# 毕设注意

**计算机毕设基本没有学校对代码进行查重，只会对论文进行查询**，只要把论文写好就可以了。

写论文最简单，最直接的方法，就是去找类似的论文，最新的论文照着写。

加下自己的创新和优化。

# 系统优化

## 系统分析

整体来看是一个偏向百科查询类的系统。

数据是怎么来的呢？

### 角色介绍

系统共设计了三个角色：游客、用户、管理员。

1. 游客：尚未进行注册和登录。具备登录注册、疾病、药品的搜索、查询详情等权限。
2. 用户：除了游客的功能权限外，还具备登录、个人资料的修改、登录密码修改、意见反馈、智能医生咨询等权限。
3. 管理员：除了用户的功能权限外，还具备疾病管理、药品管理、反馈管理等权限。

### 系统遇到的问题：

1. Ai会回复医疗之外的问题。 【还需优化】

解决方案有几种:

1. 关键字过滤（低复杂度）。设置一组与医疗相关的关键词列表，用户输入的问题中必须包含这些关键词，系统才会响应。
2. 意图分类模型（中等复杂度）。需要一定的 NLP 和机器学习知识
3. 主题识别和语义分析（较高复杂度）。利用 NLP 技术进行语义分析，识别用户输入的主题和语境，判断是否与医疗领域相关。
4. 基于知识库的限制（中等复杂度）。

本项目采用第一种。

1. UI问题，游客点击【智能医生】，加个提示，“请登录后重试” 或者 将最好的医疗咨询模块去掉，把“注册账号可获得更多权益”提上来
2. 图片问题，上传图片代码在哪？ -- 图片存在阿里云仓库，需重新上传图片
3. 药品和疾病的数据可以爬虫吗。

### 依赖

依赖涉及广泛的功能，从 Spring Boot 基础框架、数据库操作，到邮件、短信、AI 调用、文件处理等，几乎涵盖了完整的应用开发需求。每个依赖都在各自领域提供了非常成熟的功能，帮助实现智慧医疗系统中的各项业务需求。

### 前端页面解析

└─templates

│ add-illness.html ---添加疾病页面

│ add-medical.html --添加药品

│ all-feedback.html --所有反馈

│ all-illness.html --所有疾病

│ all-medical.html --所有药品

│ doctor.html --医生

│ empty.html --空

│ feedback.html --反馈

│ illness-reviews.html --疾病

│ illness.html

│ index.html

│ medicine.html

│ profile.html

│ search-illness.html

│

├─common

│ common-bar.html --- 定义公共部分，头部条、导航条、底部栏、搜索等

│

└─error

400.html

401.html

404.html

500.html