|  |
| --- |
|  |

视障人士友好的资讯辅助软件

详细设计说明书



四川大学软件学院

二〇二二年五月

目录

[1 引言 8](#_Toc104413216)

[1.1 编写目的 8](#_Toc104413217)

[1.2 背景 8](#_Toc104413218)

[1.3 定义 8](#_Toc104413219)

[1.4 参考资料 9](#_Toc104413220)

[2 设计概述 9](#_Toc104413221)

[2.1 任务和目标 9](#_Toc104413222)

[2.2 需求概述 10](#_Toc104413223)

[2.3 运行环境 11](#_Toc104413224)

[2.3.1 硬件环境 11](#_Toc104413225)

[2.3.2 软件环境 11](#_Toc104413226)

[2.4 条件与限制 11](#_Toc104413227)

[3 系统结构 12](#_Toc104413228)

[3.1 系统架构 12](#_Toc104413229)

[3.2 系统功能模块 13](#_Toc104413230)

[4 模块详细设计 16](#_Toc104413231)

[4.1 用户注册功能 16](#_Toc104413232)

[4.1.1 功能 16](#_Toc104413233)

[4.1.2 性能 16](#_Toc104413234)

[4.1.3 输入项 16](#_Toc104413235)

[4.1.4 输出项 16](#_Toc104413236)

[4.1.5 设计方法 17](#_Toc104413237)

[4.1.6 流程逻辑 17](#_Toc104413238)

[4.1.7 接口 17](#_Toc104413239)

[4.1.8 限制条件 17](#_Toc104413240)

[4.1.9 测试计划 17](#_Toc104413241)

[4.2 新闻推荐 17](#_Toc104413242)

[4.2.1 功能 17](#_Toc104413243)

[4.2.2 性能 18](#_Toc104413244)

[4.2.3 输入项 18](#_Toc104413245)

[4.2.4 输出项 18](#_Toc104413246)

[4.2.5 设计方法 18](#_Toc104413247)

[4.2.6 流程逻辑 22](#_Toc104413248)

[4.2.7 接口 23](#_Toc104413249)

[4.2.8 限制条件 23](#_Toc104413250)

[4.2.9 测试计划 23](#_Toc104413251)

[4.3 新闻推送模块 23](#_Toc104413252)

[4.3.1 功能 23](#_Toc104413253)

[4.3.2 性能 23](#_Toc104413254)

[4.3.3 输入项 23](#_Toc104413255)

[4.3.4 输出项 24](#_Toc104413256)

[4.3.5 设计方法（算法） 24](#_Toc104413257)

[4.3.6 流程逻辑 24](#_Toc104413258)

[4.3.7 接口 24](#_Toc104413259)

[4.3.8 限制条件 24](#_Toc104413260)

[4.3.9 测试计划 24](#_Toc104413261)

[4.4 语音阅读模块 25](#_Toc104413262)

[4.4.1 功能 25](#_Toc104413263)

[4.4.2 性能 25](#_Toc104413264)

[4.4.3 输入项 25](#_Toc104413265)

[4.4.4 输出项 25](#_Toc104413266)

[4.4.5 设计方法（算法） 25](#_Toc104413267)

[4.4.6 流程逻辑 25](#_Toc104413268)

[4.4.7 接口 26](#_Toc104413269)

[4.4.8 限制条件 26](#_Toc104413270)

[4.4.9 测试计划 26](#_Toc104413271)

[4.5 关键词搜索模块 26](#_Toc104413272)

[4.5.1 功能 26](#_Toc104413273)

[4.5.2 性能 26](#_Toc104413274)

[4.5.3 输入项 26](#_Toc104413275)

[4.5.4 输出项 26](#_Toc104413276)

[4.5.5 设计方法（算法） 27](#_Toc104413277)

[4.5.6 流程逻辑 27](#_Toc104413278)

[4.5.7 接口 27](#_Toc104413279)

[4.5.8 限制条件 27](#_Toc104413280)

[4.5.9 测试计划 27](#_Toc104413281)

[4.6 拍照识别模块 27](#_Toc104413282)

[4.6.1 功能 27](#_Toc104413283)

[4.6.2 性能 28](#_Toc104413284)

[4.6.3 输入项 28](#_Toc104413285)

[4.6.4 输出项 28](#_Toc104413286)

[4.6.5 设计方法（算法） 28](#_Toc104413287)

[4.6.6 流程逻辑 28](#_Toc104413288)

[4.6.7 接口 28](#_Toc104413289)

[4.6.8 限制条件 28](#_Toc104413290)

[4.6.9 测试计划 28](#_Toc104413291)

[4.7 管理员登录模块 29](#_Toc104413292)

[4.7.1 功能 29](#_Toc104413293)

[4.7.2 性能 29](#_Toc104413294)

[4.7.3 输入项 29](#_Toc104413295)

[4.7.4 输出项 29](#_Toc104413296)

[4.7.5 设计方法 29](#_Toc104413297)

[4.7.6 流程逻辑 30](#_Toc104413298)

[4.7.7 接口 30](#_Toc104413299)

[4.7.8 限制条件 30](#_Toc104413300)

[4.7.9 测试计划 30](#_Toc104413301)

[4.8 新闻爬取模块 30](#_Toc104413302)

[4.8.1 功能 30](#_Toc104413303)

[4.8.2 性能 31](#_Toc104413304)

[4.8.3 输入项 31](#_Toc104413305)

[4.8.4 输出项 31](#_Toc104413306)

[4.8.5 设计方法 31](#_Toc104413307)

[4.8.6 流程逻辑 31](#_Toc104413308)

[4.8.7 接口 31](#_Toc104413309)

[4.8.8 限制条件 31](#_Toc104413310)

[4.8.9 测试 31](#_Toc104413311)

[4.9 用户行文信息统计模块 32](#_Toc104413312)

[4.9.1 功能 32](#_Toc104413313)

[4.9.2 性能 32](#_Toc104413314)

[4.9.3 输入项 32](#_Toc104413315)

[4.9.4 输出项 32](#_Toc104413316)

[4.9.5 设计方法 32](#_Toc104413317)

[4.9.6 流程逻辑 33](#_Toc104413318)

[4.9.7 接口 33](#_Toc104413319)

[4.9.8 限制条件 33](#_Toc104413320)

[4.9.9 测试 33](#_Toc104413321)

[4.10 新闻管理模块 33](#_Toc104413322)

[4.10.1 功能 33](#_Toc104413323)

[4.10.2 性能 33](#_Toc104413324)

[4.10.3 输入项 34](#_Toc104413325)

[4.10.4 输出项 34](#_Toc104413326)

[4.10.5 设计方法 34](#_Toc104413327)

[4.10.6 流程逻辑 34](#_Toc104413328)

[4.10.7 接口 34](#_Toc104413329)

[4.10.8 限制条件 34](#_Toc104413330)

[4.10.9 测试 34](#_Toc104413331)

# 1 引言

## 1.1 编写目的

此详细设计说明书对视障人士友好的咨询辅助软件进行了实现层面上的要求与说明，对系统设计的算法与原理与具体实现进行了详细说明，为进行之后的实现、优化与测试提供参考。

该详细设计说明书的预期读者为本项目开发成员以及对该项目感兴趣、拓展和维护该项目的相关人员。

## 1.2 背景

在AI浪潮之下各种智能APP让应用程序的体验感越来越好，深刻地改变着我们的生活，然而，我们希望AI技术与移动互联网能够覆盖更多需要帮助的人群。比如半盲与低视力等视障人群，他们存在视力障碍，无法通过佩戴眼镜等方式进行矫正，我们希望能够运用云上资源以及互联网技术的帮助，让他们享受到科技进步的成果。

## 1.3 定义

移动云：移动云是中国移动基于5G和自研先进技术打造的5G智慧云品牌，为客户提供行业领先的云主机/云服务器、云手机、云空间、大数据、人工智能等产品和专业服务，针对不同行业打造场景化解决方案。

视障人士：根据残疾人分类和分级来看,视力残疾被分为盲和低视力。视野半径小于10度,属于二级盲;视野半径小于5度,称为一级盲。根据最新资料显示,我国盲人数量已达到1731万人,总人口占比达到1.26%。视障人士包含了盲人以及弱视人士，视障人士并非单独指的是盲人，也可以是能感光，但是通过眼镜等方式进行矫正的人士，此处，主要指的是有一定感光能力，但是无法看清内容的人士；

视障人士拍照：完整的软件应该是可以通过视频对盲人进行协助，并通过视频对相关文字进行识别，但是视频分析相对来说工作量太大所以此处对功能进行了简化（即拍照）。

无障碍阅读工具：出厂就具备无障碍屏幕听读的手机设备包括但不限于，华为、小米、锤子-坚果、oppo、iPhone。

OCR：（Optical Character Recognition，光学字符识别）是指电子设备（例如扫描仪或数码相机）检查纸上打印的字符，通过检测暗、亮的模式确定其形状，然后用字符识别方法将形状翻译成计算机文字的过程，相关技术有现成的API进行调度使用（考虑到公平性，限制只能使用移动云相关API）借助该能力，可以实现书籍的拍照阅读，比如：可以通过手机支架+书籍固定器+手机辅助功能，以及已有的OCR的API接口实现，视障人士的拍照阅读。大部分手机系统都有自带的无障碍阅读工具，可以帮助视障人士使用手机软件。

## 1.4 参考资料

[1]《项目计划书-第一组》

[2]《需求规格说明书-第一组》

[3]《概要设计说明书-第一组》

[4] Wu C, Wu F, An M, et al. Neural news recommendation with attentive multi-view learning[J]. arXiv preprint arXiv:1907.05576, 2019.

# 2 设计概述

## 2.1 任务和目标

该项目的概要设计目标是使软件的整个设计与实现过程必须在保证软件质量的前提下通过生产能力的提高、员工作效率的提高等等使软件开发成本最小化。

目前移动云上有着大量的运算模型接口及算力资源，但是单纯的模型接口无法直接供一般的用户使用，必须创新方法将这些功能结合起来，形成应用场景。以“视障人士友好的咨询辅助软件”为主要方向，开发具有创新性并符合行业发展趋势，商业应用，创意设计方面的优秀项目。

## 2.2 需求概述

本系统是对视障人士友好的资讯辅助软件，软件将考虑视障人士使用手机的方式以及查看新闻时的交互需求，减少视障人士因视觉缺陷带来的使用障碍，拥有良好的使用体验。系统的功能性需求分为以下几个方面：本系统提供服务器Web端界面，以及移动端语音交互，新闻推荐，提示性拍照等功能。项目的需求对应表如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 需求编号 | 需求 | 章节号 | 备注 |
| 1 | 001 | 管理员登录 | 3.1.1 |  |
| 2 | 002 | 查看服务器运行状况 | 3.1.2 |  |
| 3 | 003 | 修改新闻获取渠道 | 3.1.3 |  |
| 4 | 004 | 算法模型管理 | 3.1.4 |  |
| 5 | 005 | 用户管理 | 3.1.5 |  |
| 6 | 006 | 新闻信息统计 | 3.2.1 |  |
| 7 | 007 | 用户行为信息统计 | 3.3.1 |  |
| 8 | 008 | 友好的交互方式 | 3.4.1 |  |
| 9 | 009 | 用户登录 | 3.4.2 |  |
| 10 | 010 | 新闻推送 | 3.4.3 |  |
| 11 | 011 | 提示性拍照 | 3.4.4 |  |

表1 需求对应表

## 2.3 运行环境

### 2.3.1 硬件环境

环境设备：云服务器、内存>=16 GB、GPU、显存>=11 GB、1T 固态硬盘（SSD）；

使用设备：手机、笔记本电脑/台式电脑/平板电脑，并能连接 WLAN 或者蜂 窝网络；

其他设备：鼠标、键盘。

### 2.3.2 软件环境

运行环境：Linux 操作系统、Windows 操作系统；

使用环境：Windows 操作系统、Android 操作系统、IOS 操作系统、Chrome。

## 2.4 条件与限制

软件开发能否有效推进以及需求能否全面实现取决于以下条件：

（1）推荐模型、目标识别模型训练数据的样本量。

（2）项目成员的沟通、协调、配合的程度与效率。

（3）服务器的的工作效能，硬件条件限制

# 3 系统结构

## 3.1 系统架构



图1 系统架构图

系统利用HTTP协议解决Web客户端、手机客户端、新闻网站以及九天人工智能平台的通信，利用响应的网络协议与数据库做通信。Web客户端提供给系统管理员，方便系统管理员操作整个系统，调整服务器提供的服务；系统的主流用户将使用手机客户端享受系统提供的服务。服务器提供的服务主要在于新闻推荐、新闻查询以及更新新闻数据。比较基础的智能服务将由客户端直接调用九天人工智能平台获取服务。

Web客户端要求能够即时性配置系统服务；而服务器端在承载高并发访问的同时还要做到定时更新系统的新闻数据；手机客户端需要根据具体的服务内容从不同的模块获取服务。数据库部分负责系统新闻数据以及用户数据的存储。

## 3.2 系统功能模块

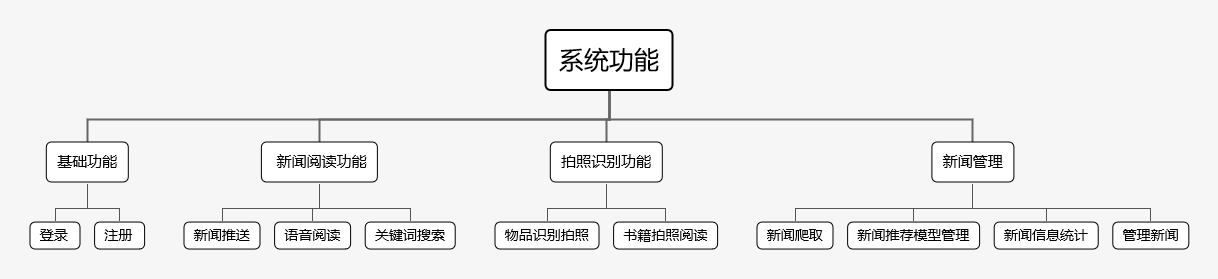


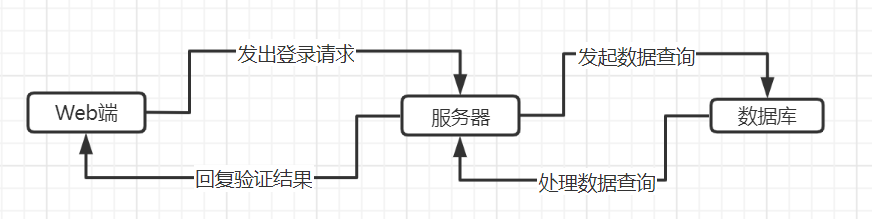
图2 系统功能模块图

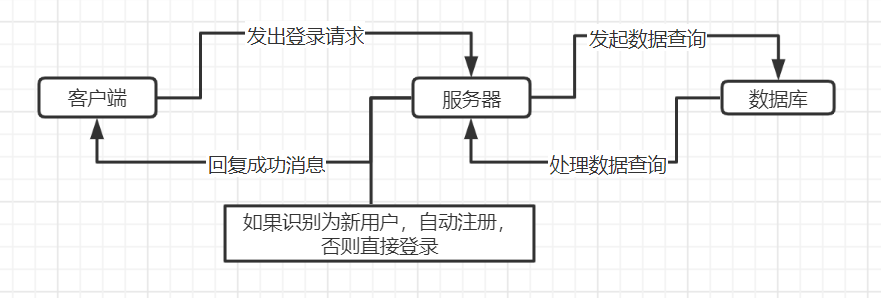
管理员界面呈现（包括登录界面、服务器运行状况界面、新闻来源编辑界面、算法模型管理界面）将在Web客户端实现。

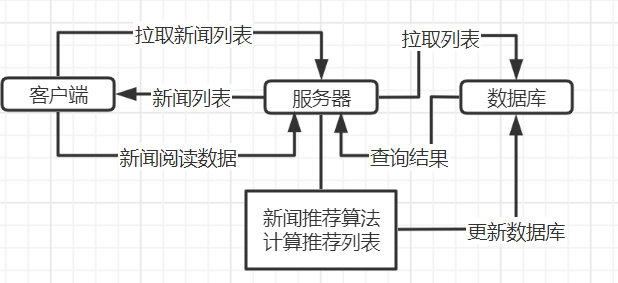
手机端服务呈现（包括用户与软件的所有可能的交互、新闻推送界面、新闻搜索界面、提示性拍照功能呈现）将在手机客户端实现。

用户登录数据验证、定时更新新闻数据、新闻推送、新闻查询处理将在服务端完成，之后通过网络信道向客户端反馈结果。

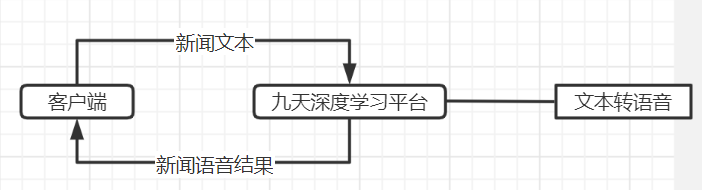
数据库将完成数据的存储，解决数据访问的问题。

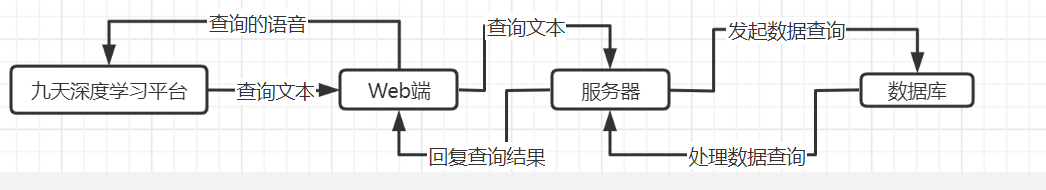
管理员登录：

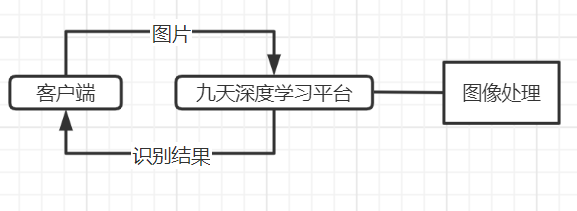
用户注册：

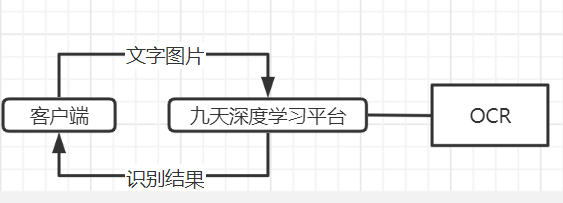
新闻推送：

语音阅读：



关键词搜索：

物体识别拍照

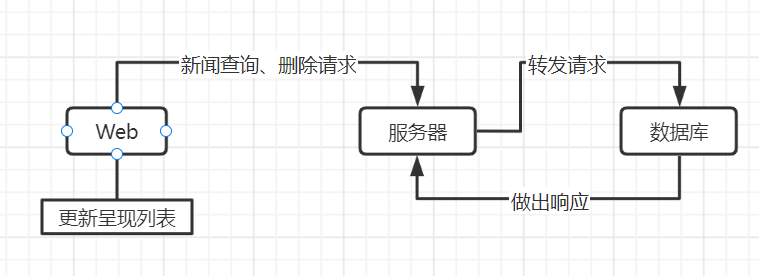
书籍拍照阅读

新闻爬取

这一部分中，服务器将从响应的新闻网站获取当前的新闻数据。

新闻信息统计：

在这一部分中，服务器利用客户端反馈的阅读数据生成相关新闻的阅读信息，这一部分的信息将会用作推荐算法的输入，并且呈现给管理员。

新闻管理：

# 4 模块详细设计

## 4.1 用户注册功能

### 4.1.1 功能

移动端需要用户登录后才能使用，注册模块能帮助没有账号的用户，自动快速注册一个登录账号。

### 4.1.2需求对应

对应需求009

### 4.1.3 性能

该模块属于建立在移动安卓客户端的实现用户交互的功能性模块，对响应速度的要求较高。 当用户初次打开进入软件后，系统会自动获取用户手机的IMEI码，系统需要在3秒内注册成功。

### 4.1.4 输入项

输入项为用户手机的IMEI码。

### 4.1.5 输出项

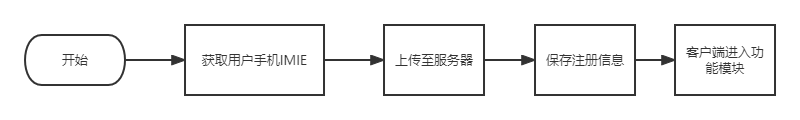
无。

### 4.1.6 设计方法

该模块实现方法是在安卓页面中用js生命钩子函数绑定获取IMEI码js函数，然后将数据发送至后台。数据成功发送至后台后，需将数据存入对应的数据库中，以保证用户后续能登录验证通过。数据成功保存至数据库后，用户直接进入到软件内。 反之，软件直接退出。 用户的IMEI数据以POST请求方式提交到后台。

|  |  |
| --- | --- |
| Function:user registration | |
| Input: The number IMEI of the user phone | |
| 1 | IMEI= plus.device.getInfo(); //获取用户的IMEI码 |
| 2 | getNewsList (IMEI)//在获取新闻列表时自动发送IMEI码并注册，发送至后端 |
| 3 | IF(select \* from user where user.IMEI==null){ //对IMEI码进行注册 |
| 4 | regist(IMEI) |
| 5 | } |
| 6 | return "SUCESS" //返回注册成功信息 |

### 4.1.7 流程逻辑



### 4.1.8 接口

用户注册接口，后端接收前端发起的GET请求，请求数据为对应的注册表单数据。接口返回注册成功或失败数据。

### 4.1.9 限制条件

成功获取用户手机的IMEI码。

### 4.1.10 测试计划

用户点击进入软件，查看后台数据库是否保存用户的IMEI码，软件客户端是否正常进入软件功能界面。 接口测试：使用POSTMAN工具，采用POST请求方式，构建好对应的表单数据后，点击“Send”发起请求，对注册接口的测试。

## 4.2 新闻推荐

### 4.2.1 功能

推荐模块包含定时推荐模块以及在线推荐模块，定时模块根据不同用户的兴趣每三小时计算出给所有用户所有类别所推荐的新闻并存入数据库，服务器端在用户使用软件时候将数据库中的数据推送给该用户，实时推荐模块能根据用户历史行为实时给用户推荐其感兴趣的新闻，当用户阅读完定时推荐模块推荐的新闻就会请求实时推荐模块实时为用户推荐新闻，定时推荐与实时推荐的结合，能给与用户更好的体验。

### 4.2.2需求对应

对应需求010

### 4.2.3 性能

用户在体验阅读新闻功能频繁切换不同种类新闻时，后台能在3秒内将推荐的新闻进行推送。

### 4.2.4 输入项

新闻的种类或无。

### 4.2.5 输出项

相应种类的推荐新闻。

### 4.2.6 设计方法

推荐模块分为定时推荐和实时推荐，推荐过程分为召回阶段和排序阶段，两模块在召回阶段和排序阶段使用了相同的算法，只是细节上有些不同，召回阶段使用基于文章向量相似度的协同过滤和基于矩阵分解的协同过滤两种算法作为召回阶段的算法，同时，在召回阶段还加入了随机抽样来扩展用户兴趣，保证推荐的新闻不局限于用户的历史行为。在排序阶段，本系统选用了Naml（Neural news recommendation with attentive multi-view learning）模型做为排序阶段的模型，该模型由微软研究院于2020年发布，并在微软发布的News Recommendation Competition比赛中取得了前三的良好成绩。

定时推荐模块实现是先构建Naml新闻推荐排序模型，然后用大量数据数据训练该模型得到能直接使用的排序模型。然后每三小时使用用户历史行为数据进行基于矩阵分解的协同过滤和利用文章向量相似度先对不同种类文章进行召回，然后将召回的文章放入到之前训练好的Naml新闻推荐排序模型进行排序，最后将最终的不同种类的推荐新闻写入到数据库。后台接受到用户请求阅读的行为从数据库中读取推荐的新闻进行语音转换后发送至客户端。

实时阶段使用事先计算好的文章相似度模型进行召回，并将召回数据放入Naml排序模型进行排序，最后发送至客户端。

基于矩阵分解的协同过滤：

LFM(latent factor model)隐语义模型核心思想是通过隐含特征联系用户和物品，如下图：

表格

中度可信度描述已自动生成

P矩阵是User-LF矩阵，即用户和隐含特征矩阵。LF有三个，表示共总有三个隐含特征。Q矩阵是LF-Item矩阵，即隐含特征和物品的矩。R矩阵是User-Item矩阵，有P\*Q得来。LFM能处理稀疏评分矩阵。

利用矩阵分解技术，将原始User-Item的评分矩阵（稠密/稀疏）分解为P和Q矩阵，然后利用还原出User-Item评分矩阵。整个过程相当于降维处理，其中：

* 矩阵值表示用户1对隐含特征1的权重值
* 矩阵值表示隐含特征1在物品1上的权重值
* 矩阵值就表示预测的用户1对物品1的评分

这样通过矩阵分解的协同过滤就能计算出用户对其他为点击的文章评分从而进行召回。

基于文章向量相似度：

为了计算文章相似度，我们使用word2Vec将文章的单词全部转化成向量，并用文章向量的均值来表示该文章向量，之后将文章先进行局部敏感哈希进行聚类，然后计算每个类里面的欧式距离，选举出与用户看过最相似的k篇文章进行召回。

Word2Vec：

通过一个三层神经网络得出，由约书亚.本吉奥（Yoshua Bengio）提出word2vec模型：

图示

描述已自动生成

Word2vec网络模型

通过窗口输入句子中的连续三个词，w1,w2,w3输入网络中已是初始化的向量，如w1:[0,0,0,0,…..,0]，值的向量长度自定义三个词向量，输入到网络中，目标值为原句子的后面一个w4，通过onehot编码定义网络训练，网络参数更新，自动调整w1,w2,w3的向量值，达到经过最后的softmax(多分类概率)，输出预测概率，与目标值计算损失

Naml模型：

图示

描述已自动生成

Naml模型网络

在该模型中主要分为三个部分，即绿色、红色以及黄色。

1.绿色部分对应的是新闻编码部分：

在该部分，作者利用新闻的各个不同的信息来进行新闻表示，对于Title部分，对于输入word：W1、W2…WM，通过Word Embedding（GloVe）进行词向量表示，对于得到的词向量，为了获取上下文信息，使用CNN卷积层进行处理，得到了包含上下文信息的词向量表示c，最后，为了获取不同词的权重信息，使用一层注意力机制来进行处理，获取到信息最大的词表示，得到  。

在该部分，CNN处理过程如下：



注意力处理表达式如下：





该处使用的注意力和见到的不太一样，其对于词向量c首先利用Vt进行全连接处理，再进行注意力机制的处理。

对于Body使用同样的方法得到。

对于Categories，其利用了category以及subcategory来进行训练，最后得到 以及。

对于得到的新闻各个view的表示，其利用一层注意力机制得到最终的新闻表示：





2.黄色对应的是用户刻画部分：

对用户点击过的所有新闻，利用新闻编码构建所有的新闻表示，接着使用注意力机制来进行用户的刻画，得到用户表示 u 。

表达式与之前无异：





3.红色为点击预测部分：

利用新闻表示以及用户表示，可以计算出新闻与用户的相关性，最终依据得分确定是否为用户推荐该新闻。

4.负采样与模型训练：

通过负采样减少训练规模：



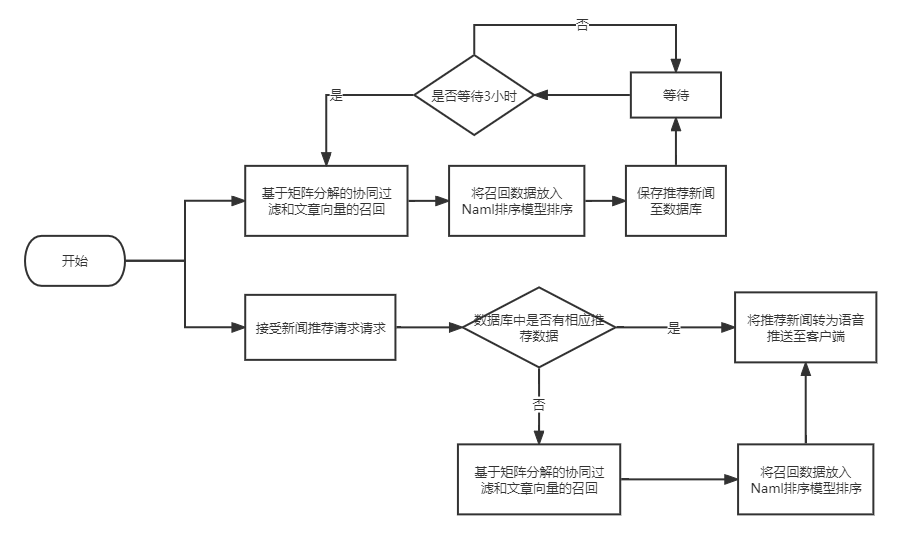
训练LOSS如下：



伪代码：

|  |  |
| --- | --- |
| Function:news recommendation | |
| Input: The kind of news the user wants to see | |
| 1 | while(ture){ //这里循环每3小时进行新闻内容的召回和排序 |
| 2 | DoRecall(); |
| 3 | Naml(); |
| 4 | thread.sleep(10800000) |
| 5 | } |
| 6 | Kind=GetKind(); //获取用户想要看的新闻内容 |
| 7 | AJAX.POST(Kind) //发送到后端 |
| 8 | news=getNewsRecommend("Kind") //获取对应用户和新闻类别的推荐文章 |
| 9 | json=getWord2Audio() //将文章转为音频 |
| 10 | return json //发送至前端 |

### 4.2.7 流程逻辑



### 4.2.8 接口

推荐新闻接口，后端接收前端发起的POST请求，请求数据为对应的注册表单数据。接口返回所需类别的推荐新闻。

### 4.2.9 限制条件

网络连接通畅。

### 4.2.10 测试计划

用户点击进入软件，进入到新闻阅读模块，进行阅读新闻，体验一段时间后，查看推荐新闻是否为自己感兴趣的。管理员获取数据库中用户行文数据，计算出用户的跳过率和阅读率。

## 4.3 新闻推送模块

新闻推送通过用户兴趣给用户智能推送新闻。该模块使用AJAX 来进行数据的异步交互。

### 4.3.1 功能

客户端发送用户识别id信息，通过ajax请求发送给后端，后端传入相应的新闻数据。

### 4.3.2 需求对应

对应需求010

### 4.3.3 性能

该模块属于建立在客户端的实现用户交互的功能性模块，对响应速度的要求较高。

性能；新闻发送至客户端的速度取决于用户的网络性能。

### 4.3.4 输入项

输入项为从客户端读取的用户识别信息数据。

### 4.3.5 输出项

输出项为发送到客户端的新闻数据。

### 4.3.6 设计方法（算法）

预加载思路：

uniapp页面维护一个预加载列表数组

preloadList，[{

audioname,

isready,filePath,

preloadNews:{

newsId:0,

newsTitle:"新闻标题",

newsContent:"新闻内容"

}}

]

在载入时，检查preloadList[0].isready，false，则向播放列表要个新闻id，监听isready直到true，用此新闻id加载第一条新闻，将文件地址返回载入预加载列表中，将isready设为true，开始播放。

下一条播放开始时，设定预加载当前播放的下标，将下一项isready设为false，并加载下一条新闻id+1，加载完成后将其设为true。

上一条播放开始时，设定预加载当前播放的下标，将上一项isready设为false，并加载上一条新闻id+1，加载完成后将其设为true。

preloadNextNews(){//预加载下一条新闻

this.preloadList[(this.preloadListIndex+1)%3].isReady=false

this.preload(this.newsList.newsIdList[this.newsList.newsIndex+1],(this.preloadListIndex+1)%3)

预加载：

下一条播放开始时，设定预加载当前播放的下标，将下一项isready设为false，并加载下一条新闻id+1，加载完成后将其设为true。

上一条播放开始时，设定预加载当前播放的下标，将上一项isready设为false，并加载上一条新闻id+1，加载完成后将其设为true。

getNewsContent(newsId)

.then(news=>{

that.preloadList[preloadListIndex].filePath=src

that.preloadList[preloadListIndex].isReady=true

})

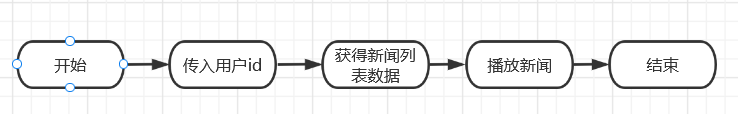
切换时检查：

切换下一条时，检查列表下一项isready，如果为true，则播放下一条新闻；如果为false，则应该监听isready直至其为true，并播放新闻。

切换上一条时，检查列表上一项isready，如果为true，则播放上一条新闻；如果为false，则应该监听isready直至其为true，并播放新闻。

播放结束时，检查列表下一项isready，如果为true，则播放下一条新闻；如果为false，则应该监听isready直至其为true，并播放新闻。(等于切换下一条)

### 4.3.7 流程逻辑



### 4.3.8 接口

前后端新闻推送接口，通过客户端获取用户id数据，返回新闻列表数据。

### 4.3.9 限制条件

（1） 每次推送的新闻不能重复。

### 4.3.10 测试计划

对于该模块的测试预期对不同新闻数量、不同新闻大小建立对应的有效等价

类和无效等价类的测试用例，用于新闻推送模块的“通过测试”和“失败测试”。

接口测试：使用apifox工具对推送新闻的接口进行接口测试。

## 4.4 语音阅读模块

客户端获得新闻语料后，将新闻语料用语音播放。

### 4.4.1 功能

客户端获得新闻语料后，将新闻id发送到后端，后端将新闻文字发送给移动云，后端收到移动云返回的分片base64编码后，将其拼接并保存为wav文件，保存后将文件地址返回给前端。

### 4.4.2 需求对应

对应需求：008

### 4.4.3 性能

该模块属于建立在客户端的实现用户交互的功能性模块，对响应速度的要求较高。

性能；新闻发送至客户端的速度取决于用户的网络性能以及后端处理器的速度和移动云的处理速度。

### 4.4.4 输入项

输入项为从服务端获取的新闻id。

### 4.4.5 输出项

输出项为从客户端发出的新闻语音。

### 4.4.6 设计方法（算法）

客户端获得新闻语料后，将新闻id通过ajax请求发送到后端，后端将新闻文字内容分片发送给移动云，后端收到移动云返回的分片base64编码后，将其转换为字节数组，将字节数组拼接并加上wav文件头后，保存为wav文件，保存后将文件地址返回给前端。

readText(text){

textArr=subMystring()

for(i in textArr){

urlArr.push(textToSound(textArr[i]))

}

result=soundMerge(urlArr)

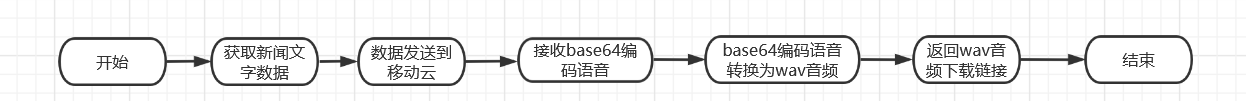
resolve(result)

}

Audio.src= readText(text)

Audio.play()

### 4.4.7 流程逻辑



### 4.4.8 接口

移动云语音听写接口，通过服务端发送新闻语料，从移动云获取阅读语音。

### 4.4.9 限制条件

（1） 每次合成的文本字符数最大为300字符。

### 4.4.10 测试计划

对于该模块的测试预期对不同新闻大小建立对应的有效等价类和无效等价类的测试用例，用于语音阅读模块的“通过测试”和“失败测试”。

接口测试：使用apifox工具对推送新闻的接口进行接口测试。

## 4.5 关键词搜索模块

用户语音输入搜索内容后，为用户推送相应内容。

### 4.5.1 功能

用户语音输入搜索内容后，通过识别关键词，推送相应内容。

### 4.5.2 需求对应

对应需求008

### 4.5.3 性能

该模块属于建立在客户端的实现用户交互的功能性模块，对响应速度的要求较高。

性能；新闻发送至客户端的速度取决于用户的网络性能。

### 4.5.4 输入项

输入项为从客户端获取的语音数据。

### 4.5.5 输出项

输出项为从服务端获取的新闻列表。

### 4.5.6 设计方法（算法）

用户语音唤起语音助手，输入搜索内容，将语音内容传给移动云后，返回相应文字。

首先在本地识别文字中是否包含与指令相关的文字，有对应指令文字则执行指令。

若无指令，判断文字中是否有指定12类新闻之一，有指定12类新闻之一则向后端请求响应的推荐新闻。

若无指定12类新闻之一，则将文字传给移动云进行关键词提取，将返回的关键词返回给服务端，服务端会根据关键词返回新闻列表对象。

若无关键词，则提醒用户语音未识别成功。

Recorder()

.success: res=>{

stt()

.then(text=>{

If(hasCommond(text) process;

Else if(hasMyClass(text))process;

Else if(searchFromKeywords)process;

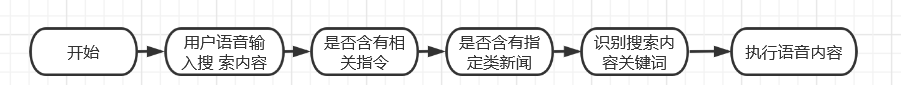
Else{ “no result”}

)

}

)}

### 4.5.7 流程逻辑



### 4.5.8 接口

移动云关键词提取接口，发送实时语音转写分片接口，前后端新闻关键词搜索接口。

### 4.5.9 限制条件

（1） 只能识别出提前已定义的关键词和命令。

（2） 关键词识别可能不是用户想要的结果。

### 4.5.10 测试计划

对于该模块的测试预期对不同语音内容和语音输入时间建立对应的有效等价类和无效等价类的测试用例，用于语音阅读模块的“通过测试”和“失败测试”。

接口测试：使用apifox工具对推送新闻的接口进行接口测试。

## 4.6 拍照识别模块

### 4.6.1 功能

用户拍照物体或文字后，为用户识别物体或文字并播放。

### 4.5.2 需求对应

对应需求：011

### 4.6.3 性能

该模块属于建立在客户端的实现用户交互的功能性模块，对响应速度的要求较高。

性能；新闻发送至客户端的速度取决于用户的网络性能。

### 4.6.4 输入项

输入项为从客户端获取的图片数据。

### 4.6.5 输出项

输出项为从移动云获取的识别数据。

### 4.6.6 设计方法（算法）

使用uni.chooseImage()API调用相机，用户通过点击屏幕照相后，图片数据通过ajax传递给移动云图像识别接口，收到传回的图像识别内容，然后将识别内容通过语音转写接口识别并播报。

伪代码:

pic=openCamera()

if(state==”readText”){

result=identifyText(pic)

readText(result)

}

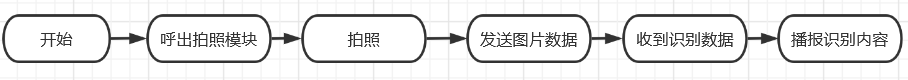
Else{

result=identifyObject(pic)

readText(result)

}

### 4.6.7 流程逻辑



### 4.6.8 接口

通用图像识别接口，图片文字识别接口，移动云语音听写接口

### 4.6.9 限制条件

（1） 只能让用户自己拍照生成图片，不能帮助用户拍照。

### 4.6.10 测试计划

对于该模块的测试预期对不同拍照距离和拍照角度建立对应的有效等价类和无效等价类的测试用例，用于语音阅读模块的“通过测试”和“失败测试”。

接口测试：使用apifox工具对推送新闻的接口进行接口测试。

## 4.7 管理员登录模块

### 4.7.1 功能

用于实现管理员在Web端的登录功能，完成对操作应用系统服务的人员进行身份验证。

### 4.7.2 性能

该模块属于建立在Web端的实现用户交互的功能性模块，对响应速度的要求比较高。将个人信息发送至服务端的速度取决于用户的网络性能，但是对服务端的并发能力也有一些要求。

### 4.7.3 输入项

输入项为管理员向网页中输入的个人信息数据。

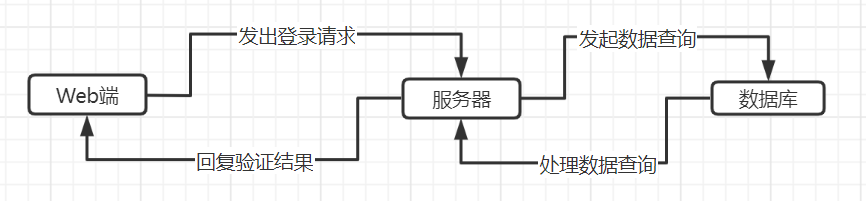
### 4.7.4 输出项

输出项为由服务端发送回本地的验证结果数据。

### 4.7.5 设计方法

该模块首先在展示界面中实现出输入框，包括管理员账号输入框以及管理员密码输入框，用于承载管理员输入的账号以及密码。其次该模块需要在展示界面中实现出按钮组件用于抽象提交动作。之后通过HTTP协议发送到后端处理。服务器将根据请求查询数据库，再根据数据库的回馈生成响应，之后通过HTTP协议向Web回送响应结果。

### 4.7.6 流程逻辑



用户点击账号输入框，输入自己的账号；点击密码输入框，输入账号相关的密码；点击登录按钮，前端发送数据至服务端，由服务端处理身份验证请求，验证之后向前端回馈特定的消息。

### 4.7.7 接口

前后端登录接口，初始时的账号与密码作为服务端身份验证请求的输入项，之后服务端发送给前端的信息作为前端做出下一步动作的输入项。

### 4.7.8 限制条件

限制密码必须由ASCII code组成，长度不能超过20个字符。

### 4.7.9 测试计划

对于该模块的测试预期使用错误的账号、错误的密码、错误的账号与错误的密码进行测试。

## 4.8 新闻爬取模块

### 4.8.1 功能

本模块用于维护系统内新闻数据的实时性，一方面更新系统的新闻数据，另一方面对维持系统的用户体验有重要意义。本模块的主要功能在于周期性拉取新闻数据。

### 4.8.2 性能

该模块的所有功能由服务端实现。由于是周期性任务，对响应速度的要求不高，只需要及时将新近拉取的新闻数据放入数据库中即可。

### 4.8.3 输入项

新闻更新周期，本模块将根据这个周期拉取新闻数据。

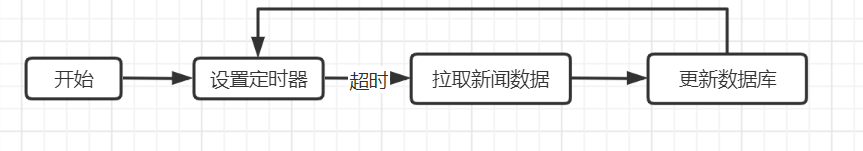
### 4.8.4 输出项

使用拉取到的新闻数据更新数据库。

### 4.8.5 设计方法

制定好数据拉取周期后，将响应的功能实现为一个函数，设置定时器，每次超时则执行这一个函数，同时重设定时器。函数的功能为：从数据源拉取两次间隔内新出现的新闻，之后向数据库发起更新请求。保证数据存放成功。

### 4.8.6 流程逻辑



### 4.8.7 接口

数据源的请求接口，通过访问这个接口获取相关的新闻数据。

### 4.8.8 限制条件

定时器的超时时间不得短于10s，不超过1天。

### 4.8.9 测试

通过将定时器的时间设置为10s，10min，让系统在超时的时候拉取相应的新闻数据，之后检查数据库，合适数据是否完成更新。

## 4.9 用户行文信息统计模块

### 4.9.1 功能

行为信息主要用于构建用户特征，作为模型计算的输入，另外管理员也能在WEB 服务端查看用户行为信息。

### 4.9.2 性能

该模块从用户手机端实时接收数据，并写入数据库，数据交互频率高，能够应对高并发。

### 4.9.3 输入项

输入项为客户端返回的用户浏览新闻数据。

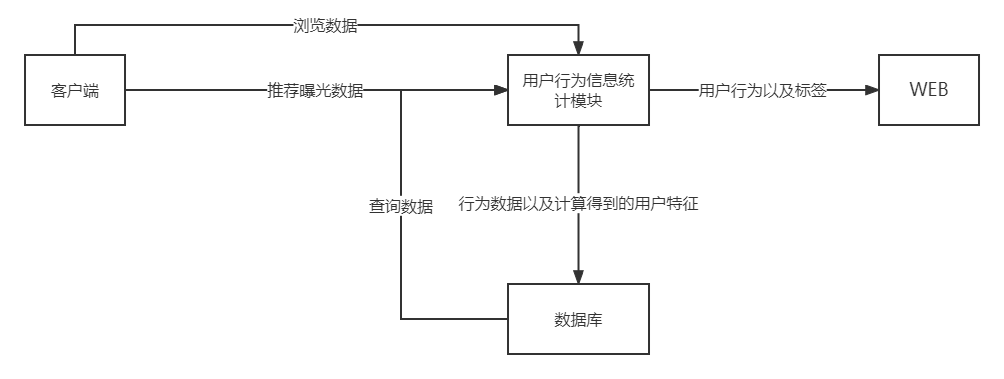
### 4.9.4 输出项

输出项为由浏览数据计算得到的用户特征。

### 4.9.5 设计方法

当有客户端有点击行为，实时传入到该模块，模块更新数据库数据，并计算用户特征。另外，软件给用户推荐了新闻后，需要返回推荐列表中哪些是用户点击了的以及哪些是没点击的。当管理员在WEB端查看时，模块会展示用户的浏览行为，包括浏览新闻以及浏览时长等，还能看到计算出的用户特征标签，比如用户爱看什么类型的新闻。

### 4.9.6 流程逻辑



### 4.9.7 接口

对于客户手机端和WEB服务端都有接口。

### 4.9.8 限制条件

行为数据少的用户的计算特征比较有限。

### 4.9.9 测试

对于高并发情况下的性能进行测试。

## 4.10 新闻管理模块

### 4.10.1 功能

本模块用于管理员对新闻进行增删查改。管理员对新闻的增删保证了数据库内新闻的实时性。管理员除了能查看新闻基本信息，也可以查看新闻的浏览量以及热度值等附加信息。管理员可以修改新闻信息来保证新闻信息的准确性和有效性。

### 4.10.2 性能

管理模块不需要很高的时效性，也不会有高并发的情况，所以对性能要求不高。

### 4.10.3 输入项

添加新闻和修改新闻需要输入新闻基本信息。查询和删除可以通过关键字来搜索，对指定新闻进行操作。

### 4.10.4 输出项

返回相关数据以及操作是否成功的状态。

### 4.10.5 设计方法

新闻管理包含添加、修改、查询和删除四个子功能。在添加和修改部分，web端会检测输入内容是否含有敏感词汇等，保证内容的合法性。而在查询和删除部分，会有输入框用于搜索指定新闻，主要通过计算新闻标题和输入框文字的相似度。另外，该模块直接对数据库进行操作，不需要依赖其他模块。

### 4.10.6 流程逻辑



### 4.10.7 接口

管理员WEB端访问和操作新闻数据库的接口。

### 4.10.8 限制条件

搜索框只能通过标题检索新闻，无法通过新闻内容检索。

### 4.10.9 测试

在增删改操作后，检查数据库是否及时更新。在查询操作后，检查是否检索出所有相关新闻。