**摘 要**

知识库管理可以对组织内部的信息、数据、文件等所有称之为“知识”的进行系统的管理，达到提高组织效率的目的。

本文设计了一个知识库管理系统，该系统的目的是可以让不同权限的用户来存储、获取及管理知识。知识库管理系统选择在java平台上采用B/S结构进行开发，基于ssm的架构模式，使用mysql数据库进行信息的存储，检索使用基于Lucene的搜索服务器Elasticsearch进行提供全文搜索，极大的提高了大数据量下的检索速度。系统还根据用户的不同行为习惯进行分析，采用User-CF的协同过滤算法提取出用户可能感兴趣的内容，然后对不同用户进行有针对性的内容推荐。系统还提供了后台管理界面，可以对系统的“知识”进行管理；也可以对用户进行管理，包括添加、删除和密码修改;管理员还可以对用户自己添加的“知识”进行审核，根据内容决定是否通过。

关键词：知识库，管理系统，全文搜索，协同过滤

**Abstract**

Knowledge base management can carry out systematic management of all the information, data and documents that are called "knowledge" within the organization, so as to improve the efficiency of the organization.

In this paper, a knowledge base management system is designed, which allows users of different privileges to store, access and manage knowledge. The knowledge base management system in Java platform using B/S structure, based on SSM architecture model, using MySQL database for information storage, retrieval using Elasticsearch search server based on Lucene to provide full text search, a large amount of data greatly improve the retrieval speed. The system is also analyzed according to the user's different behavior habits, and the User-CF collaborative filtering algorithm is used to obtain the content which the user may be interested in, and then the content of the recommendation is given to different users. The system also provides a background management interface, the system can manage "knowledge"; also can manage users, including add, delete and modify password; the administrator can also verify the users to add their own "knowledge", according to the contents of deciding whether or not to pass.

Keywords: knowledge base, management system, full text search, collaborative filtering

**目录**

**[1 引言](#_Toc22294)** [4](#_Toc22294)

**[1.1 研究背景及意义](#_Toc5763)** [4](#_Toc5763)

**[1.2 研究现状和发展趋势](#_Toc1411)** [4](#_Toc1411)

**[1.3 主要研究内容](#_Toc26009)** [5](#_Toc26009)

**[2 关键技术介绍](#_Toc24389)** [7](#_Toc24389)

**[2.1 java语言](#_Toc236)** [7](#_Toc236)

**[2.2 mysql数据库](#_Toc6822)** [10](#_Toc6822)

**[3 系统总体设计](#_Toc4442)** [12](#_Toc4442)

**[3.1功能分析](#_Toc25031)** [12](#_Toc25031)

**[3.2系统架构设计](#_Toc4692)** [15](#_Toc4692)

**[4 数据库设计](#_Toc7391)** [23](#_Toc7391)

**[4.1数据库连接配置](#_Toc25496)** [23](#_Toc25496)

**[4.2数据库逻辑结构设计和物理结构设计](#_Toc27534)** [24](#_Toc27534)

**[5 模块设计与实现](#_Toc20920)** [28](#_Toc20920)

**[5.1登录注册](#_Toc11721)** [28](#_Toc11721)

**[5.2 系统主界面](#_Toc5076)** [29](#_Toc5076)

**[5.3 资源内容](#_Toc16682)** [31](#_Toc16682)

**[5.3 个人信息](#_Toc28724)** [33](#_Toc28724)

**[5.4 后台管理](#_Toc32162)** [34](#_Toc32162)

**[6 系统测试](#_Toc1702)** [37](#_Toc1702)

**[6.1 测试意义和目标](#_Toc5346)** [37](#_Toc5346)

**[6.2 测试用例](#_Toc30406)** [38](#_Toc30406)

**[7 总结](#_Toc20989)** [40](#_Toc20989)

**[致谢](#_Toc2467)** [41](#_Toc2467)

**[参考文献](#_Toc25348)** [42](#_Toc25348)

**[附录](#_Toc341)**[： 43](#_Toc341)

**1 引言**

## **1.1 研究背景及意义**

在知识经济时代，对一个团队内部知识资本进行管理已经成为提高核心竞争力的必要之选。而知识库管理系统，是以互联网和信息技术为基础，帮助企业对异构环境下的知识资源进行明晰化、系统化的管理，定位拥有专门技能的员工，简历协作专家网络，使得所有员工能够快速方便地访问到所需要的知识资源。无论数据库、文档、政策、业务流程还是员工的个人技能都能得到高校的贡献利润。从而提高整体的应变和创新能力。

现在随着知识管理概念的深入人心，进行知识管理时间的行业、组织已经越来越多，除了传统的企业组织进行知识管理外，会有越来越多的政府机构、行业协会、项目组织、虚拟社团等进行知识管理。例如大型运动会的知识管理、环境保护的知识管理、大江大河治理的知识管理、各类基金会社会慈善机构的知识管理、行业的知识管理等。[1]

知识管理以成为企业管理中的一项基础管理，通过战略管理、营销管理、项目管理、人力资源管理等组织各方面的管理职能展现出来。更多人认识到了知识成为战略性资源的价值。

但是现在大部分知识管理系统都是针对与大型企业而建立的，而对于中小型团队没有必要或者没有能力去分出一部分人专门对系统进行管理，所以需要一个小型的、简单易操作的知识库管理来帮助团队管理知识。对团队进行知识的有效归纳和整理，可以实现团队的敏捷开发的目标，从而提高团队的工作效率，提高产出。

## **1.2 研究现状和发展趋势**

知识管理系统作为当前世界管理学科比较热门的研究领域，已经受到学术界和企业界的重视。知识管理以成为知识经济时代的重要课题。

知识管理是网络新经济时代的新兴管理思潮与方法，管理学者彼得·杜拉克早在一九六五年即预言：知识将取代土地、劳动、资本与机器设备，成为最重要的生产因素。受到1990年代的信息化(资讯化)蓬勃发展，知识管理的观念结合网际网络建构入口网站、数据库以及应用电脑软件系统等工具，成为组织累积知识财富，创造更多竞争力的新世纪利器。

在信息时代里，知识已成为最主要的财富来源，而知识工作者就是最有生命力的资产，组织和个人的最重要任务就是对知识进行管理。知识管理将使组织和个人具有更强的竞争实力，并做出更好地决策。在2000年的里斯本欧洲理事会上，知识管理更是被上升到战略的层次："欧洲将用更好的工作和社会凝聚力推动经济发展，在2010年成为全球最具竞争力和最具活力的知识经济实体。对于组织和个人，知识管理都已经成为伟大机遇和挑战。

国内知识库管理系统的发展：近年来知识库管理系统的结构研究成为知识管理系统研究的热点。主要涉及系统的基本结构、基于web环境下的企业知识管理系统框架、虚拟企业知识管理系统架构等等。李勇等（2002）提出，企业知识管理系统是一个由计算机基础平台、企业MIS系统、知识库、知识库管理系统、知识管理人员和系统用户组成的人机交互系统。

国外知识库管理系统的发展：知识管理已经日趋发展成熟。国外先进制造业企业走过了从孤立的CAX技术的应用到全面采用数字化的三维设计、虚拟装配、并行工程等技术，并开始全面实施设计制造的协同工作，最大限度地利用了行业内的资源和设计制造能力，而实现这种协同设计制造的基础保障就是底层的基于知识的协同管理平台。[2]

## **1.3 主要研究内容**

1、用户登录：

用户输入所使用的用户名和密码，判断用户是管理员还是普通用户，判断用户的合法性，如果是合法用户，则允许登录知识库管理系统显示首页；否则显示错误页面，点击跳转登录页面。

2、用户上传文档

用户可以根据系统所提供的分类按文件类型上传相关文件，并给文件添加相应的标签。

3、用户查找、预览、下载文档

用户可以对整个系统当中的文档以及用户自己上传过的文档进行搜索，显示匹配内容并支持所有文件下载。部分文件支持在线预览。

4、管理员

管理员登录后页面跳转至管理员界面，管理员可以对系统的普通用户进行修改密码以及删除用户操作，同时可以对分类文件进行管理，并对不合适的文件进行屏蔽删除以保证系统的合法纯洁性。

5、形成知识

用户搜索文件之后可以选择保存此次搜索结果形成相关知识，方便下次查找使用；用户上传文件时，同一分类下的文档自动关联形成知识。

6、推荐系统

文档链接：用户在浏览文档时可发送文档至常用目录方便整个系统所有用户查看，也可发送文档至常用文档以便自己查看。

# **2 关键技术介绍**

本系统的开发环境：

集成开发环境 [IntelliJ IDEA](http://www.baidu.com/link?url=bwHMTA17PLL2qdWWPlZTLTPiPHuO7PwOR8kXn99WTxtWihyfg-NUnaePHJVXUlr3" \t "https://www.baidu.com/_blank)2017

中间件 tomcat 8.0

数据库 mysql 5.7.17

框架支持：springmvc 4.2.5、spring 4.2.5、mybatis 3.4.0、guava 19.0

本系统的实现主要利用的现在流行的web应用程序设计语言——java语言和 Oracle公司旗下数据库产品mysql。Java设计语言是随着internet发展起来的一门语言，它的一个突出特点就是特别适合web应用程序的开发。而myql是现在流行的比较成熟的数据库产品。

## **2.1 java语言**

**2.1.1 历史**

自从1946年世界上第一台电子计算机问世以来，计算模式的发展经历了三个阶段：集中计算模式、分散计算模式和网络计算模式。Internet的出现，为在网络计算模式下的科学计算和信息处理提供了良好平台。然而，要充分利用网络计算模式的功能与效益，平台的运行环境是个制约因素，而java语言和java技术则是解决这一问题的最佳途径。

1991年，SUN公司的Jame Gosling，Bill Joe等人试图为电视、烤箱等家用电器开发一种交互式的软件系统，这种应用需要的是那种小儿可靠的系统，它能够移植，并且实时性好，适用于网络分布环境。流星的C++是自然的选择。不过，开发过程中遇到了许多难以克服的困难使得他们意识到必须彻底解决问题了，最终，开发成功了一个名为Oak的软件，此中蕴藏着Java的原型，当时，并未引起人们的注意。1994年下半年，随着Internet的迅猛发展，环球信息网WWW的快速增长，促进了Java语言研制的进展，使得它逐渐成为Internet上倍受欢迎的开发与编程语言。1995年5月Java正式问世，一些著名的计算机公司纷纷购买了Java语言的使用权，如Microsoft、IBM、Netscape、Novell、Apple、DEC、SGI等。因此，Java语言被美国著名杂志PC Magazine评为1995年十大优秀科技产品之一，随之大量出现了用Java编写的软件产品，受到工业界的重视与好评，认为“Java是八十年代以来计算机界的一件大事”。微软总裁比尔·盖茨在认真观察了一段时间后，确认“Java是长时间依赖计算机界的以来最卓越的程序设计语言”，并确认微软整个软件的开发战略从PC单机时代向以网络为中心的计算时代转移，而购买Java则是他的重大战略决策的实施部署。因此，Java的诞生必将对真个计算机产业发生深远的影响，对传统计算机模型提出了新的挑战。SUN Microsystem 公司的总裁Scott McNealy认为：Java对Internet和WWW开辟了一个崭新的时代。[3]

**2.1.2 特点**

Java语言是一种适用于网络编程的语言，它的基本结构与C++极为相似，但却简单得多。它集成了其他一些语言的特点和优势，又避开了它们的不足之处。它的主要特点如下：

1.简单性

Java和C++相比，剔除了C++的操作符过载、多继承及广泛的自动强制等易混淆和较少使用的特性。Java能够自动处理对象的引用和间接引用，实现自动的无用单元收集，使用户不必为存储管理问题烦恼，能更多的时间和精力花在研发上。

2.面向对象

面向对象的技术是近年来软件开发中用得最为普遍的程序设计方法，它通过把客观事物进行分类组合、参数封装、用成员变量来描述对象的性质、状态，而用方法来实现其行为和功能。面向对象技术具有继承性、封装型、多态性等众多特点，Java在保留这些有点的基础上，又具有动态联编的特性，更能发挥出面向对象的优势。

3.稳健性

Java是一个强类型语言，它允许扩展编译时检查潜在类型不匹配问题的功能。Java要求显式的方法声明，它不支持C风格的[隐式声明](http://baike.baidu.com/item/%E9%9A%90%E5%BC%8F%E5%A3%B0%E6%98%8E" \t "http://baike.baidu.com/_blank)。这些严格的要求保证编译程序能捕捉调用错误，这就导致更可靠的程序。并且Java不支持指针，它消除重写存储和[讹误](http://baike.baidu.com/item/%E8%AE%B9%E8%AF%AF" \t "http://baike.baidu.com/_blank)数据的可能性。类似地，Java自动的“无用单元收集”预防存储漏泄和其它有关动态存储分配和解除分配的有害错误。

4.安全性

Java的存储分配模型是它防御[恶意代码](http://baike.baidu.com/item/%E6%81%B6%E6%84%8F%E4%BB%A3%E7%A0%81" \t "http://baike.baidu.com/_blank)的主要方法之一。Java没有指针，所以程序员不能得到隐蔽起来的内幕和伪造指针去指向[存储器](http://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E5%99%A8" \t "http://baike.baidu.com/_blank)。更重要的是，Java编译程序不处理存储安排决策，所以程序员不能通过查看[声明](http://baike.baidu.com/item/%E5%A3%B0%E6%98%8E/13130358" \t "http://baike.baidu.com/_blank)去猜测类的实际存储安排。编译的Java代码中的存储引用在运行时由Java解释程序决定实际存储地址。

Java运行系统使用字节码验证过程来保证装载到网络上的代码不违背任何Java语言限制。这个安全机制部分包括类如何从网上装载。例如，装载的类是放在分开的名字空间而不是局部类，预防恶意的小应用程序用它自己的版本来代替标准Java类。

5.可移植性

Java使得语言声明不依赖于实现的方面。例如，Java显式说明每个基本数据类型的大小和它的运算行为（这些数据类型由Java语法描述）。

Java环境本身对新的硬件平台和操作系统是可移植的。Java编译程序也用Java编写，而Java运行系统用ANSIC语言编写。

**2.1.3 Java框架结构**

Spring是一个开源框架，Spring是于2003年兴起的一个轻量级的Java开发框架，由Rod Johnson在其著作Expert One-On-One J2EE Development and Design中阐述的部分理念和原型衍生而来。它是为了解决企业应用开发的复杂性而创建的。Spring使用基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅限于服务器端的开发。从简单性、可[测试](http://lib.csdn.net/base/softwaretest" \o "软件测试知识库" \t "http://blog.csdn.net/baidu_31657889/article/details/_blank)性和松耦合的角度而言，任何Java应用都可以从Spring中受益。 简单来说，Spring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架。[4]

Spring MVC属于Spring Framework的后续产品，已经融合在Spring Web Flow里面，它原生支持的Spring特性，让开发变得非常简单规范。Spring MVC 分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制。

MyBatis本是apache的一个开源项目iBatis, 2010年这个项目由apache software foundation 迁移到了google code，并且改名为MyBatis 。MyBatis是一个基于Java的持久层框架。iBATIS提供的持久层框架包括SQL Maps和Data Access Objects（DAO）MyBatis消除了几乎所有的JDBC代码和参数的手工设置以及结果集的检索。MyBatis使用简单的XML或注解用于配置和原始映射，将接口和Java的POJOs（Plain Old [Java](http://lib.csdn.net/base/java" \t "http://blog.csdn.net/baidu_31657889/article/details/_blank" \o "Java 知识库)Objects，普通的 Java对象）映射成[数据库](http://lib.csdn.net/base/mysql" \o "MySQL知识库" \t "http://blog.csdn.net/baidu_31657889/article/details/_blank)中的记录。可以这么理解，MyBatis是一个用来帮你管理数据增删改查的框架。[5]

## **2.2 mysql数据库**

**2.2.1 概述**

MySQL是一个[关系型数据库管理系统](http://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F" \t "http://baike.baidu.com/_blank)，由瑞典MySQL AB 公司开发，目前属于 [Oracle](http://baike.baidu.com/item/Oracle" \t "http://baike.baidu.com/_blank) 旗下产品。MySQL 是最流行的[关系型数据库管理系统](http://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F" \t "http://baike.baidu.com/_blank)之一，在 WEB 应用方面，MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件。

MySQL是一种关系数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

MySQL所使用的 SQL 语言是用于访问[数据库](http://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93" \t "http://baike.baidu.com/_blank)的最常用标准化语言。MySQL 软件采用了双授权政策，分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是[开放源码](http://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%BA%90%E7%A0%81" \t "http://baike.baidu.com/_blank)这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。[6]

**2.2.2 特点**

高速：高速是MySQL的显著特性，在MySQL中，使用了极快的“B树”磁盘表(MyISAM)和索引压缩;通过使用优化的“单扫描多连接”，能够实现极快的连接;SQL函数使用高度优化的类库实现，运行速度快。一直以来，高速都是MySQL吸引众多用户的特性之一，这一点可能只有亲自使用才能体会。

支持多平台：MySQL支持超过20种开发平台，包括Linux、Windows、FreeBSD、IBM AIX、HP-UX、Mac OS、OpenBSD、Solaris等，这使得用户可以选择多种平台实现自己的应用，并且在不同平台上开发的应用系统可以很容易在各种平台之间进行移植。

功能强大：强大的存储引擎使MySQL能够有效应用于任何数据库应用系统，高效完成各种任务，无论是大量数据的高速传输系统，还是每天访问量超过数亿的高强度的搜索Web站点。MySQL 5是MySQL发展历程中的一个里程碑，使MySQL具备了企业级数据库管理系统的特性，提供强大的功能，例如子查询、事务、外键、视图、存储过程、触发器、查询缓存等功能。

安全：灵活和安全的权限和密码系统，允许基于主机的验证。连接到服务器时，所有的密码传输均采用加密形式，从而保证了密码安全。

# **3 系统总体设计**

## **3.1功能分析**

**3.1.1系统功能分析**

知识库管理系统的功能分解图如图3-1所示，主要功能如下：

（1）登录、注册、密码找回：管理员登录，普通用户登录，普通用户注册，普通用户密码找回。

（2）文件上传：支持所有格式的文件上传以及文档新建.

（3）文件检索：可以进行全文检索，或者根据不同条件进行组合检索，并加以保存。

（4）查看文件：查看文件的基本信息，并支持将发送至常用目录、发送至常用文档，并提供文件下载功能，对可预览文件进行预览。

（5）推荐系统：根据用户之间的不同浏览习惯进行分析，提供用户可能感兴趣的内容进行推荐。

（6）后台管理：提供后台管理功能，可以对文件进行审核管理，对用户进行管理，对系统其他设置进行管理等。

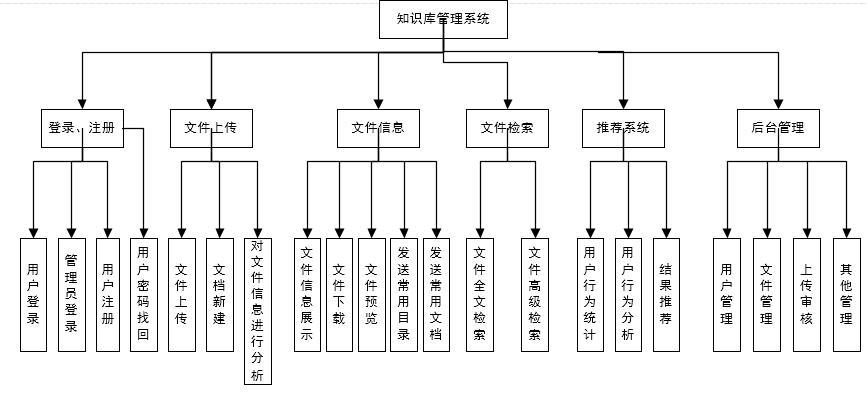


图 3-1系统功能分解图

**3.1.2系统用例图分析**

根据在需求调研与需求分析结果，使用Rational Rose[8]建模工具建立了知识库管理系统用例。系统的主要业务角色分为管理员、普通用户，系统主要功能模块包括删除用户、删除文件、搜索文件、上传文件、下载文件、查看文件详细信息、预览文件、发送至常用文档、发送至常用目录等。系统主要用例图分析如图3-2所示：



图 3-2系统主要用例图

**3.1.3主要用例图的用例规范**

系统主要功能模块包括删除用户、删除文件、搜索文件、上传文件、下载文件、查看文件详细信息、预览文件、发送至常用文档、发送至常用目录等。主要用例分析及用例规约如下：

前置条件：用户已登录，并且角色是管理员

基本事件流：

1. 账户管理模块：

①管理员进入删除用户界面，进行删除用户操作

②管理员进入用户信息页面，进行更改用户密码操作

1. 删除文件模块：

管理员进入指定文件页面，进行删除文件操作

1. 搜索文件模块：

管理员进入搜索文件页面，进行搜索文件操作

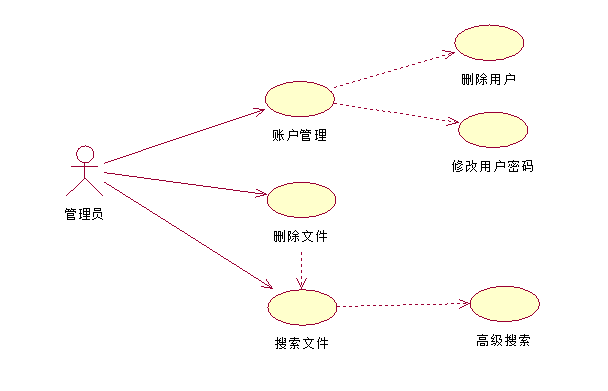


图 3-3系统管理员用例图



图 3-4普通用户用例图

前置条件：用户已登录，并且角色是普通用户

基本事件流：

1. 搜索文件模块：

①用户进入搜索页面，选择全文检索

②用户进入搜索页面，选择高级搜索

1. 上传文件模块：

①用户进入压缩包导入页面，进行上传压缩包操作

②用户进入单文件上传页面，进行单个文件上传操作

③用户进入多文件上传页面，进行多个同一分类文档上传操作

1. 查看文件详细信息模块：

①用户进入文件详细信息页面，点击下载进行下载文件操作

②用户进入文件详细信息页面，点击预览进行预览文件操作

③用户进入文件详细信息页面，点击发送至常用目录以便其他用户查看

④用户进入文件详细信息页面，点击发送至常用文档以便 自己查看

## **3.2系统架构设计**

**3.2.1 系统架构**

系统采取基于B/S结构的软件设计，在这种结构下，用户工作界面是通过WWW浏览器来实现，极少部分业务逻辑在前端实现，但是主要业务逻辑在服务器端实现。大大简化了客户端电脑载荷，减轻了系统维护与升级的成本和工作量，降低了用户的总体成本。

B/S最大的优点就是可以在任何地方进行操作而不用安装任何专门的软件，只要有一台能上网的电脑就能使用，客户端零安装、零维护。B/S架构的软件只需要管理服务器就行了，所有的客户端只是浏览器，根本不需要做任何的维护。无论用户的规模有多大，有多少分支机构都不会增加任何维护升级的工作量，所有的操作只需要针对服务器进行。

系统采用传统的MVC架构模式，将视图、模型、控制器进行分离，每一个模块负责不同模块的功能，降低了系统的耦合度，提高了代码的可重用性，并在完成之后项目的可维护性高。

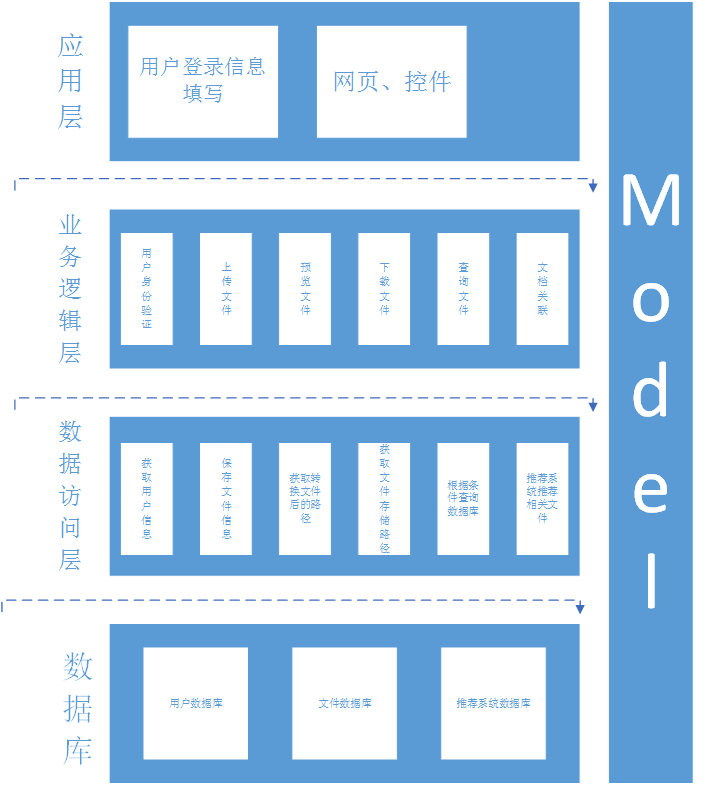


图3-5 总体架构图

**3.2.2 包结构说明**

根据系统结构的设计思想，现在将本系统具体的模块和组建包的放着位置说明如下：

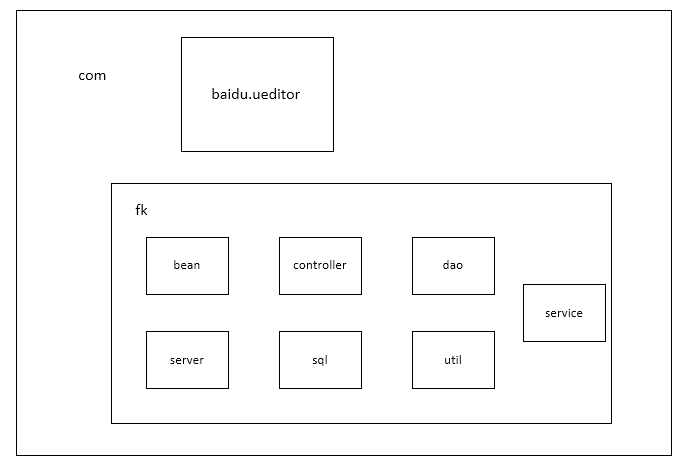


图3-6 系统包结构

说明：

【系统】com——知识库所有的类都放在这个包

【模块】baidu.ueditor——百度富文本编辑器相关支撑类

【系统】fk——系统相关逻辑处理，业务支持等等全部在这个包

【模块】bean——系统所有模型类都在这个包

【模块】controller——控制层的类，负责页面跳转等

【模块】dao——数据库交互层，负责对数据库进行操作，进行数据的操作

【模块】server——系统逻辑处理内容都在这个包

【模块】sql——本系统数据库文件都在这个里

【模块】util——相关支撑功能，例如全文检索、邮件发送系统等，并提供部分工具类和配置文件初始化功能

【模块】service——系统启动时初始化服务

**3.2.3 表现层**

表现层主要通过一些通用的技术把数据呈现给用户，其中主要用到的技术是jsp技术、jstl标准标记库和el表达式。

jstl是一个实现web应用程序中常见的通用功能的定制标记库集，这些功能包括迭代和条件判断、数据管理格式化、XML操作以及数据库访问。通过jstl可以实现服务器端java应用程序常用的基本功能。

jstl定制标记支持一种用于动态属性值的机制。可以用简化的表达式（EL）来指定jstl操作的属性值。EL提供了一些标识符、存取器和运算符，用来检索和操作驻留在JSP容器中的数据。EL擅长寻找对象及其特性，然后对它们执行简单操作，它不是编程语言，甚至不是脚本语言。但是与jstl标记一起使用时，它就能使用简单而又方便的符号来表示复杂的行为。

表现层和业务逻辑层进行数据交互的时候，媒体是form，在memoryfrom中提供了setParamater（String key， Object o），该方法允许想Form中设置对象，在view层可以通过key直接引用这个对象。而在view层中简单数据可以通过getParameter（String: key）来获得。如要取得一个table中的数据，可以把这个table注册成程序中对应的一个类registerProperty（String: key,Class.class）,然后通过getParameter（String：key）获得。

**3.2.4 搜索服务**

系统中会存储大量数据，并且支持对可以抽取的文件进行抽取，在全文检索时对抽取的内容也进行检索。在大量数据的情况下，使用传统的数据库进行检索会发生性能瓶颈。因此希望有一个运行速度快，配置简单并且同时可以免费使用的搜索模式，而且在使用方面可以使用JSON通过HTTP来索引数据，而且搜索服务应该始终可用，并且在数据量不断上涨的情况下，可以很方便的扩展，并且达到实时搜索。

Elasticsearch可以解决上述所有问题以及可能出现的更多其他问题。ES是一个基于Lucene的搜索服务器。它提供了一个分布式多用户能力的全文搜索引擎，基于RESTful web接口。Elasticsearch是用Java开发的，并作为Apache许可条款下的开放源码发布，是当前流行的企业级搜索引擎。设计用于[云计算](http://baike.baidu.com/item/%E4%BA%91%E8%AE%A1%E7%AE%97" \t "http://baike.baidu.com/_blank)中，能够达到实时搜索，稳定，可靠，快速，安装使用方便。[7]

系统采用Elasticsearch作为搜索服务的模块。在管理员上传文件时，将文件的基本信息的所有字段进行存储到数据库和Elasticsearch中，如果该文件是文本文件，那么对文本文件进行文字抽取，将文本文件中的所有文字抽取也存储到Elasticsearch中。这样在进行全文检索请求的时候可以对所有基本信息字段进行查询，也可以在抽取的文字中进行查询，实现快速的实时全文检索的目的。并且在接受多条件的组合检索查询的请求时，可以通过基本信息字段之间的条件组合进行查询，实现多条件的组合查询服务。经过Elasticsearch查询返回的结果只返回对应数据库中的ID，然后服务器对数据库相应ID的内容信息取出返回给用户。

Elasticsearch搜索服务流程图如图3-7所示：

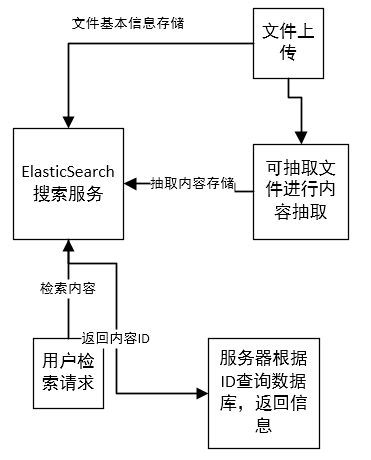


图3-7 搜索服务

系统通过对Elasticsearch增加中文分词ik来对中文内容进行精确的检索。中文分词指的是将一个汉字序列切分成一个一个单独的词。分词就是将连续的字序列按照一定的规范重新组合成词序列的过程。我们知道，在英文的行文中，单词之间是以空格作为自然分界符的，而中文只是字、句和段能通过明显的分界符来简单划界，唯独词没有一个形式上的分界符，虽然英文也同样存在短语的划分问题，不过在词这一层上，中文比之英文要复杂的多、困难的多。因此在使用Elasticsearch作为搜索服务时，内置的分词会对中文检索的精确度产生严重的影响。[IKAnalyzer](http://baike.baidu.com/item/IKAnalyzer" \t "http://baike.baidu.com/_blank)是一个开源的，基于[java语言](http://baike.baidu.com/item/java%E8%AF%AD%E8%A8%80" \t "http://baike.baidu.com/_blank)开发的轻量级的中文分词工具包,使用IK分词可以有效的对中文进行分词，使得Elasticsearch进行有效的索引，从而使得检索结果更加精确。

**3.2.5 推荐系统服务**

系统为了让用户可以快速了解到自己想要了解到的相关知识，借鉴了电子商务系统中的个性化推荐系统。在电子商务系统中个性化推荐是根据用户的兴趣特点和购买行为，向用户推荐用户感兴趣的信息和商品。随着电子商务规模的不断扩大，商品个数和种类快速增长，顾客需要花费大量的时间才能找到自己想买的商品。个性化推荐系统是建立在海量数据挖掘基础上的一种高级商务智能平台，以帮助电子商务网站为其顾客购物提供完全个性化的决策支持和信息服务。

知识库管理系统的个性化推荐系统是为了让用户进行知识浏览和选择提供完全个性化的决策支持和信息服务。是根据用户的信息需求、兴趣等，将用户感兴趣的信息、产品等推荐给用户的个性化信息推荐系统。和搜索引擎相比推荐系统通过研究用户的兴趣偏好，进行个性化计算，由系统发现用户的兴趣点，从而引导用户发现自己的信息需求。现在主要推荐方法有如下六种：基于内容推荐、协同过滤推荐、基于关联规则的推荐、基于效用的推荐、基于知识的推荐、组合推荐。基于内容推荐是建立在项目的内容信息上作出推荐的，而不需要一句用户对项目的评价建议，这样用户的口味必须能够用内容特性形式来表达，不能显示地得到其他用户的判断情况，而知识库管理系统对知识的关联性应该有用户的主观行为进行评判。协同过滤推荐是才用最近邻技术，利用用户的历史喜好信息计算用户之间的距离，然后利用目标用户的最近邻居用户对商品评价的加权评价值来预测目标用户对特定商品的喜好程度，系统根据这一个喜好程度来对目标进行推荐，这种推荐还是比较符合知识库系统的原则。基于关联规则的推荐是针对零售业的推荐系统，不符合需求。基于效用推荐是根据你在用户使用项目的效用情况上计算的，也不符合需求。基于知识推荐是在某种程度是可以看成一种推理，建立在用户需要和偏好基础上，但不是建立在用户需要和偏好基础上推荐的，因此也不符合需求。组合推荐是由于上述方法各有优缺点和应用场景，将几种推荐方法进行组合后避免或拟补各自推荐技术的弱点。考虑到技术的复杂性和相关技术的优缺点和应用场景，知识库管理系统采用协同过滤推荐。系统将用户的行为进行分析，对知识和用户进行相互关联，并且具有相应的评分，为其他用户进行推荐。其基本思想非常易于理解，在日常生活中，我们往往会利用好朋友的推荐来进行一些选择。协同过滤正是把这一思想运用到推荐系统中来，基于其他用户对某一内容的评价来向目标用户进行推荐。[8]协同过滤推荐的基本思想图如3-8：

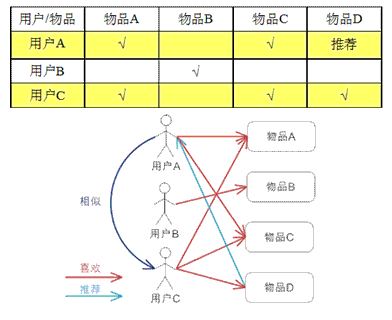


图3-8 协同过滤推荐

知识库管理系统中推荐系统模块的流程图如下：

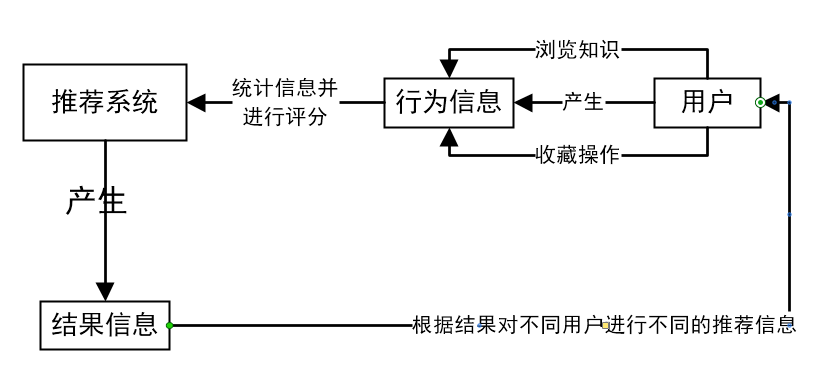


图3-9 推荐系统模块

# 

# **4 数据库设计**

数据库采用了mysql数据库在系统和的编写过程中，主要才用mybatis将关系型数据库里的数据映射成java里的对象（POJO）。这样在编写程序时我们只需要对java对象进行操作。数据持久化的细节则由mybatis进行完成。

## **4.1数据库连接配置**

java通过jdbc连接数据库通讯，mybatis封装了具体的连接细节，可以通过简单的xml配置实现与数据库的互联互通。而系统通过spring将mybatis整合进系统系统，配置文件如下：

<bean id="propertyConfigurer" class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">  
 <property name="location" value="classpath:jdbc.properties"/>  
</bean>  
  
<!-- 配置数据源 -->  
<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource"  
 destroy-method="close">  
 <property name="driverClassName" value="${driverClasss}"/>  
 <property name="url" value="${jdbcUrl}"/>  
 <property name="username" value="${username}"/>  
 <property name="password" value="${password}"/>  
 <!-- 初始化连接大小 -->  
 <property name="initialSize" value="${initialSize}"></property>  
 <!-- 连接池最大数量 -->  
 <property name="maxActive" value="${maxActive}"></property>  
 <!-- 连接池最大空闲 -->  
 <property name="maxIdle" value="${maxIdle}"></property>  
 <!-- 连接池最小空闲 -->  
 <property name="minIdle" value="${minIdle}"></property>  
 <!-- 获取连接最大等待时间 -->  
 <property name="maxWait" value="${maxWait}"></property>  
</bean>  
  
<!-- mybatis和spring完美整合，不需要mybatis的配置映射文件 -->  
<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">  
 <property name="dataSource" ref="dataSource"/>  
 <!-- 自动扫描mapping.xml文件 -->  
 <property name="mapperLocations" value="classpath:mapping/\*.xml"></property>  
</bean>  
  
<!-- DAO接口所在包名，Spring会自动查找其下的类 -->  
<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">  
 <property name="basePackage" value="com.fk.dao"/>  
 <property name="sqlSessionFactoryBeanName" value="sqlSessionFactory"></property>  
</bean>

## **4.2数据库逻辑结构设计和物理结构设计**

经过分析系统的功能需求，将所需要的数据库表进行整理和完善，通过powerdesign进行数据库逻辑结构的设计，如图4-1所示。系统通过分析得出12个数据库表，分别进行知识信息的统计记录、用户信息记录、用户标签统计、用户文档信息记录、知识库常用文档信息记录、用户历史浏览记录、文件类型记录和一些辅助存储的表。[9]

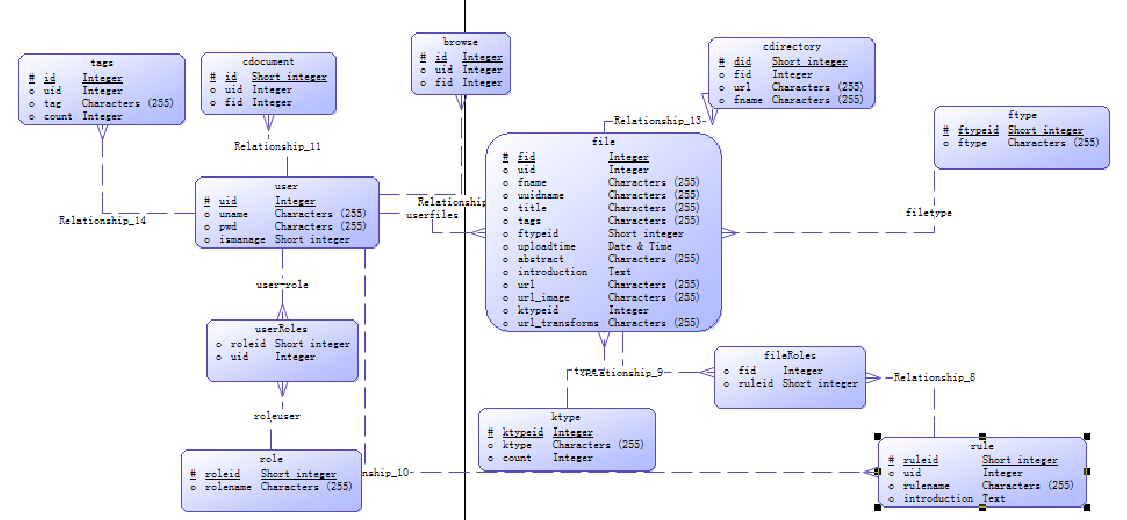


图4-1 数据库逻辑结构设计

通过数据库逻辑设计，得到了数据库的逻辑结构，和具体业务逻辑对应，确定了数据库实体属性、数据类型、长度、精度确定等。对系统后期的完善和调优带来了很大的方便性。

根据设计的逻辑数据库得到相应的物理数据库，物理数据库表如下：

标签存储数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| id | 唯一ID |
| uid | 标签对应的用户 |
| tag | 标签的内容 |
| count | 标签出现的次数 |

常用文档，历史记录数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| id | 唯一id |
| uid | 对应的用户 |
| fid | 对应的文件 |

用户数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| uid | 用户的ID |
| uname | 用户的名称 |
| pwd | 用户的密码 |
| isamage | 是否为管理员 |

文件信息数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| fid | 文件的ID |
| uid | 上传的用户 |
| fname | 文件的名称 |
| uuidname | 文件的随机ID |
| title | 文件的标题 |
| tags | 文件的标签 |
| ftypeid | 文件的类型ID |
| uploadtime | 文件的上传时间 |
| abstract | 文件的摘要 |
| introduction | 文件的描述 |
| url | 文件的存储为主 |
| url\_image | 文件的缩略图存储位置 |
| ktypeid | 文件的分类ID |
| url\_transforms | 文件的预览文件的存储位置 |

分类数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| Ktypeid | 文件的分类ID |
| ktype | 文件分类的名称 |

规则数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| ruleid | 规则的ID |
| uid | 规则对应的用户 |
| rulename | 规则的名称 |
| introduction | 规则的具体描述 |

类型数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| ftypeid | 文件类型的ID |
| ftype | 文件类型的名称 |

常用目录数据库

|  |  |
| --- | --- |
| 名称 | 注释 |
| did | 常用目录文件的唯一ID |
| fid | 对于的文件ID |
| url | 对应文件的存储位置 |
| fname | 文件的名称 |

# **5 模块设计与实现**

## **5.1登录注册**

用户可以通过登录系统来使用系统其他功能，系统可以通过注册功能申请帐号，在注册时需要认真填写邮箱信息，用于忘记密码时候进行密码找回；注册结束之后可以通过注册的帐号密码进行快速登录，注册登录页面如下：



图5-1 用户注册



图5-2 用户登录

如果用户忘记密码可以根据以下流程来进行密码找回：首先填写用户邮箱，然后系统会给用户邮箱中发送确认链接，用户点击链接则会重置密码，系统会将重置之后的密码发送到用户的邮箱。具体流程图如下：



图5-3 邮箱密码找回



图5-3 找回密码链接



图5-4 密码重置邮箱

## **5.2 系统主界面**

系统主界面提供了全文检索和常用目录的功能，不管登录还是未登录用户都可以通过系统来查询自己想要的内容，并且可以看到大家添加到常用目录的内容，常用目录是用户推荐到首页的内容，可由管理员进行管理；首页还提供了推荐系统的展示内容，如果用户以未登录，则随机为用户进行展示，如果用户登录则按照推荐分析的算法结果进行推荐。

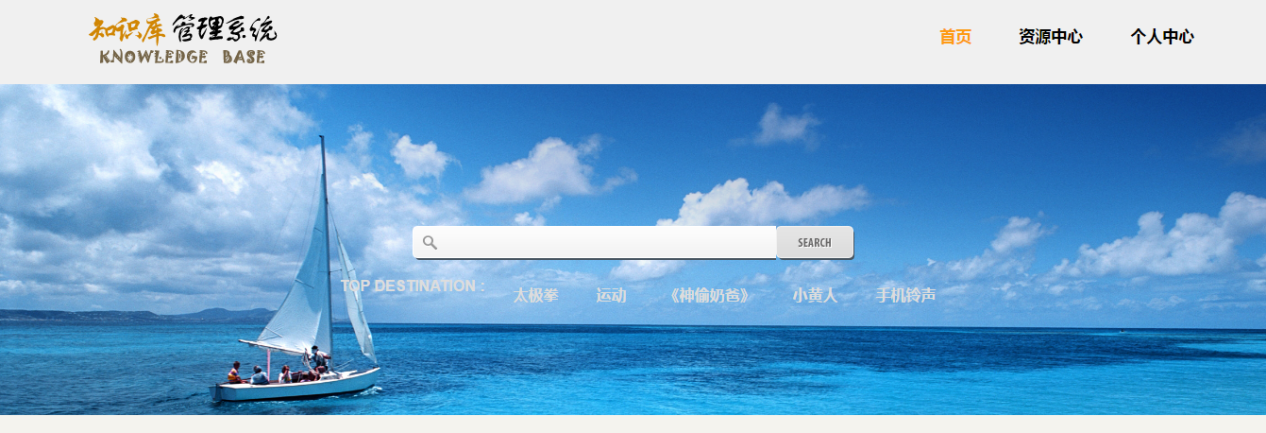


图5-5 系统主界面全文检索



图5-6 常用目录



图5-7 推荐系统展示内容

## **5.3 资源内容**

系统资源界面为用户展示所有的资源内容，并提供全文检索和多条件检索的功能，全文检索和首页提供的功能相同，多条件检索功能可以根据用户的选择自行添加不同的条件，并且为这些条件指定与或关系，进行更加精确的检索功能，同时系统的检索速度并没有下降。

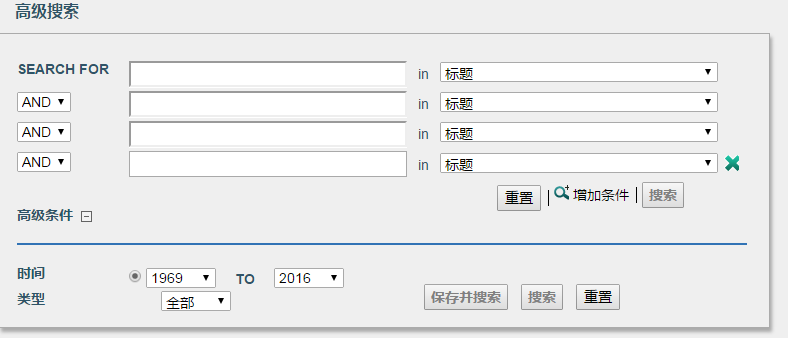


图5-8 高级检索内容



图5-9 文件信息列表页面

用户可以点击文件信息，或者点击查询结果的文件信息，进入文件信息界面，在文件信息界面将会展示文件的具体录入时候的填写的详细信息，并加入了很方便的链接可以直接找到相关内容。已登录的用户可以讲文件发送至常用目录让其他用户也可以看到也可以讲文件发送到常用文档进行收藏，以便自己之后进行查看；所有访问用户均可下载有附件的文件附件，附件可以是任何格式的文件；如果文件可以进行预览的话，例如office系列文件，系统可以讲word、excel、ppt、pdf等文件进行转换为swf格式文件在系统中存储，然后方便用户进行预览，对于视频文件系统会讲视频文件转换为统一格式，方便用户进行预览播放，对于音频文件可以进行直接播放等预览功能。



图5-10 文件信息界面

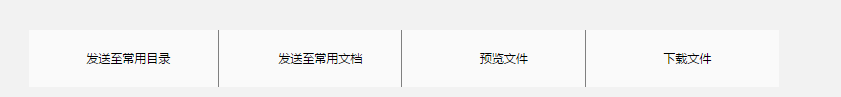


图5-11 文件操作内容界面

## **5.3 个人信息**

用户可以进入个人中心页面查看自己的相关信息，用户个人信息界面可以上传自己想要上传的知识文件，用户非管理员上传的文件需要经过管理员审核通过才能发布到系统中，否则只能自己进行查看；个人中心页面展示了用户历史浏览的记录、用户常用文档、用户上传的文件和用户在审核的文件。



图5-12 用户个人界面

系统根据用户的行为信息进行对用户的统计，整理出相应的标签云进行展示。标签云是一套相关的标签以及与此相应的[权重](http://baike.baidu.com/item/%E6%9D%83%E9%87%8D" \t "http://baike.baidu.com/_blank)。典型的标签云有30至150个标签。权重影响使用的字体大小或其他视觉效果。同时，直方图或饼图表是最常用的代表约12种不同的权数。因此，标签[云彩](http://baike.baidu.com/item/%E4%BA%91%E5%BD%A9" \t "http://baike.baidu.com/_blank)能代表更多的权，尽管不那么准确。此外，标签云通常是可以交互的：标签是典型的超链接，让用户可以仔细了解他们的内容。



图5-13 用户标签云

## **5.4 后台管理**

系统提供了后台管理界面对知识、分类、文件类型和用户进行添加的模块。还提供了对知识、分类、文件类型、用户、常用文档、用户标签和用户上传内容审核的模块。系统分为普通管理员和超级管理员两种，普通管理员由超级管理员创建，普通管理员之间不可管理各自信息。

管理员访问后台网址时需要先验证身份进行登录，登录界面如下：

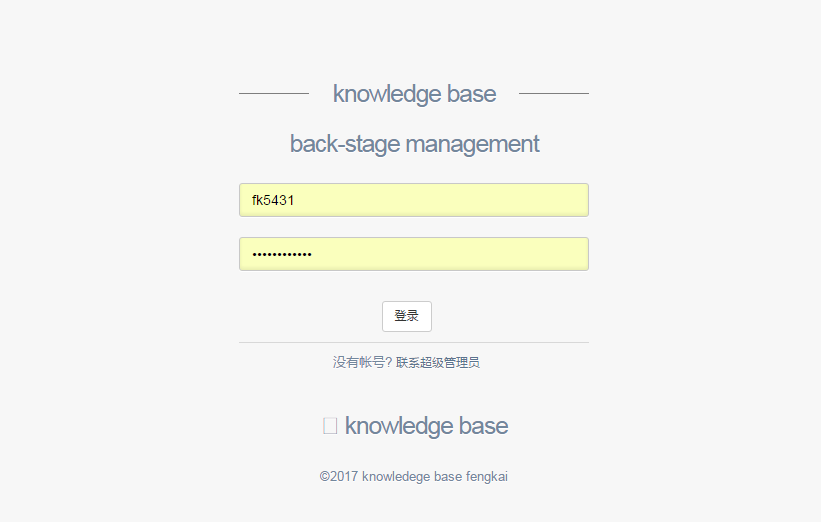


图5-14 管理员登录

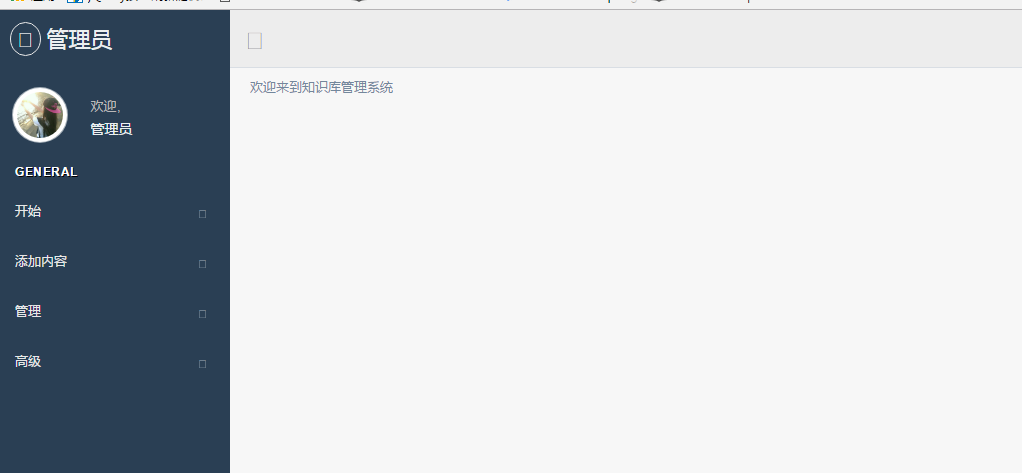


图5-15 后台界面

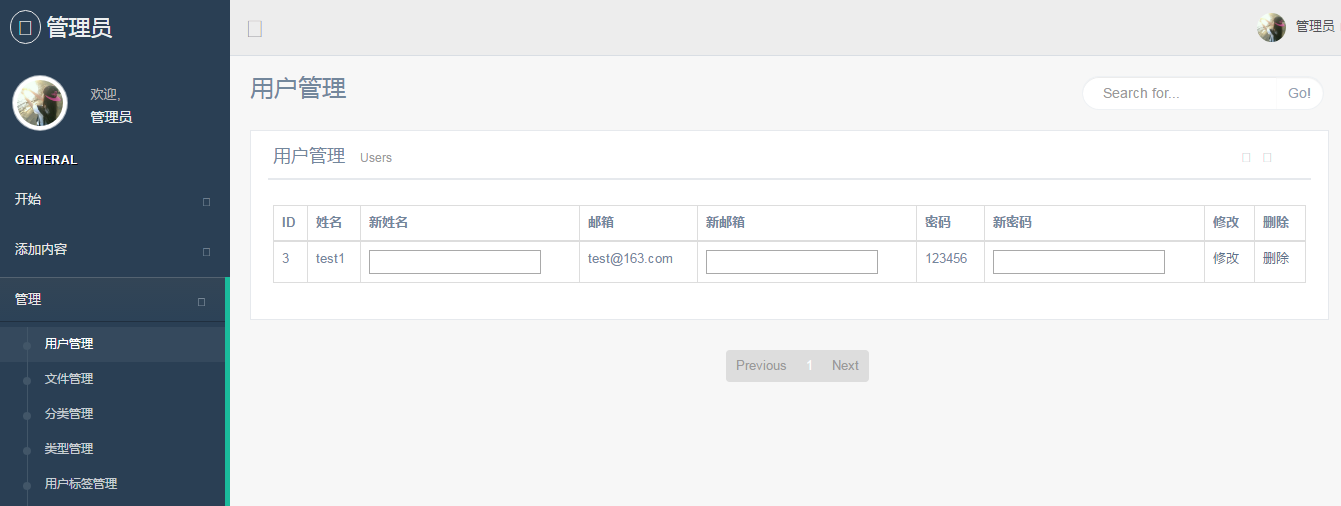


图5-16 用户管理界面

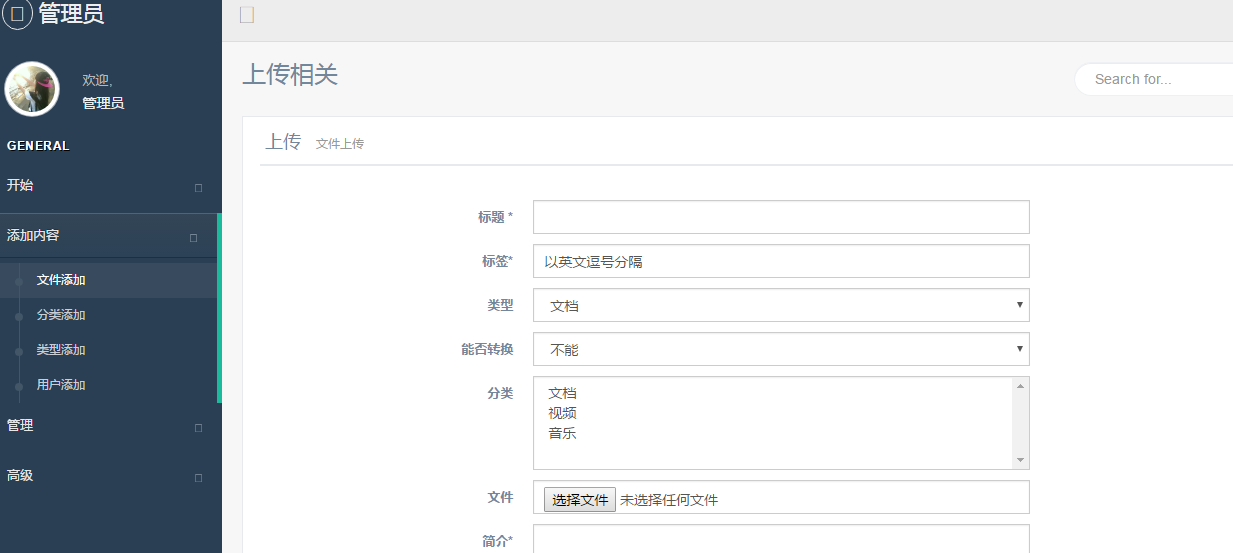


图5-17 管理员上传内容



图5-18 用户上传审核

# **6 系统测试**

## **6.1 测试意义和目标**

系统测试的目的是在真实系统工作环境下通过与系统的需求定义作比较，检验完整的软件配置项能否和系统正确连接，发现软件与系统/子系统设计文档和软件开发合同规定不符合或与之矛盾的地方。系统测试是将通过确认测试的软件，作为整个基于计算机系统的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素结合起来，在实际运行（使用）环境下，对计算机系统进行的测试。是为了发现缺陷并度量产品质量，按照系统的功能和性能需求进行的测试。而且，系统测试还要检验系统的文档等是否完整、有效。另外，系统测试的测试用例应根据需求分析说明书来设计，并在实际使用环境下来运行。

软件测试技术归结为两大类：白盒测试和黑盒测试。白盒测试有可分为静态测试和动态测试；静态测试主要包括代码检查法、静态结构分析法等；动态测试技术主要包括程序插桩、逻辑覆盖、基本路径测试等。黑河测试一般可分为功能测试和非功能测试两大类：功能测试主要包括等价类划分、边值分析、因果图法、错误推测、强度测试、兼容性测试、配置测试、安全测试等。本系统测试主要采用黑盒测试。[10]

对开发的系统网站做测试主要是为了检测设计的功能是否能达到设计时的需求目标，运行过程是否会出现错误的跳转，系统是否能在浏览器中正常显示运行等，通过一系列测试发现问题、解决问题，从而完善系统。网站测试对于网站开发的成败有着至关重要的作用，其内容也比普通的软件测试更具复杂性。在此，我设计了该网站主要功能的测试用例。

## **6.2 测试用例**

**6.2.1 用户登录测试**

表6-1 用户登陆模块测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 前提条件 | 输入 | 预期结果 | 实际结果 |
| 1-1 | 用户已在该网站成功注册了账号密码 | 输入正确的账户和密码，点击登录按钮 | 用户成功登录，进入网站首页 | 页面跳转到网站首页，可进入个人信息页面 |
| 1-2 | 输入错误的密码，点击登录按钮 | 提示密码错误，不能成功登录 | 提示“密码错误，请您重新登录！” |
| 1-3 | 输入错误的账号，点击登录按钮 | 提示账号错误，不能成功登录 | 提示“该用户不存在，请您重新登录!” |
| 1-4 | 在后台管理页面修改了用户登录密码，登录时仍输入修改前的密码 | 提示密码错误，不能成功登录 | 提示“密码错误，请您重新登录！” |
| 1-5 | 在后台管理页面修改了用户登录密码，登录时输入修改后的密码 | 登录成功 | 登录成功，页面跳转至网站首页 |
| 1-6 | 在后台删除了某个用户信息，登录时仍使用被删除的账号登录 | 提示错误，不能成功登录 | 提示“该用户不存在，请您重新登录！” |

## 

**6.2.2 搜索功能测试**

表6-2 搜索功能测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 输入 | 预期结果 | 实际结果 |
| 2-1 | 在搜索框中输入后台信息中存在的某一知识中的相关字段，点搜索按钮 | 搜索出字段相应的知识信息 | 页面跳转至搜索字段相关的知识信息 |
| 2-2 | 在搜索框中输入后台信息中没有的知识相关字段名 | 无搜索结果 | 显示无结果 |

**6.2.3 管理员管理功能测试**

表6-3 管理员管理功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 前提条件 | 输入 | 预期结果 | 实际结果 |
| 3-1 | 管理员已登录 | 管理员进行文件上传，缺少任何一个字段值 | 提示错误 | 系统提示缺少相应字段，请完善重试 |
| 3-2 | 管理员进行文件上传，完整的填写信息 | 上传成功 | 系统文件上传成功，返回上传页面，进入管理页面可显示刚刚上传的文件 |
| 3-3 | 管理员进行用户，文件的删除、信息查看的等管理功能 | 删除成功，信息显示正确 | 管理员成功删除，在数据库中显示不到信息，查看信息与数据库中的一致 |
| 3-4 | 管理员进行用户上传的文件审核，进行查看、通过、不通过等操作 | 可以正确查看信息，不通过时删除审核记录，通过时，文件可添加到系统中显示 | 查看信息正确，不通过时删除审核记录，通过时，文件在系统可查看到审核通过的文件 |

# 

# **7 总结**

在本课题里主要讨论用j2ee平台来设计和实现一个基于B/S模式的web应用程序，整个系统的实现，经历了从做用户需求分析、系统架构设计、系统的模块设计，到最后用代码实现一个比较完整的知识库系统的整个过程。整个系统分析工作存在很多经验教训，可以总结如下：

首先，对于开发一个软件系统，做一份详细的用户需求分析是非常必要的。通过这样的一份分析说明，我们会对用户所需要的系统有一个比较深刻的了解，知道用户具体需要用这个系统来做什么、需要一个什么样的功能来减轻他们的体力劳动从而提高工作效率。同时，通过这个分析，我们也能为自己在做软件开发时确定一个比较具体的计划，包括确定一个大概的开发周期，在系统中具体要求实现什么功能，系统最后要达到什么样的性能指标。在了解这些具体信息后，我们才能进行下一步开发工作，否则会对以后的工作造成极大的不便。

其次，就是在整个系统设计的初期要有一个完善而且正确的开发流程，系统的开发要按照流程来进行。像这个系统是一个web应用系统，针对这个系统的开发流程应该是这样的：第一步，确定真个系统需要实现的功能，并且在开发能力里能够实现的；第二步，由开发人员设计系统的整体架构，架构设计的好坏直接影响到系统的性能。第三步，分模块实现系统的所有功能，在这个阶段，系统界面的的设计，模块流程的设计以及代码的编写。

最后一步，就是对整个系统进行相应的性能测试，测试不仅仅是找出系统的错误，还可以验证系统设计的优劣。积累设计经验。优秀的架构设计加上合适的开发方法可以保证系统在完成后较少的bug。

最后系统设计完成后是要用户使用的，除了完成系统必须的基础功能，我们还应该站在用户的角度考虑一下，花一些时间来提高用户使用的舒适程度，即用户的体验度。好的用户体验可以叫用户工作起来不感觉到累，这样系统的评价也会高。

# **致谢**

本论文感谢山东科技大学四年来对我的辛苦培育，让我在大学这四年来学到很东西，特别感谢计算机科学与工程学院为我提供了良好的学习环境、感谢领导、老师们四年来对我无微不至的关怀和指导，让我得以在这四年中学到很多有用的知识。在此，我还要感谢在班里同学和朋友，感谢你们在我遇到困难的时候帮助我，给我支持和鼓励，感谢你们。

特别感谢我的指导老师刘连山，在本系统开发中给予我悉心指导，从系统开发到结束中过程遇到很多困难都是他给我鼓励与指引，使我能够克服重重困难，将系统做完成，在此谨向刘老师致以诚挚的谢意和崇高的敬意。谢谢!

# **参考文献**

[1] 李勇、陈钰、曹国华.企业知识管理系统的结构探讨.重庆：大学学报（自然科学版），2002（11）：107-109

[2] Brent Gallupe. Knowledge management systems：surveying the landscape[J]International Journal of Management Review.2001,（3）

[3] 张桂珠 [刘丽](https://www.baidu.com/s?wd=%E5%88%98%E4%B8%BD&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YkPvc3PADkujD4P1fdPjPB0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHTknW63njf3" \t "https://zhidao.baidu.com/question/_blank) 陈爱国 Java面向对象程序设计（第2版）[北京邮电大学出版社](https://www.baidu.com/s?wd=%E5%8C%97%E4%BA%AC%E9%82%AE%E7%94%B5%E5%A4%A7%E5%AD%A6%E5%87%BA%E7%89%88%E7%A4%BE&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YkPvc3PADkujD4P1fdPjPB0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHTknW63njf3" \t "https://zhidao.baidu.com/question/_blank)

[4] 林信良. Spring2.0技术手册[M]. 北京：电子工业出版社，2005

[5] 魏静敏;刘欢杰. [基于Mybatis框架的批量数据插入的性能问题的探讨](http://www.cnki.com.cn/Article/CJFDTOTAL-GPRJ201319091.htm" \t "http://www.cnki.com.cn/Article/_blank)[J] 计算机光盘软件与应用;2013年19期

[6] 王志刚.MySQL高效编程.人民邮电出版社.2012

[7] 郎小伟，王申康. 基于lucene的全文检索系统的研究与开发.计算机工程.2006.2.

[8] Lin Hongfei,et al Text Browsing Based on Latent Semantic Indexing[J]. Joural of Chinese Information Processing, 2000, 14(5):241-245.

[9] 姜承尧.高性能网站MySQL数据库实践.维普中文科技期刊数据库.2013

[10] (美)霍普 等著，傅鑫 等译 Web 安全测试 清华大学出版社 2011.03.

# **附录**：

**Neural Collaborative Filtering**

**ABSTRACT**

In recent years, deep neural networks have yielded immense success on speech recognition, computer vision and natural language processing. However, the exploration of deep neural networks on recommender systems has received relatively less scrutiny. In this work, we strive to develop techniques based on neural networks to tackle the key problem in recommendation — collaborative filtering — on the basis of implicit feedback.

Although some recent work has employed deep learning for recommendation, they primarily used it to model auxiliary information, such as textual descriptions of items and acoustic features of musics. When it comes to model the key factor in collaborative filtering — the interaction between user and item features, they still resorted to matrix factorization and applied an inner product on the latent features of users and items.

By replacing the inner product with a neural architecture that can learn an arbitrary function from data, we present a general framework named NCF, short for Neural networkbased Collaborative Filtering. NCF is generic and can express and generalize matrix factorization under its framework. To supercharge NCF modelling with non-linearities, we propose to leverage a multi-layer perceptron to learn the user–item interaction function. Extensive experiments on two real-world datasets show significant improvements of our proposed NCF framework over the state-of-the-art methods. Empirical evidence shows that using deeper layers of neural networks offers better recommendation performance. **Keywords**

Collaborative Filtering, Neural Networks, Deep Learning, Matrix Factorization, Implicit Feedback

**INTRODUCTION**

In the era of information explosion, recommender systems play a pivotal role in alleviating information overload, having been widely adopted by many online services, including E-commerce, online news and social media sites. The key to a personalized recommender system is in modelling users’ preference on items based on their past interactions (e.g., ratings and clicks), known as collaborative filtering [31, 46]. Among the various collaborative filtering techniques, matrix factorization (MF) [14, 21] is the most popular one, which projects users and items into a shared latent space, using a vector of latent features to represent a user or an item. Thereafter a user’s interaction on an item is modelled as the inner product of their latent vectors.

Popularized by the Netflix Prize, MF has become the de facto approach to latent factor model-based recommendation. Much research effort has been devoted to enhancing MF, such as integrating it with neighbor-based models [21], combining it with topic models of item content [38], and extending it to factorization machines [26] for a generic modelling of features. Despite the effectiveness of MF for collaborative filtering, it is well-known that its performance can be hindered by the simple choice of the interaction function — inner product. For example, for the task of rating prediction on explicit feedback, it is well known that the performance of the MF model can be improved by incorporating user and item bias terms into the interaction function1 . While it seems to be just a trivial tweak for the inner product operator [14], it points to the positive effect of designing a better, dedicated interaction function for modelling the latent feature interactions between users and items. The inner product, which simply combines the multiplication of latent features linearly, may not be sufficient to capture the complex structure of user interaction data.

This paper explores the use of deep neural networks for learning the interaction function from data, rather than a handcraft that has been done by many previous work [18, 21]. The neural network has been proven to be capable of approximating any continuous function [17], and more recently deep neural networks (DNNs) have been found to be effective in several domains, ranging from computer vision, speech recognition, to text processing [5, 10, 15, 47]. However, there is relatively little work on employing DNNs for recommendation in contrast to the vast amount of literatureon MF methods. Although some recent advances [37, 38, 45] have applied DNNs to recommendation tasks and shown promising results, they mostly used DNNs to model auxiliary information, such as textual description of items, audio features of musics, and visual content of images. With regards to modelling the key collaborative filtering effect, they still resorted to MF, combining user and item latent features using an inner product.

This work addresses the aforementioned research problems by formalizing a neural network modelling approach for collaborative filtering. We focus on implicit feedback, which indirectly reflects users’ preference through behaviours like watching videos, purchasing products and clicking items. Compared to explicit feedback (i.e., ratings and reviews), implicit feedback can be tracked automatically and is thus much easier to collect for content providers. However, it is more challenging to utilize, since user satisfaction is not observed and there is a natural scarcity of negative feedback. In this paper, we explore the central theme of how to utilize DNNs to model noisy implicit feedback signals. The main contributions of this work are as follows.

1. We present a neural network architecture to model latent features of users and items and devise a general framework NCF for collaborative filtering based on neural networks.

1. We show that MF can be interpreted as a specialization of NCF and utilize a multi-layer perceptron to endow NCF modelling with a high level of non-linearities.
2. We perform extensive experiments on two real-world datasets to demonstrate the effectiveness of our NCF approaches and the promise of deep learning for collaborative filtering.

**神经协同过滤**

**摘要**

　　近年来，深层神经网络在语音识别，计算机视觉和自然语言处理方面都取得了巨大的成功。然而相对的，对应用深层神经网络的推荐系统的探索却受到较少的关注。在这项工作中，我们力求开发一种基于神经网络的技术，来解决在含有隐形反馈的基础上进行推荐的关键问题————协同过滤。

　　尽管最近的一些工作已经把深度学习运用到了推荐中，但是他们主要是用它（深度学习）来对一些辅助信息（auxiliary information）建模，比如描述文字的项目和音乐的声学特征。当涉及到建模协同过滤的关键因素（key factor）————用户和项目（item）特征之间的交互的时候，他们仍然采用矩阵分解的方式，并将内积（inner product）做为用户和项目的潜在特征点乘。通过用神经结构代替内积这可以从数据中学习任意函数，据此我们提出一种通用框架，我们称它为NCF（Neural network-based Collaborative Filtering，基于神经网络的协同过滤）。NCF是一种通用的框架，它可以表达和推广矩阵分解。为了提升NFC的非线性建模能力，我们提出了使用多层感知机去学习用户-项目之间交互函数（interaction function）。在两个真实世界（real-world）数据集的广泛实验显示了我们提出的NCF框架对最先进的方法的显著改进。

**关键字**

　　协同过滤,神经网络,深度学习,矩阵分解,隐性反馈（Implicit Feedback）.

**1. 引言**

　　在信息爆炸的时代，推荐系统在减轻信息过载方面发挥了巨大的作用，被众多在线服务，包括电子商务，网络新闻和社交媒体等广泛采用。个性化推荐系统的关键在于根据过去用户交互的内容（e.g. 评分，点击），对用户对项目的偏好进行建模，就是所谓的协同过滤**[31,46]**。在众多的协同过滤技术中，矩阵分解（MF）**[14,21]**是最受欢迎的，它将用户和项目映射到共享潜在空间（shared latent space），使用潜在特征向量（latent features），用以表示用户或项目。这样一来，用户在项目上的交互就被建模为它们潜在向量之间的内积。

　　由于Netflix Prize的普及，MF已经成为了潜在因素（latent factor）建模推荐的默认方法。已经有大量的工作致力于增强MF（的性能），例如将其与基于邻居（相邻用户or项目）的模型集成**[21]**，与项目内容的主题模型**[38]**相结合，还有将其扩展到因式分解机（factorization machines）**[26]**，以实现特征的通用建模。尽管MF对协同过滤有效，但众所周知，其性能可以被简单选择交互函数（内积）所阻碍。例如，对于显式反馈的评估预测任务，可以通过将用户和项目偏移项纳入交互函数来改善MF模型的性能**1**。虽然内积算子**[14]**似乎只是一个微不足道的调整，但它指出了设计一个更好的专用交互函数，用于建模用户和项目之间的潜在特征交互的积极效果。简单地将潜在特征的乘积线性组合的内积可能不足以捕捉用户交互数据的复杂结构。

本文探讨了使用深层神经网络来学习数据的交互函数，而不是那些以前已经完成的手动工作（handcraft）[18,21]。神经网络已经被证明有拟合任何连续函数的能力[17]，最近深层神经网络（DNN）被发掘出在几个领域的出色表现：从计算机视觉，语音识别到文本处理[5,10,15,47]。然而，与广泛的MF方法文献相比，使用DNNs进行推荐的工作相对较少。虽然最近的一些研究进展[37,38,45]已经将DNN应用于推荐任务，并展示出不错的效果，但他们大多使用DNN来建模一些辅助信息，例如物品的文字描述，音乐的音频特征和图像的视觉内容（描述）。对于影响建模效果的关键因素——协同过滤，他们仍然采用MF，使用内积结合用户和项目潜在特征。

　　我们这项工作是通过形式化用于协同过滤的神经网络建模方法来解决上述研究问题。我们专注于隐性反馈（implicit feedback），通过参考观看视频，购买产品和点击项目等间接反映用户偏好的行为。与显性反馈（explicit feedback）（i.e. 评级和评论）相比，隐性反馈可以自动跟踪，从而更容易为内容提供商所收集。但是，由于隐性反馈不能反映用户的满意度，并且负反馈（negative feedback）存在自然稀疏（natural scarcity）问题，使得这个问题更具挑战性。在本文中，我们探讨了如何利用DNN来模拟噪声隐性反馈信号的中心问题。

　　这项工作的主要贡献有如下几点：

　　1、我们提出了一种神经网络结构来模拟用户和项目的潜在特征，并设计了基于神经网络的协同过滤的通用框架NCF。

　　2、我们表明MF可以被解释为NCF的特例，并利用多层感知器来赋予NCF高水平的非线性建模能力。

　　3、我们对两个现实世界的数据集进行广泛的实验，以证明我们的NCF方法的有效性和对使用深度学习进行协作过滤的承诺。