详细设计说明书

V1.1

1 引言

1.1 编写目的

本设计说明书的编写目的在于研究基于协同过滤的在线教育平台的开发途径和应用方法，主要是为了对该在线教育平台进行使用和维护。

本项目任务提出者为方涛，主要开发人员有方涛、胡恒昌、姜美羡。我们的目标用户主要为在校本科大学生，教师等人员；项目运行条件包括一台腾讯云服务器。

本设计说明书详细划分了功能模块以及规定了软件主要功能模块之间的交互关系，并且明确了模块内部功能需求划分、输入输出格式、处理流程、算法实现、限制条件、接口规定等具体要求。

1.2 背景

本项目全称为基于协同过滤的在线教育平台，是使用协同过滤算法实现的教育系统。目标用户在于在校大学生群体，提供互联网的在线教育信息支持，同时也满足自学者群体的需求。

随着网络技术的进步，互联网的普及，网络阅读成本降低。在校大学生能够熟练使用计算机，是网络阅读的主要群体。调查显示，在线阅读、手机阅读、手持式阅读器阅读等数字阅读方式开始普及。网络在线阅读排第一位，手机阅读排第二位。数字媒介阅读代替书面阅读。廉价的电子书成为他们主要选择的阅读对象。惯以纸质为载体的报纸、杂志和图书所占比例较小。深受大学生依赖的网络阅读，呈现一种时效性、阶段性、冗杂性的本质，从阅读的形式上分析，这种网络阅读就是一种浅阅读。浅阅读已经成为一种流行，作为网络时代新的阅读方式，浅阅读除了与传统阅读一样获取信息外，更注重追求阅读过程中的视觉快感和心理愉悦，而难以获得实质上的深刻学习。而我们的项目希望将用户引向网络深阅读。

数据显示，2014 年中国职业在线教育市场规模为 257 亿元，同比增长 18.1%，预计之后几年将保持 20%左右的速度增长，到 2018 年达 515.5 亿元。我们分析认为，职业在线教育和用户的职业生涯息息相关，用户的付费意愿强，在线教育的形式能够 更好的被该类人

群接受。用户在选择机构时，更关注品牌知名度和机构的后续服务能力，如工作推荐等。

在线教育指的是通过应用信息科技和互联网技术进行内容传播和快速学习的方法。与传统教育机构的教育方式相比，在线教育具有效率高、方便（打破了时空限制，可碎片化学习）、低门槛、教学资源丰富的特点。基于上述特点，再加上“互联网+”推动，在线教育平台兴起，市场需求也与日俱增。

1.3 定义

（1）在线教育：

或称远程教育、在线学习，现行概念中一般指的是指一种基于网络的学习行为，与网络培训概念相似。

（2）文本分类：

文本分类用电脑对文本集(或其他实体或物件)按照一定的分类体系或标准进行自动分类标记。

（3）网络爬虫（Reptilia）：

是一种自动获取网页内容的程序。是搜索引擎的重要组成部分，因此搜索引擎优化很大程度上就是针对爬虫而做出的优化。

（4）协同过滤推荐（Collaborative Filtering recommendation）：

协同过滤分析用户兴趣，在用户群中找到指定用户的相似（兴趣）用户，综合这些相似用户对某一信息的评价，形成系统对该指定用户对此信息的喜好程度预测。

（5）朴素贝叶斯算法（Naive Bayesian Model）：

朴素贝叶斯法是基于贝叶斯定理与特征条件独立假设的分类方法。

1.4 参考资料

[1]刘露,彭涛,左万利,戴耀康. 一种基于聚类的 PU 主动文本分类方法[J]. 软件学报,2013,11:2571-2583.

[2]平源. 基于支持向量机的聚类及文本分类研究[D].北京邮电大学,2012.

[3]杨杰明. 文本分类中文本表示模型和特征选择算法研究[D].吉林大学,2013.

[4]李荣陆. 文本分类及其相关技术研究[D].复旦大学,2005.

[5]王煜. 基于决策树和K最近邻算法的文本分类研究[D].天津大学,2006.

[6]苏金树,张博锋,徐昕. 基于机器学习的文本分类技术研究进展[J]. 软件学报,2006,09:1848-1859.

[7]周平红. 我国高等教育信息化水平测评与发展预测研究[D].华中师范大学，2012.

[8]范坤. 推进我国教育信息化建设进程的对策研究[D].华中师范大学，2004.

[9]牛龙飞. 基于Web的我国教育信息化公共服务平台的设计与实现[D].华中师范大学，2013.

[10] 艾瑞咨询2015中国在线教育行业发展报告

2 程序系统的结构

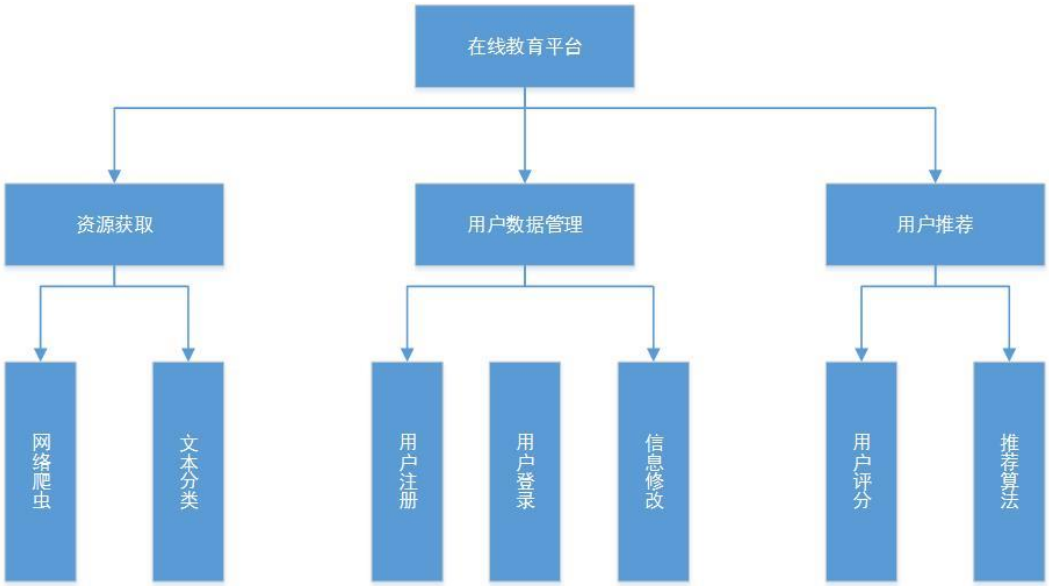

图 1

系统分为三大板块，分别是资源获取、用户数据管理和用户推荐。网络爬虫爬取教育资源，文本分类为资源打上标签，存储于数据库。这是资源获取阶段的任务。用户数据管理涵盖三个方面：用户注册、用户登录和信息修改。用户推荐板块根据用户对书籍的评分，运用推荐算法实现个性推荐。

3 用户注册模块设计说明

3.1 程序描述

本程序模块主要应用于允许用户在本系统中注册信息，从而进一步查看和使用本系统中的服务。本模块与数据库交互，允许用户提出请求并接收用户发送的用户名、密码、个人邮箱等个人信息并校验数据格式符合后存入数据库，已注册的用户将被允许正常登录本系统。本程序模块常驻内存，并且用户邮箱信息不可重复，同时能够并发处理多个用户同时提出的注册请求。

3.2 功能

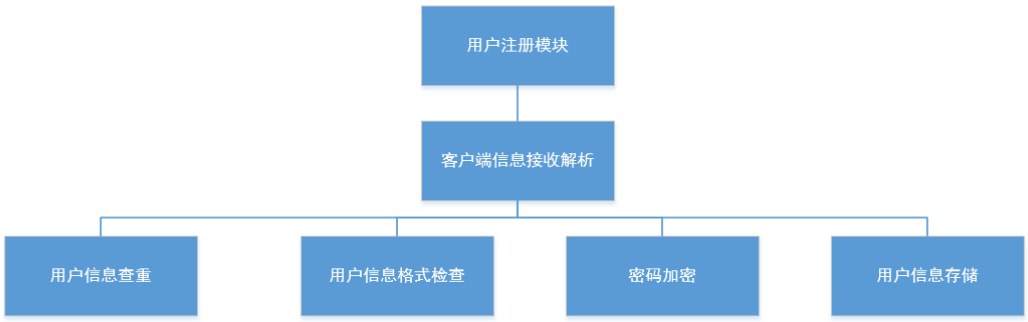

图 2

用户注册模块也就是对客户端信息接收解析，包括用户信息查重、用户信息格式检查、密码加密和用户信息存储。

3.3 性能

系统接收客户端用户注册请求的延迟时间不超过 2.0 秒；

系统处理用户注册请求、写入数据库并且返回注册成功信息至客户端时间上限 2.0 秒。

3.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密

用户名	String	1~20 字符	N	‘’	InputBox	否
用户密码	String	8~20 字符	N	‘’	InputBox	是
用户邮箱	String	10~20 字符	N	‘’	InputBox	否
用户性别	Bool	True/False	N	True	radio button	否

3.5 输出项

输出名称	数据类型	有效范围	是否加密
Check	Bool	True/False	否

3.6 算法

本模块运行时，客户端提示用户输入用户名、用户密码、用户邮箱等信息，并通过 JavaScript 使用正则表达式实现对用户邮箱输入合法性的检查。用户点击提交后，客户端发送 POST 请求并由服务端接收，服务端在检查用户邮箱不重复后写入数据库，并向客户端返回注册成功提示；如果重复信息则返回注册失败。

本功能模块主要涉及用户密码加密算法。

本功能采用 MD5 算法对密码进行加密。MD5 为计算机安全领域广泛使用的一种散列函数，用以提供消息的完整性保护。MD5 的全称是 Message-Digest Algorithm 5（信息-摘要算法），MD5 用于确保信息传输完整一致。是计算机广泛使用的杂凑算法之一（又译摘要算法、哈希算法），主流编程语言普遍已有 MD5 实现。将数据（如汉字）运算为另一固定长度值，是杂凑算法的基础原理，MD5 的前身有 MD2、MD3 和 MD4。

MD5 的作用是让大容量信息在用数字签名软件签署私人密钥前被"压缩"成一种保密的格式（就是把一个任意长度的字节串变换成一定长的十六进制数字串）。

MD5 以 512 位分组来处理输入的信息，且每一分组又被划分为 16 个 32 位子分组，经过了一系列的处理后，算法的输出由四个 32 位分组组成，将这四个 32 位分组级联后将生成一个 128 位散列值。

在 MD5 算法中，首先需要对信息进行填充，使其位长对 512 求余的结果等于 448。因此，信息的位长（Bits Length）将被扩展至 N*512+448，N 为一个非负整数，N 可以是零。

填充的方法如下，在信息的后面填充一个 1 和无数个 0，直到满足上面的条件时才停止用 0 对信息的填充。然后，在这个结果后面附加一个以 64 位二进制表示的填充前信息长度。经过这两步的处理，信息的位长=N*512+448+64=(N+1）*512。

3.7 流程逻辑

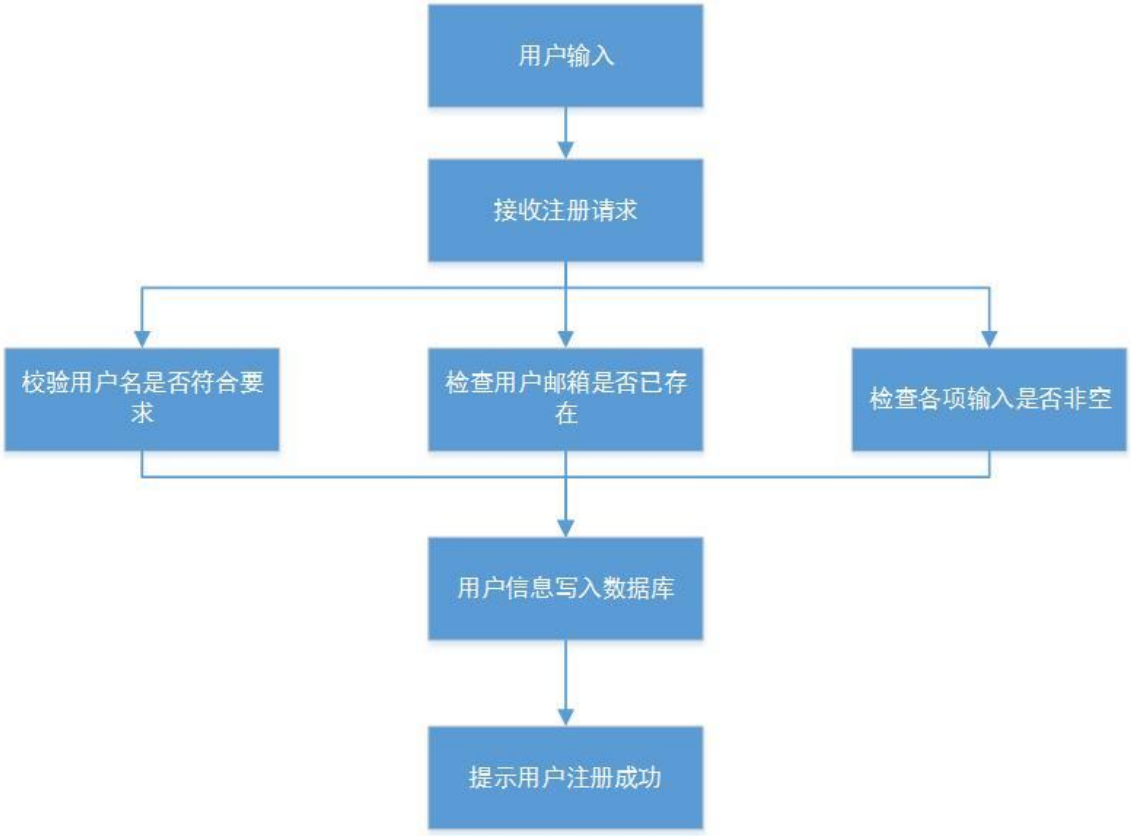

图 3

当系统接收用户的注册请求时，要进行一系列检查，包括：校验用户名是否符合要求，检查用户邮箱是否已存在，检查各项输入是否非空。若符合要求，便将用户信息写入数据库，并提示用户注册成功。

3.8 接口

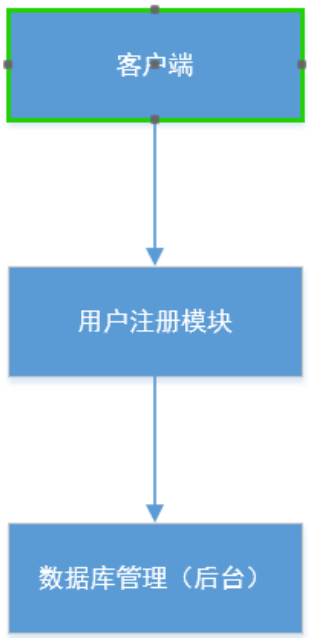

图 4

（a） 客户端发送请求

客户端按规定 POST 字段到服务端 URL；

POST 字段内容：用户名(user_name),char 类型，输入 1-20 个字符；用户密码（password),char 类型，输入密码必须为英文字母加数字组合形式，且长度大于 8 个字符，用户邮箱(user_mail),char 类型，输入 1-20 个字符。

（b） 调用数据库管理方法：

接口方法：

(1)bool saveUserInfo(String[] user)

传入参数为用户信息列表，包括所有用户输入信息；返回是否写入数据库成功；

(2)bool isRepeated(String mailBox)

传入用户邮箱，该方法在数据库中检查是否已存在该邮箱。

3.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 10MB，数据库建立于服务端，是用 MySQL 存储。

3.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

3.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

3.12 测试计划

本模块测试人员为姜美羡；输入数据为测试用户信息（包括用户名，用户密码，用户邮箱），用户名(user_name),char 类型，输入 1-20 个字符；用户密码（password),char 类型，输入密码必须为英文字母加数字组合形式，且长度大于 8 个字符,包括合法与不合法输入格式、重复与不重复用户邮箱信息。

输出：用户信息正确：跳转网页至主页；用户名称不存在：提示不存在该用户，并提示重新输入信息；用户密码输入错误：网页显示密码输入错误，并提示用户重新输入或找回密码。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

3.13 尚未解决的问题

当用户规模增大时可能面临数据库存储空间不足的问题。

4 用户登陆模块设计说明

4.1 程序描述

本程序模块主要应用于允许用户从客户端登陆本系统，从而进一步查看和使用本系统中的服务。本模块与数据库交互，允许用户提出登陆请求并接收用户发送的用户邮箱、密码并校验数据符合数据库中已存在用户条目后用户将被允许正常登录本系统。本程序模块常驻内存，能够并发处理多个用户同时提出的登陆请求。

4.2 功能

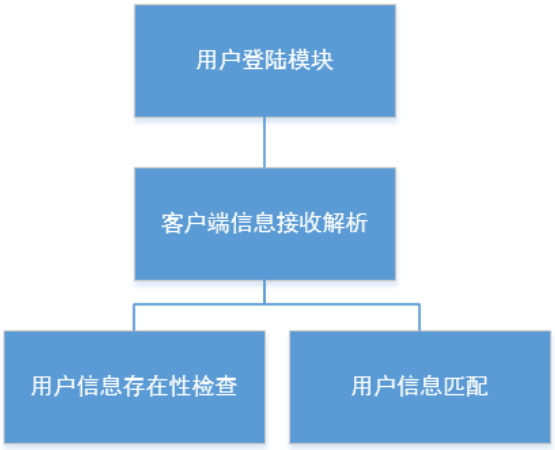

图 5

用户登录模块是客户端接收用户信息并解析，进行用户信息存在性检查和用户信息匹配。

4.3 性能

系统接收客户端用户登陆请求的延迟时间不超过 1.0 秒；

系统处理用户登陆请求并且返回登陆成功或失败信息至客户端时间上限 1.0 秒。

4.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密
用户邮箱	String	1~20 字符	N	‘’	InputBox	否
用户密码	String	8~20 字符	N	‘’	InputBox	是

4.5 输出项

输出名称	数据类型	有效范围	是否加密
Check	Bool	True/False	否

4.6 算法

本模块运行时，客户端提示用户输入用户邮箱、用户密码，并通过 JavaScript 使用正则表达式实现对用户邮箱输入合法性的检查。用户点击登陆后，客户端发送 POST 请求并由服务端接收，服务端先检查用户邮箱是否存在于数据库，若不存在则返回用户不存在错误给客户端；之后匹配密码是否与数据库中相同，若不相同则返回用户密码错误；否则返回登陆成功，客户端自动跳转页面。

4.7 流程逻辑

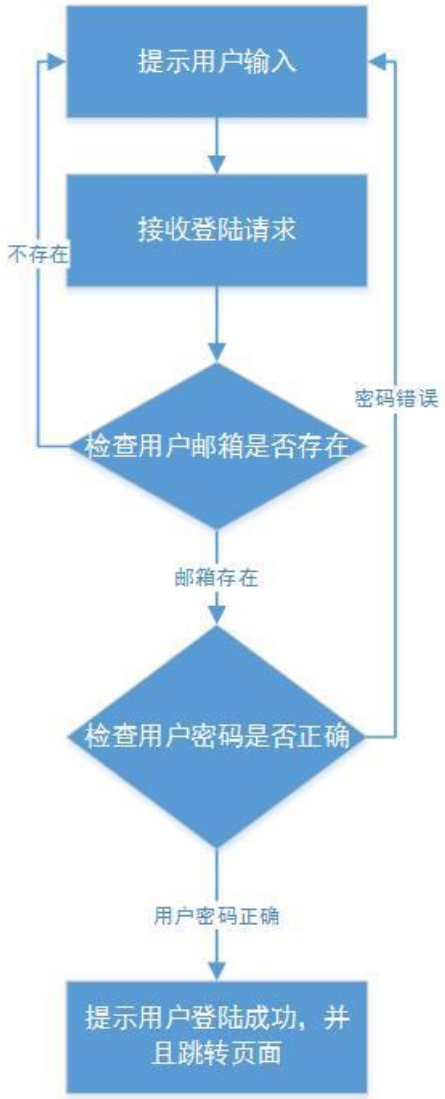

图 6

登录界面提示用户输入，系统接收登录请求。检查用户邮箱是否存在，若不存在则返回登录界面重新输入。再检查用户密码是否正确，若不正确则返回登录界面重新输入。若密码正确，则提示用户登录成功，并且跳转页面。

4.8 接口

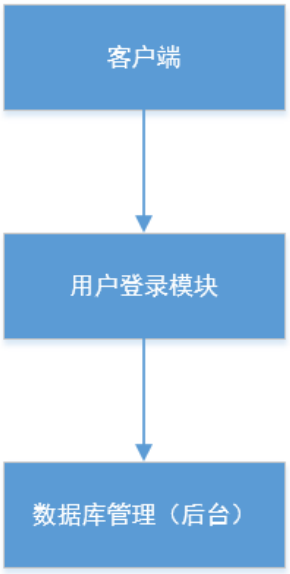

图 7

（c） 客户端发送请求

客户端按规定 POST 字段到服务端 URL；

POST 字段内容:；用户邮箱(user_mail),char 类型，输入 1-20 个字符；用户密码（password),char 类型，输入密码必须为英文字母加数字组合形式，且长度大于 8 个字符。

（d） 调用数据库管理方法：

接口方法：

(1)String searchUserInfo(String mailBox)

传入参数为用户邮箱；若数据库中存在该用户信息接口返回用户密码，若未查询到该用户信息则返回空字符串。

(2)bool isRepeated(String mailBox)

传入用户邮箱，该方法在数据库中检查是否已存在该邮箱。

4.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 10MB，数据库建立于服务端，是用 MySQL 存储。

4.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

4.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

4.12 测试计划

本模块测试人员为姜美羡；输入数据为测试用户信息（包括用户密码，用户邮箱），用户邮箱,char 类型，输入 1-20 个字符；用户密码,char 类型，输入密码必须为英文字母加数字组合形式，且长度大于 8 个字符,包括合法与不合法输入格式、重复与不重复用户邮箱信息。

输出：用户信息正确：跳转网页至主页；用户邮箱不存在：提示不存在该用户，并提示重新输入信息；用户密码输入错误：网页显示密码输入错误，并提示用户重新输入或找回密码。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

4.13 尚未解决的问题

无。

5 网络爬虫模块设计说明

5.1 程序描述

本程序模块用于将网上优质的教育资源爬下来，包括优质网站如国内的知乎、豆瓣网，国外的 goodreads 等综合资源型网站的文本资源、图片资源等，用于进行数据集的训练。同时后续也需要自动的爬虫程序用于每天抓取全网的知识信息放入神经网络进行分级分类。

5.2 功能

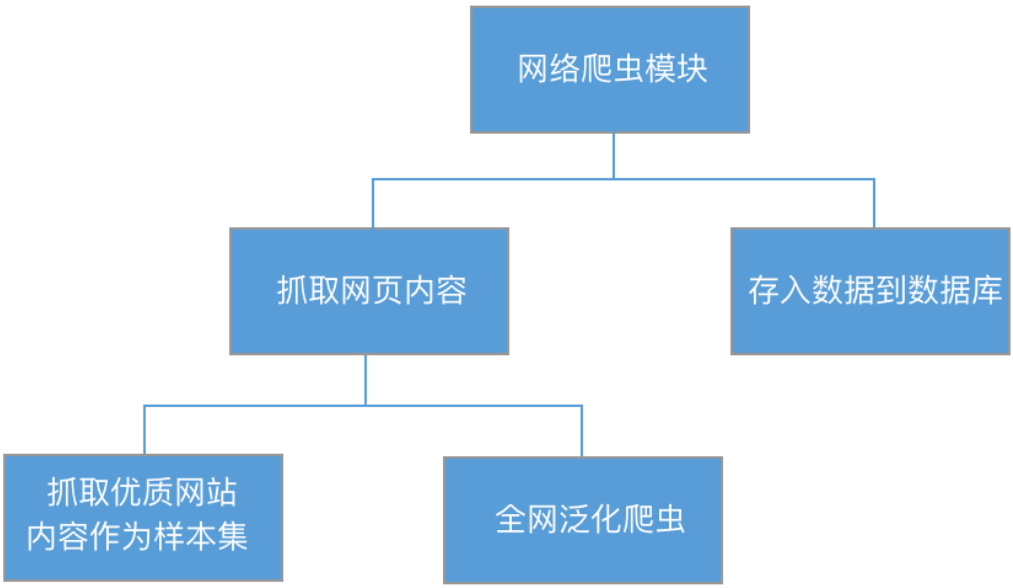

图 8

网络爬虫模块的第一步是抓取网页内容，包括抓取优质网站内容作为样本集和全网泛化爬虫，第二步将数据存入数据库。

5.3 性能

爬虫程序对一篇国外文章的抓取约为 0.3s，主要由网站响应时间决定。相比起来国内网

站的响应时间较快，能够达到 1s 抓取 6～7 篇文章。

5.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密
爬取数量	Integer	Int 有效值内	N	0	人为规定	否

5.5 输出项

输出名称	数据类型	有效范围	是否加密
Labels	String[]	-	否
BookName	String	-	否
BookContent	String	-	否
Author	String	-	否

5.6 算法

1、利用 urllib2 与目标网站建立连接并发送 request。

2、得到返回的 response，获取目标网页的 html 内容

3、利用正则表达式将目标输出项提取出来

5.7 流程逻辑

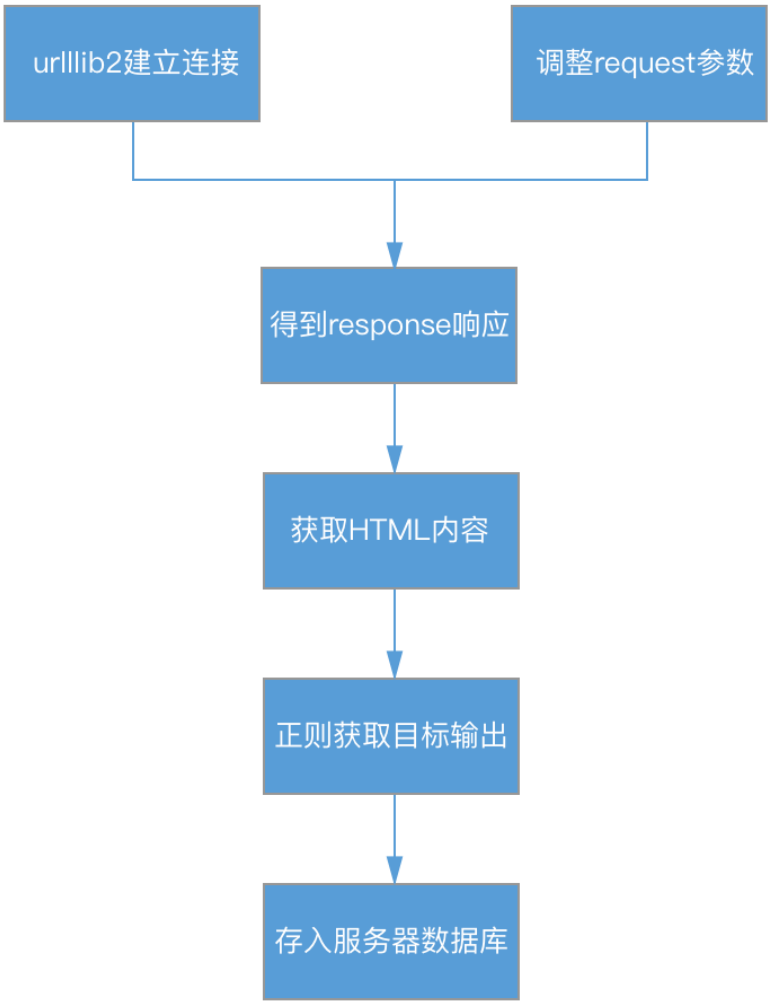

图 9

爬虫程序调用 urllib2 库，调整 request 参数，得到 response 响应以获取 HTML 内容。通过正则表达式匹配目标信息，获取输出并存入服务器数据库。

5.8 接口

网络爬虫模块封装成方法，给出接口：Boolean crawl_from_goodreads(int page_number) 输入爬取的数量，返回 true 或者 false，表示成功完成或者失败。

5.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 100MB，数据库建立于服

务端，是用 MySQL 存储。

5.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

5.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

5.12 测试计划

本模块测试人员为姜美羡；输入数据为希望爬取的数量，输入正负数,包括合法与不合法输入格式。

输出：封装函数返回 true 或 false，并且检测数据库里已经存入希望存入的数据。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

5.13 尚未解决的问题

网络爬虫的泛化爬虫模型未成熟，不能很好地自动式爬取不同类型的网站。

6 文本分类模块设计说明

6.1 程序描述

本程序模块主要应用于将网络爬虫所获取的信息依据其内容在给定的标签集合内进行文本分类。本模块需要预先使用样本文本对分类器进行训练。本程序模块无需常驻内存，并且最终生成标签列表不可重复、避免相似标签，同时能够并发处理多个文本分类请求。

6.2 功能

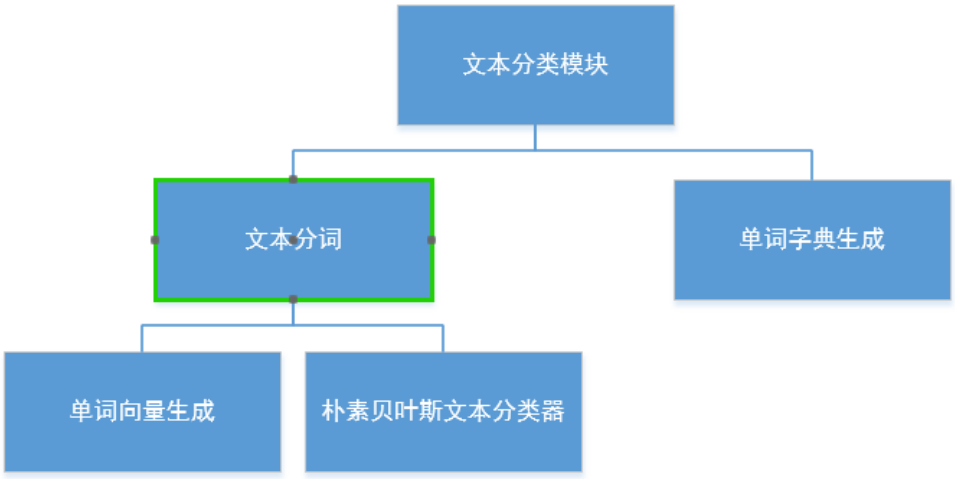

图 10

文本分类模块由文本分词和单词字典构成。文本分词分为两部分：单词向量生成和朴素贝叶斯文本分类器。

6.3 性能

系统对一篇 500 字左右文章进行分类的工作时间不超过 5.0 秒。

6.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密

书本 ID	Int	1~8 字符	N	‘’	参数传值	否
文本内容	String	8~20 字符	Y	‘’	参数传值	否

6.5 输出项

输出名称	数据类型	有效范围	是否加密
Check	Bool	True/False	否

6.6 算法

本模块由系统管理人员启动或停止服务，本功能使用时需传入书本 ID、文本内容（书本简介等），然后使用结巴分词对文本内容进行切割并生成单词向量，向贝叶斯分类器中传入单词矩阵，分类结果为标签矩阵，之后按照标签字典返回相应标签。

本模块主要涉及朴素贝叶斯文本分类算法：

朴素贝叶斯算法里面的各个类条件是独立的，所以一会在后面的计算中会起到很多方便的作用。它的原理如下：

首先在这里用到了一个概率公式：

P(B|A)的意思是在 A 事件的情况下，发生 B 事件的概率，可以理解为概率论中的条件概率，而贝叶斯公式的巨大作用就是对因果关系进行了交换，通过上面的公式就可以计算P(A|B)的概率，只要通过上述的转换。

朴素贝叶斯分类的正式定义如下：

1、 设 x={a1,a2,……,am}为一个待分类项，而每个 a 为 x 的一个特征属性。

2、 有类别集合 C={y1,y2,……yn}。

3、 计算$P(y_1|x), P(y_2|x), ..., P(y_n|x)$ 。

4、 如果 P(yk|x)=max{P(y1|x,P(y2|x)……P(yn|x))}，则 属于 yk。

那么现在的关键就是如何计算第 3 步中的各个条件概率。我们可以这么做：

1、找到一个已知分类的待分类项集合，这个集合叫做训练样本集。

2、统计得到在各类别下各个特征属性的条件概率估计。即$P(a_1|y_1), P(a_2|y_1), ..., P(a_m|y_1); P(a_1|y_2), P(a_2|y_2), ..., P(a_m|y_2); ...; P(a_1|y_n), P(a_2|y_n), ..., P(a_m|y_n)$

3、如果各个特征属性是条件独立的，则根据贝叶斯定理有如下推导：

$P(y_i|x) = \frac{P(x|y_i)P(y_i)}{P(x)}$ 。

因为分母对于所有类别为常数，因为我们只要将分子最大化皆可。又因为各特征属性是条件独立的，所以有：

$$P(x|y_i)P(y_i) = P(a_1|y_i)P(a_2|y_i)...P(a_m|y_i)P(y_i) = P(y_i)\prod_{j=1}^{m} P(a_j|y_i)$$

可以根据数量上的统计进行计算，本项目的程序将基于此原理进行。

6.7 流程逻辑

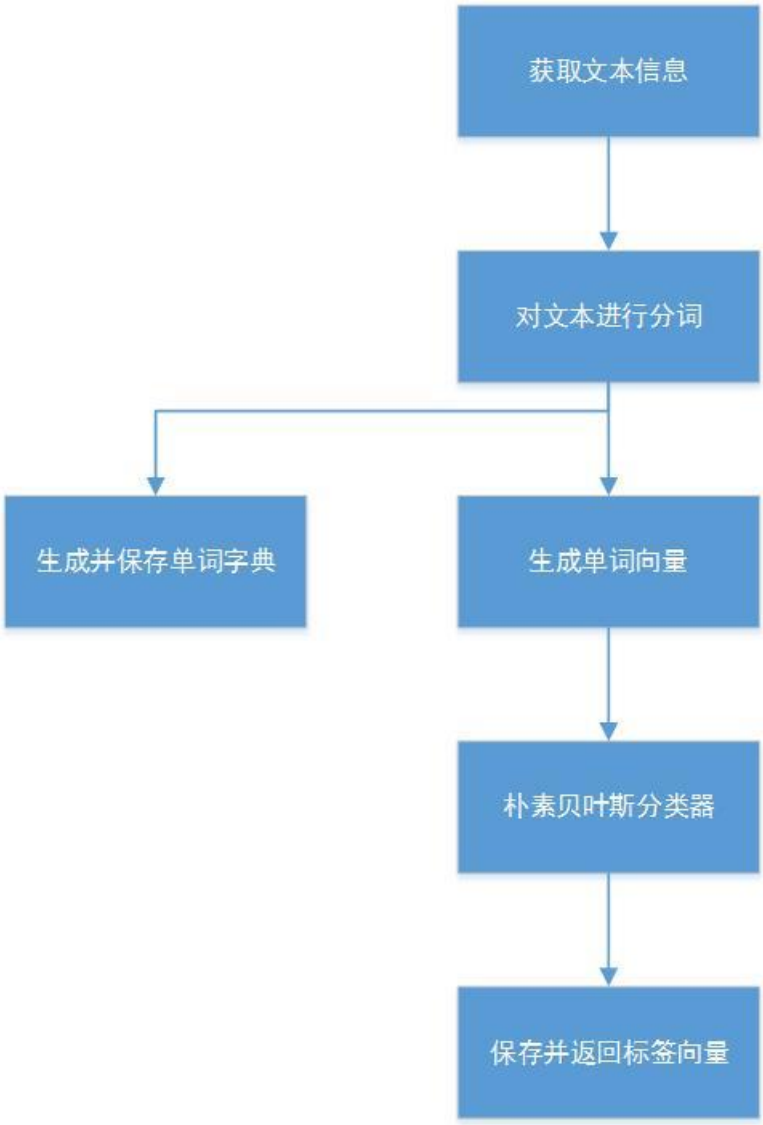

图 11

对文本进行分词，生成并保存单词字典，将生成的单词向量输入背也是分类器，保存并返回标签向量。

6.8 接口

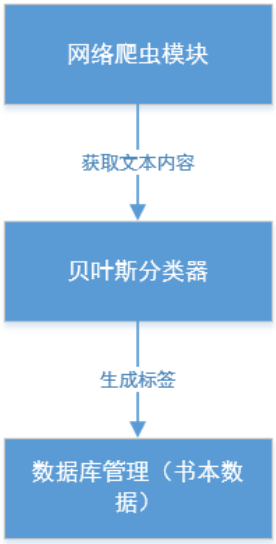

图 12

接口方法：

String[] bayesClassify(string text,int bookID)

该接口传入书本简介文本内容，以及书本 ID，分类之后返回生成的标签列表。

6.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 100MB，数据库建立于服务端，是用 MySQL 存储。

6.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

6.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

6.12 测试计划

本模块测试人员为姜美羡；输入数据为测试文本，输入 300-500 个字符,包括合法与不合法输入格式、重复与不重复用户邮箱信息。

输出：依据测试文本内容产生的分类标签列表，与正确分类列表对比计算准确率。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

6.13 尚未解决的问题

网络爬虫所获取的文本内容有限，标签类别有限，分类器性能与精度比较差。

7 系统推荐模块设计说明

7.1 程序描述

本程序功能目的在于为特定用户推送可能喜好的书籍列表。使用本功能前提是用户在本系统中注册过，且有过一定数目的浏览记录。使用本功能时用户向服务端发送获取推荐书单列表请求，服务端根据发送请求的用户邮箱，调用本功能生成相似用户列表并返回推荐书籍列表。

7.2 功能

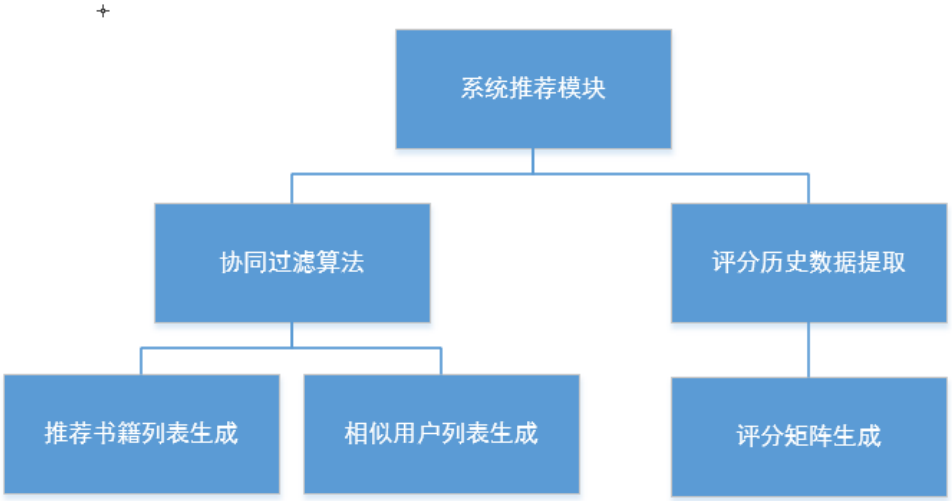

图 13

系统推荐模块由评分历史数据提取和协同过滤算法组成。评分数据以矩阵的形式生成，协同过滤算法依靠推荐相似用户生成推荐书籍列表。

7.3 性能

系统返回生成的推荐数目列表时间不超过 3.0 秒。

7.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密
用户邮箱	String	10~20 字符	N	‘’	InputBox	否

7.5 输出项

输出名称	数据类型	有效范围	是否加密
RecommendBooks	String[]	-	否

7.6 算法

本模块运行时，客户端向服务端发送生成推荐列表请求并附带用户邮箱字段，然后该模块会检查用户历史记录里是否存在该用户的浏览记录，若没有则返回空值；若有足够的历史浏览记录，则从数据库中调取所有历史记录并传入协同过滤算法模块，最终生成并返回用户推荐书籍列表。

本功能模块主要涉及协同过滤算法。

主要的功能是预测和推荐。算法通过对用户历史行为数据的挖掘发现用户的偏好，基于不同的偏好对用户进行群组划分并推荐品味相似的商品。协同过滤推荐算法分为两类，分别是基于用户的协同过滤算法，和基于物品的协同过滤算法。简单的说就是：人以类聚，物以群分。本功能模块使用基于用户的协同过滤算法。

基于用户的协同过滤算法基于这样一个事实，如果 A 和 B 在电影方面的喜好相同，那么把 B 喜欢的的电影推荐给 A 是有道理的。 根据这个事实，基于用户的协同过滤算法出现了。根据这个事实需要求出两个用户的相似度。这个相似度可以是公式 1 或者公式 2(余弦公式)。 如果想计算每两个用户的相似度需要的时间复杂度为 O(n*n*d)。n 为用户数目，d 为商品的数目。

$$\frac{|N(u)\cap N(v)|}{|N(u)\cup N(v)|}$$ （公式 1）

$$\frac{|N(u)\cap N(v)|}{\sqrt{|N(v)|*|N(u)|}}$$ （公式 2）

通过公式 1 或者 2 我们能得到一个相似度矩阵。然而在很多应用中这个相似度矩阵式非常稀疏的。也就是很多用户相互之间没有对相同的商品产生行为。如果我们直接先把相似度不为 0 的用户对数求出来，然后只计算这些不为 0 的用户对，这样子会剩很多复杂度。 用数组 C[u][v]记录用户 u 和 v 有相同商品行为的数目。首相建立一个倒排表。每个物品都保存被产生过行为的用户。然后对于每个物品所有的用户对数（u,v）,C[u][v]加 1。这样结束以后就可以只利用相似度不为 0 的用户对数了。

得到相似度矩阵后利用公式 3 预测用户 u 对物品 i 的感兴趣程度。其中 S(u,k)表示与用户 u 最接近的 k 个用户 N(i)表示对物品 i 有过行为的用户集合。 wuv 表示用户 u 和 v 的相似度，rvi 表示用户 v 对物品 i 的感兴趣程度。

$$p(u,i)=\sum_{v\in S(u,k)\cap N(i)} w_{uv}r_{vi}$$ （公式 3）

7.7 流程逻辑

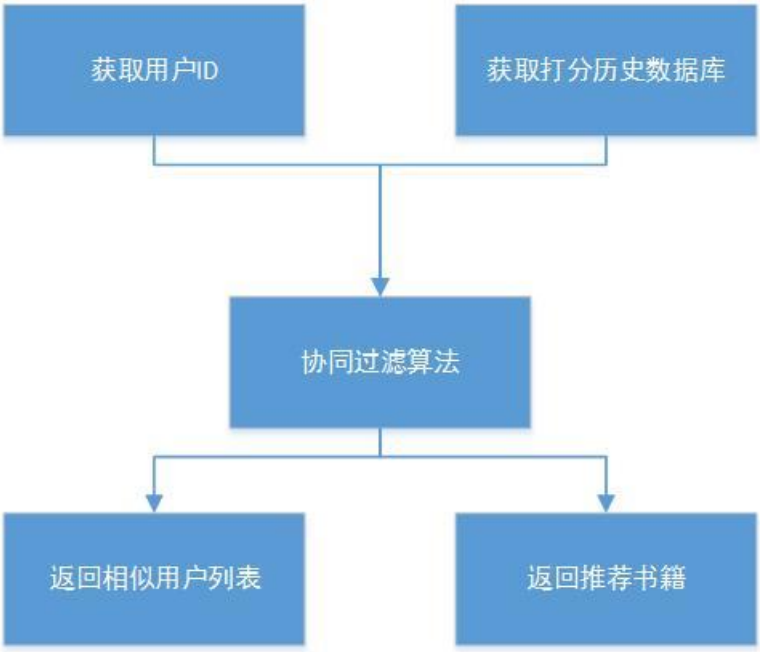

图 14

协同过滤算法通过获取用户 ID 和打分历史数据库返回相似用户列表和推荐书籍。

7.8 接口

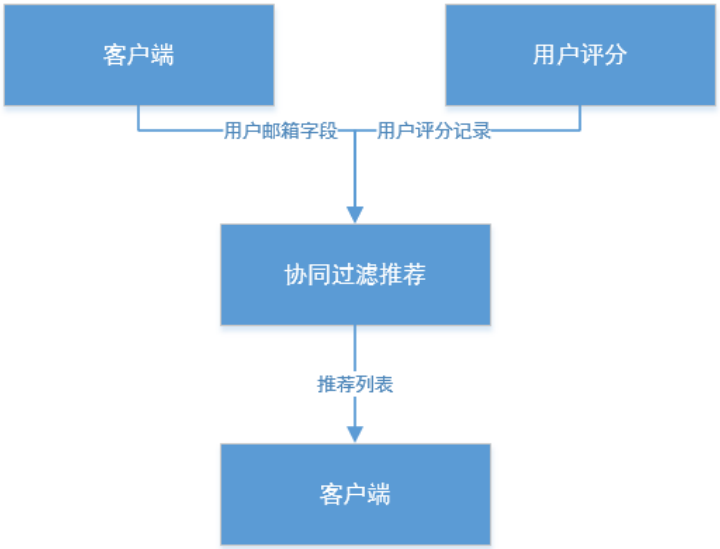

图 15

（e） 客户端发送请求

客户端发送 GET 请求至服务端请求推荐功能即可，并且附带用户邮箱字段，char 类型，输入 1-20 个字符。返回 String 列表类型的推荐书籍目录。

接口方法：

String[] getRecommendList(String mailBox)

传入用户邮箱地址，该方法返回生成的对该用户的推荐书籍列表。

7.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 10MB，数据库建立于服务端，是用 MySQL 存储。

7.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

7.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

7.12 测试计划

本模块测试人员为姜美羡；输入数据为测试用户邮箱地址字段，并且数据库中应准备相应的历史评分记录。

输出：用户信息正确：返回用户推荐书籍列表；用户信息不存在：提示不存在该用户，并提示重新输入信息。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

7.13 尚未解决的问题

该功能模块建立初期因为缺乏大量用户信息而无法实现推荐功能。

8 用户评分模块设计说明

8.1 程序描述

此模块为留给用户为书本打分使用，基于统计与分析，能够获取到书本的优质信息、受欢迎程度，每一个用户的喜好、偏好倾向。满分为 5 分，才用取平均的方式，将书本的评分显示给用户看，并且每一个 ip 地址也可以自主为每一本书打分。

8.2 功能

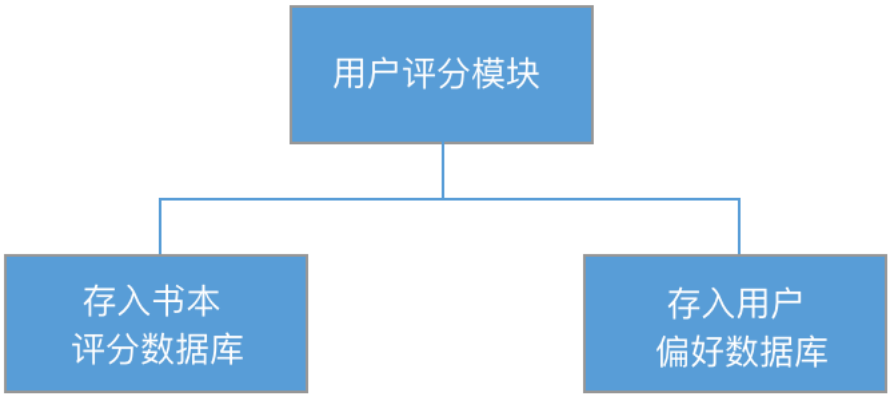

图 16

系统获取用户评分分别存入书本评分数据库和用户偏好数据库。

8.3 性能

评分模块的响应时间主要在于于数据库之间的响应速度，一般评分时间在 0.2～0.4s。

8.4 输入项

输入名称	数据类型	有效范围	能否为空	默认值	输入方式	是否加密
评分分数	Integer	Int 有效值内	N	0	用户选择	否

8.5 输出项

输出名称	数据类型	有效范围	是否加密
Average_Score	Double	0.00~5.00	否

8.6 算法

无

8.7 流程逻辑

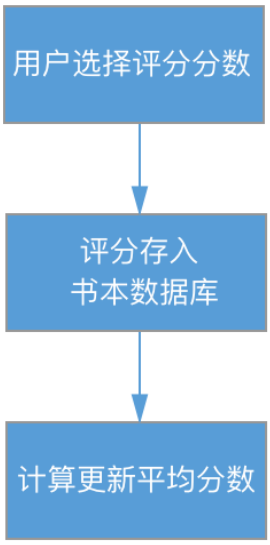

图 17

系统获取用户评分分数并存入书本数据库，计算更新平均分数并显示于书籍信息页面。

8.8 接口

网络爬虫模块封装成方法，给出接口：Double Set_Score(int score)

输入用户打分的分数，函数内部完成存取数据，计算平均值的过程，将更新的平均值返回。

8.9 存储分配

本程序主要将用户数据存储于数据库，分配硬盘内存不超过 100MB，数据库建立于服务端，是用 MySQL 存储。

8.10 注释设计

模块首部添加注释：包括该模块主要开发者姓名，版本号，最终版本开发时间（年/月/日），该模块功能简述等内容；

各分支点处添加注释：描述分支点条件，以及每个分支大致的处理流程或功能描述；

对各个变量添加注释：描述每个变量大致功能，以及变量所需的约束（数值范围）等；

对每个 FOR 或 WHILE 循环添加注释：描述每次循环所实现的功能，以及每次循环对数值的更新。

8.11 限制条件

服务端提供网络带宽不大于 1MB；

服务端存储空间最大为 40GB；

处理器（1GHz）性能限制；

数据库用户数据字段长度限制在 20 个字符以内；

服务器正常运行。

8.12 测试计划

本模块测试人员为姜美羡；输入数据为评价的分数。检测从评价、存入数据到呈出平均

分的过程是否出错。

输出：返回平均分数，观察是否计算错误。

进度安排：2017/5/6 ~ 2017/5/7 测试完毕本功能测试用例。

8.13 尚未解决的问题

因前期缺少大量用户的大数据的支持，因此很难建立起完善的评分体系。