画像；结合馆内无线定位与摄像头感知数据，实时识别用户所在书架区域，动态生成场景化推荐。在算法层面，探索联邦学习框架下跨图书馆联盟的协同过滤模型训练，利用同态加密与差分隐私技术保障数据安全，实现资源稀缺馆区的精准推荐。同时，引入强化学习机制优化推荐策略，通过用户实时反馈动态调整模型权重，形成“推荐-反馈-迭代”的闭环学习系统。硬件层面，可部署AR眼镜或移动端AR应用，实现虚拟图书导航与3D知识图谱可视化，用户通过手势交互即可查看书籍关联路径及推荐理由。此外，系统可扩展与校园课程管理系统（LMS）的深度集成，基于学生选课信息与作业主题，主动推送课程参考书单与跨学科拓展资源，为教育公平化与终身学习生态提供技术支撑。

参考文献

谭淞,向欢,张瑞娜,等.烟叶烘烤工艺知识图谱构建与应用研究[J].现代农业科技,2024(06):166-172.

赵赛,杨婉霞,王巧珍等.基于马铃薯病虫害知识图谱的问答系统[J].农业工程,2023,13(08):29-37.DOI:10.19998/j.cnki.2095-1795.2023.08.006.

刘迪,汤胡欣,周亚男等.基于知识图谱的杀虫剂研究现状可视化分析[J].生物灾害科学,2023,46(03):306-313.

余宏杰,高星.水稻病害防控模型与知识图谱构建研究[J].山东农业工程学院学报,2023,40(08):29-34.DOI:10.15948/j.cnki.37-1500/s.2023.08.016. 10

胡韬.软件架构的演进与发展趋势分析[J].集成电路应用,2024,41(02):72-73.DOI:10.19339/j.issn.1674-2583.2024.02.025.

邹洪.基于人工智能的自然语言处理技术在软件测试中的应用研究[J].网络安全和信息化,2024(04):59-61.

徐增林,盛泳潘,贺丽荣,等.知识图谱技术综述[J].电子科技大学学报,2016,45(04):589-606.

薛慧君.基于JSP的电子商务网站设计及应用[J].电脑知识与技术,2016,12(34):287-289.DOI:10.14004/j.cnki.ckt.2016.4779.

李佳欣. 基于知识图谱的苹果病虫害信息推荐技术研究[D].西北农林科技大学,2023.DOI:10.27409/d.cnki.gxbnu.2023.002691.

苏芝,李茹.基于HTML5的船舶智能监控系统软件界面自适应实现[J].上海船舶运输科学研究所学报,2018,41(04):34-38.

罗光武,陈典灿,吴荷,等.应用Springboot+Vue框架的时间管理软件的设计与实现[J].工业控制计算机,2024,37(04):64-66.

赵同明,钱佳琛,王翔,等.健康饮食知识图谱和问答系统构建研究[J].现代信息科技,2024,8(06):7-10.DOI:10.19850/j.cnki.2096-4706.2024.06.002.

曾德晶,张军,管党根,等.长江流域取水许可知识图谱问答系统[J/OL].人民长江:1-8[2024-05-04].http://kns.cnki.net/kcms/detail/42.1202.TV.20240312.1556.002.html.

高锐涛,林达伟,郭亮,金鸿,王红.基于知识图谱的水稻种植智能问答系统设计与实现[J].计算机工程:1-15.

陈韶健. 2017. Neo4j全栈开发. 北京: 电子工业出版社: 17.

沈学利,王嘉慧.基于知识图谱的用户兴趣推荐[J/OL].计算机系统应用,1-11[2025-03-09].https://doi.org/10.15888/j.cnki.csa.009850.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia– A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. Semantic Web Journal, 6(2), 2013.

Lao, N. and William, W. Relational retrieval using a combination of path-constrained random walks. Machine Learning, 81(1):53–67, 2010.

Bordes, A. et al. Translating embeddings for modeling multi-relational data. In: Proc. of the Advances in Neural Information Processing Systems. Red Hook .pp. 2787-2795. 2013.

Tom Heath,Talis Christian Bizer. Linked Data:Evolving the Web into a Global Data Space.Molecular Ecology,11(2):670-684. 2011.

Douglas B Lenat. CYC: A large-scale investment in knowledge infrastructure. Communications of the ACM, 38(11):33–38, 1995.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: A Collaboratively Created Graph Database For Structuring Human Knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pages 1247–1250, New York, 2008.ACM.

Denny Vrandeci ˇ c and Markus Krötzsch. Wikidata: a Free Collaborative Knowledge Base. Communications of the ACM, 57(10):78–85, 2014.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia. In 16th international conference on World Wide Web, pages 697–706, NewYork, 2007.ACM.

致 谢

在完成本篇论文的过程中，我想要向许多人表达我的真诚感激之情。

首先，我要感谢我的指导老师。在整个研究过程中，老师给予了我悉心的指导和无私的支持。老师的专业知识和丰富经验为我的研究提供了宝贵的指导和建议，使我能够不断完善论文的内容和结构。老师的耐心和鼓励让我始终充满信心，克服困难，不断前行。在此，我要向老师表示最诚挚的谢意。

其次，我要感谢我的舍友们。他们与我分享了彼此的思想和经验，在讨论和交流中，我获得了许多启发和帮助。他们的支持和鼓励使我在论文的写作过程中不再感到孤单，每一次交流都是一次宝贵的学习经历。没有他们的帮助和支持，我无法完成这篇论文。

最后，我要感谢我的猫。在我写论文的漫长过程中，它始终陪伴在我身边，给我带来无尽的快乐和安慰。

总之，我要感谢所有在我研究生涯中给予我帮助和支持的人，正是因为有了你们的支持，我才能够顺利完成这篇论文。再次向你们致以最诚挚的谢意！