

**南京中医药大学**

**《Java程序设计》课程设计**

**——大三学年秋季学期**

|  |  |
| --- | --- |
| 项目名称： | 中医处方基础分析系统 |
| 专业/班级： | 17软件 |
| 组 号： | 14 |
| 组 名： | TZ |
| 组 长： | 084317108 陶妍心 |
| 组 员： | 084317108 陶妍心 |
| 指导老师： | 吴丽 |

日期： 2019 年 11 月 27 日

**人工智能与信息技术学院**

# 1. 中医处方基础分析系统规划

系统主要面向三类用户，其中包括中医师个体用户、中医药院校类用户以及中医研究所类用户。

## 1.1 引言

### 1.1.1 项目背景及现状

20世纪以来，随着科技进步和社会生产力的提高，人类得以创造出前所未有的物质财富，生活的方方面面也在逐步迈入信息化大时代。但似乎在中医学方面，信息化趋势仍不乐观。

中医学信息化的关键在于名老中医的经验挖掘，而名老中医的经验挖掘主要体现在其临证医案上，对医案的分析是其经验继承的关键环节。医案中有处方、临床表现、诊断、辨证、立法、评述等若干要素。“处方”是医生辨证论治的落脚点，它虽然是主观思维的产物，但是所有疗效的取得都是源于处方的具体内容，这是没有争议的。从这个角度来说，处方是研究医生经验最客观的数据，而为了挖掘名老中医的经验，需要了解其处方取得疗效的缘由，同时分析其中规律，而现有中医药处方分析软件却存在许多问题：如功能单一，各相关系统间没有紧密联系、交互性差，造成信息重复采集、重复统计；缺乏综合分析能力，可扩充性、灵活性差，缺乏强有力的后续技术支持，无法不断完善软件。

### 1.1.2 术语定义

Java：Java是一种可以撰写跨平台应用软件的面向对象的程序设计语言。Java 技术具有卓越的通用性、高效性、平台移植性和安全性，广泛应用于个人PC、数据中心、游戏控制台、科学超级计算机、移动电话和互联网，同时拥有全球最大的开发者专业社群。在全球云计算和移动互联网的产业环境下，Java更具备了显著优势和广阔前景。

### 1.1.3 项目研究目的

中药处方是中医医疗和中药制剂配制的一项重要书面文件，也是医生为患者预防和治疗疾病开写的用药书面保证。而于中医药而言，处方的存在更为关键，其囊括了开方医师的经验、用药规律，两两药物为何共同存在等诸多信息，但由于过往人们对于中医处方分析方面并不重视，同时中医处方分析的资料也未妥善保管，因此流失了大量的数据资源，直接影响到用药的效果和信息的挖掘。

开发中药处方基础分析系统就能够极大地提高工作效率，让研究人员、医师可以通过该系统快速方便对处方进行研究分析，包括对病史、症状、药物等两两间联系的挖掘，有利于医疗技术水平的不断提高。

本项目旨在于开发一款综合性中药处方基础分析系统，便于用户使用本系统进行中医处方基础信息录入及数据基础分析（包括关联分析、条件概率分析等），同时选择不同可视化形式展现结果，便于用户观察及整理，为中医工作者、研究人员以及学生提供一个触达中医用药规律与中医学奥秘的新通道。

## 1.2 研究项目可行性分析

本项目信息范围主要限于中医处方与用户之间，其中用户囊括了使用者以及管理人员（通过给予不同权限予以区分）；信息来源主要是中医处方（处方信息）、管理人员（可对数据库中药物进行增添等）；人力资源（开发初期）需要部分用户以及管理人员、以及部分内测使用者；所需设备包括手机、PC等。

本项目研究开发所需软件及支持工具资源主要有：Eclipse（前后端）、MySQL（支持数据库）、Microsoft Office Word 2013（报告撰写）、ProcessOn（图类资源绘制）。

本项目预备前期构建雏形，引导部分使用者使用本项目进行简单的中医处方分析，项目内部也会安排管理人员通过数据采集与传输做前期准备，随后开展内测，面向部分自愿参与内测用户开放，广征意见并予以改进；中期预备投放市场进行测试，记录使用数据并予以统计分析计算，找出最佳方案并予以试用；后期预备同时在移动端及服务器端构建平台，扩大受众。

在系统过渡方面，考虑到以下几个问题：

①如果同时运行两个系统，会给客户造成多大的开销；

②如果直接运行新系统，客户面对的风险有多大；

③对新系统试运行时的查错和纠错，以及出现严重错误而导致停止运行时的应急措施；

④客户运行新系统将面临的不利因素有哪些；

⑤人员操作培训方面；

因此，使用不同的系统过渡方案意味着不同的风险，不同的费用及不同的复杂度。目前考虑到三种方案，直接过渡⑴、并行过渡⑵以及阶段过渡⑶三种，联系本项目实际情况以及特点，选用阶段过渡方案。因为玩家社区属于分阶段开发的系统，因此选用阶段过渡计划，在每阶段都设计步骤以及过程完成整个系统过渡（譬如在某阶段对某种功能进行改进等），这样便可减少风险，即使存在失败情况，也只是局部，这对拥有诸多用户的系统（联机）来说更容易接受，带来的损失也更小。

⑴直接过渡

这是一种快速的系统过渡方式，当新系统运行时，立即关闭原来的系统。这种过渡方式非常简单，没有后勤保障的问题，也不要消耗很多资源。同时，它也意味着大风险，目标系统的特性决定了风险的大小。设计者主要要权衡当新系统失败时，系统停止运行或者勉强运行给客户带来的损失有多大。由于这种过渡方式简单而费用低廉，对于可以容忍停机一段时间的系统的实践者，可以采用这种方式。

⑵并行过渡

设计者采用并行过渡方式，让新系统和旧系统在一段时间里同时运行，通过这样的旧系统作为新系统的备份，可以大大降低系统过渡的风险。可是并行过渡显然比直接过渡要消耗更多的资源：现有的硬件资源必须保证能同时跑两套系统，这常常意味着增加服务器和额外的存储空间，需要增加人员来同时使用两套系统，或者增加现有员工的工作量，让他们同时操作两套系统。这种方式同时也增加了管理和后勤保障的复杂度。据统计，并行过渡时期的开销是旧系统单独运行时的 2.5～3 倍。

设计者还会发现有些系统无法使用并行过渡的方式，主要是客户没有足够的资源来维持两个系统同时运行，另外一种情况是新、旧系统差别太大，旧系统的数据无法为新系统采用。当客户无法使用并行过渡，又想尽可能地减少风险，设计者可以使用部分并行过渡的策略，使并行的开销减少到客户能够接受的范围内。

⑶阶段过渡

通常在系统非常复杂、过于庞大以至于无法一次性进行过渡时采用，也适用于分阶段开发的系统。设计者需要设计一系列步骤和过程来完成整个系统的过渡，这种过渡方式和系统的复杂程度相关，随着系统的不同往往有很大的不同。和并行过渡一样，阶段过渡也能够减少风险，显然局部的失败要比全体的失败更容易接受，带来的损失更小。阶段过渡也带来了复杂性，有时候比并行过渡更加复杂。

## 1.3 研究项目所用技术与软件说明

系统环境: macOS Catalina 10.15 && ubuntu 18.04

Java版本: JDK12

开发环境: eclipse 2019-06 (4.12.0)

设计工具: notebility\Procreate

### 1.3.1 JDBC

JDBC，全称Java Database Connectivity ，即Java数据库连接。是[Java语言](https://baike.baidu.com/item/Java%E8%AF%AD%E8%A8%80" \t "_blank)中用来规范客户端程序如何来访问数据库的[应用程序接口](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F%E6%8E%A5%E5%8F%A3/10418844" \t "_blank)，提供了诸如查询和更新数据库中数据的方法。JDBC也是Sun Microsystems的商标。通常说的JDBC是面向关系型数据库的。

JDBC API 既支持数据库访问的两层模型（C/S），同时也支持三层模型（B/S）。在两层模型中，Java applet或应用程序将直接与数据库进行对话。这将需要一个JDBC驱动程序来与所访问的特定数据库管理系统进行 通讯。用户的SQL语句被送往数据库中，而其结果将被送回给用户。数据库可以位于另一台计算机上，用户通过网络连接到上面。这就叫做客户机/服务器配置，其中用户的计算机为客户机，提供数据库的计算机为服务器。网络可以是 Intranet，也可以是 Internet。

### 1.3.2 MAVEN

[Maven](https://baike.baidu.com/item/Maven)项目对象模型(POM)，可以通过一小段描述信息来管理项目的构建，报告和文档的[项目管理工具](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86%E5%B7%A5%E5%85%B7/6854630" \t "_blank)软件。

Maven 除了以程序构建能力为特色之外，还提供高级项目管理工具。由于 Maven 的缺省构建规则有较高的可重用性，所以常常用两三行 Maven 构建脚本就可以构建简单的项目。由于 Maven 的面向项目的方法，许多 Apache Jakarta 项目发文时使用 Maven，而且公司项目采用 Maven 的比例在持续增长。

Maven可用标准化的方式构建项目，可清晰定义项目组成、便于发布项目的信息以及在多个项目中共享JARs。

### 1.3.3 ECLIPSE

Eclipse 是一个开放[源代码](https://baike.baidu.com/item/%E6%BA%90%E4%BB%A3%E7%A0%81/3969" \t "_blank)的、基于[Java](https://baike.baidu.com/item/Java/85979" \t "_blank)的可扩展开发平台，是著名的跨平台的自由[集成开发环境](https://baike.baidu.com/item/%E9%9B%86%E6%88%90%E5%BC%80%E5%8F%91%E7%8E%AF%E5%A2%83" \t "_blank)（IDE）。最初主要用来[Java语言](https://baike.baidu.com/item/Java%E8%AF%AD%E8%A8%80" \t "_blank)开发，通过安装不同的插件Eclipse可以支持不同的计算机语言，比如C++和Python等开发工具。Eclipse的本身只是一个框架平台，但是众多插件的支持使得Eclipse拥有其他功能相对固定的IDE软件很难具有的灵活性。许多软件开发商以Eclipse为框架开发自己的IDE。因此可以说，Eclipse是一个框架和一组服务，用于通过插件组件构建开发环境。此外其附带了一个标准的插件集，包括Java[开发工具](https://baike.baidu.com/item/%E5%BC%80%E5%8F%91%E5%B7%A5%E5%85%B7)（Java Development Kit，[JDK](https://baike.baidu.com/item/JDK/1011" \t "_blank)）。

### 1.3.4 MySQL

MySQL是一种开放源代码的关系型数据库管理系统（RDBMS），使用最常用的数据库管理语言—结构化查询语言（SQL）进行数据库管理。

MySQL是开放源代码的，因此任何人都可以在General Public License的许可下下载并根据个性化的需要对其进行修改。

MySQL因为其速度、可靠性和适应性而备受关注。大多数人都认为在不需要事务化处理的情况下，MySQL是管理内容最好的选择。

### 1.3.5 Navicat

Navicat是一套快速、可靠并价格相宜的数据库管理工具，他专为简化数据库的管理及降低系统管理成本而设。它的设计符合[数据库管理员](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E5%91%98" \t "_blank)、开发人员及中小企业的需要。Navicat是以直觉化的[图形用户界面](https://baike.baidu.com/item/%E5%9B%BE%E5%BD%A2%E7%94%A8%E6%88%B7%E7%95%8C%E9%9D%A2" \t "_blank)而建的，让你可以以安全并且简单的方式创建、组织、访问并共用信息。

[Navicat](https://baike.baidu.com/item/Navicat)提供多达7种语言供客户选择，被公认为全球最受欢迎的数据库前端用户界面工具。它可以用来对本机或远程的MySQL、SQL Server、SQLite、Oracle及PostgreSQL数据库进行管理及开发。

Navicat的功能足以符合专业开发人员的所有需求，而且对[数据库服务器](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93%E6%9C%8D%E5%8A%A1%E5%99%A8" \t "_blank)的新手来说又相当容易学习。有了极完备的图形用户界面(GUI)，Navicat让其可以以安全且简单的方法创建、组织、访问和共享信息。

[Navicat](https://baike.baidu.com/item/Navicat)适用于三种平台：Microsoft Windows、Mac OS X及[Linux](https://baike.baidu.com/item/Linux" \t "_blank)。它可以让用户连接到任何本机或远程服务器、提供一些实用的数据库工具如数据模型、数据传输、[数据同步](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%90%8C%E6%AD%A5" \t "_blank)、结构同步、导入、导出、备份、还原、报表创建工具及计划以协助管理数据。

### 1.3.6 ECharts

ECharts，一个纯 Javascript 的图表库，可以流畅的运行在 PC 和移动设备上，兼容当前绝大部分浏览器，底层依赖轻量级的 Canvas 类库 ZRender，提供直观，生动，可交互，可高度个性化定制的数据可视化图表。

# 2. 中医处方基础分析系统需求分析

①功能性需求

功能性需求即软件必须实现数据录入、成分分析、条件（共现）概率分析、登录注册、关联度分析等基本功能，同时为了向用户提供有用的功能所需执行需要满足“给出提示”动作。

②非功能性需求

不得存在显著BUG、要求前端在Eclipse系统中实现。界面良好，未来可继续做添加操作。

③设计约束

待开发软件须使用MySQL完成数据管理功能，要求其通过不同系统测试，以保证其运行时不会受不同环境的约束，在多种系统下都可运行实现全部功能。

## 2.1 系统目标

中医处方基础分析系统旨在于实现用定性定量的方法来分析医家处方，从而达到从处方中抽取医家辨证论治的思路、用药规律等知识，同时用量化数据的方式将结果形式化的表达出来，便于用户查看，同时设置线程方便用户更明确地查看分析进度等，一定程度上解决了传统中医药分析类系统交互性差的问题。

①成分：根据计算处方组成各药物的作用力度，来统计每一个处方的四气五味、性味归经等重要属性。从而研究处方的在中药学角度的药理性质，也便于用户评判处方综合药性、药味、归经等属性的强度。②条件（共现）概率：将接收到的处方组进行二值化，得到数值化的处方信息；然后依据数理统计的方法，分析这些处方中，每一味中药与同方药物的共现概率。通过共现概率，可以直观地看出在针对某一病症中，中药的配组原则和思路，或可作为新处方开发的重要依据。③关联度分析：改善传统组网边与节点过于繁杂的问题，通过设定阈值，剔除部分关联度偏小的边，从而达到增加关联网络精密度的目的。④社团分析：通过在网络的基础上依据各节点间密度构建社团网络，形成社团内部联系较为紧密，社团外部联系较为稀疏的结构，便于查找药对与分析中药配伍规律。

## 2.2 功能需求

### 2.2.1 数据录入功能

考虑到用户对不同类型，本系统允许用户通过两种方式录入数据，分别为文本录入方式及表格导入方式。对于普通型用户，个人PC中没有规格化的数据集，可通过文本即时输入的方式录入数据，不必通过构建Excel文件，系统化数据后再度导入，方便快捷，同时也便于测试人员测试。对于专业型用户，其个人PC中有着集成化、规格化的数据，可通过直接导入Excel表单的形式录入数据，无需手动输入，节约其工作时间，提高工作效率。

### 2.2.2 数据分析功能

2.2.2.1 成分分析功能

本系统根据用户录入的处方数据，计算处方组成各药物的作用力度，来统计每一个处方的四气五味、性味归经等重要属性。从而研究处方的在中药学角度的药理性质，也便于用户评判处方综合药性、药味、归经等属性的强度。

2.2.2.2 条件（共现）概率分析功能

本系统通过将接收到的处方组进行二值化，得到数值化的处方信息；然后依据数理统计的方法，分析这些处方中，每一味中药与同方药物的共现概率。通过共现概率，可以直观地看出在针对某一病症中，中药的配组原则和思路，或可作为新处方开发的重要依据。

2.2.2.3 关联度分析功能

本系统通过改善传统组网边与节点过于繁杂的问题，同时通过设定阈值，剔除部分关联度偏小的边，从而达到增加关联网络精密度的目的，也实现了与用户交互的目标。

2.2.2.4 社团分析功能

本系统通过在网络的基础上依据各节点间密度构建社团网络，形成社团内部联系较为紧密，社团外部联系较为稀疏的结构，便于查找药对与分析中药配伍规律。

2.2.2.5 数据库查询

数据库采用增量更新模式，将每次用户查询的处方数据更新到数据库中

### 2.2.3 对数据库基本操作

为了保证内部资源的安全性，本系统设置权限允许两类用户以不同方式访问数据库，包括普通用户及管理员用户。管理员用户可实现对数据库的增删改等写任务，普通用户仅能通过读操作访问数据库，一定程度上防止了恶意删除等攻击性行为。

## 2.3 环境需求

### 2.3.1 开发环境要求

配置了Eclipse的普通PC。

### 2.3.2 运行环境要求

配置了Eclipse的普通PC。

# 3. 中医处方基础分析系统概念结构设计

## 3.1 中医处方分析的实体

根据需求分析出的结果，可总结出普通用户、管理员用户、处方三个实体，分别包含ID、姓名、登录名、密码、邮箱、手机号、性别、年龄；ID、姓名、登录名、密码、邮箱、手机号、性别、年龄；ID、药物组。

## 3.2 全局ER图

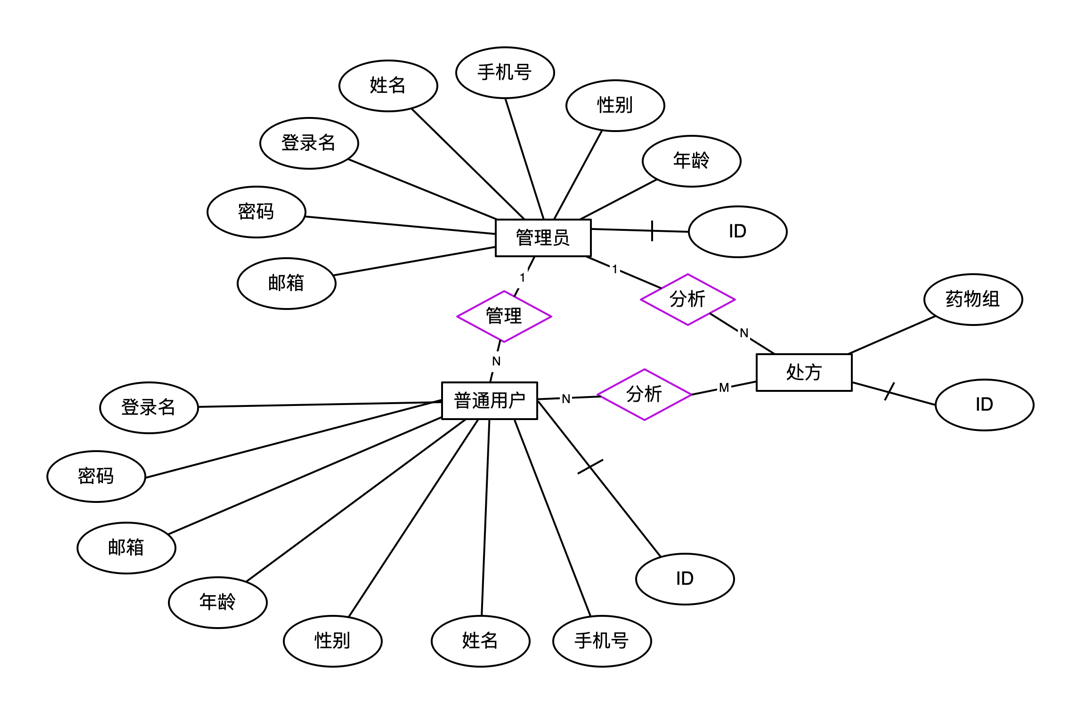


图 1 全局E-R图

# 4. 中医处方基础分析系统总体设计

## 4.1 数据库设计

### 4.1.1 数据字典（DD）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据项名 | 说明 | 类型 | 长度 | 是否主键 | 取值范围 |
| id | 序号 | 自增 | 11 | 是 |  |
| name | 姓名 | varchar | 50 |  |  |
| password | 密码 | varchar | 128 |  |  |
| email | 邮箱 | varchar | 60 |  |  |
| tell | 手机号 | varchar | 11 |  |  |
| status | 状态 | int | 1 |  | 0,1 |
| last\_login\_date | 上次登录时间 | datetime | 6 |  |  |
| login\_name | 登录用户名 | varchar | 255 |  |  |
| create\_date | 创建时间 | datetime |  |  |  |
| sex | 性别 | varchar | 255 |  | M,F |
| age | 年龄 | varchar | 3 |  |  |

表 1 DD

### 4.1.2 数据流图(DFD)



图 2 DFD

## 4.2 功能设计

主要分三部分功能实现，具体如下：

①数据库操作实现

基础增删改查功能，通过SQL语句实现基础操作，同时设置触发器，实现对不规范操作的纠正；分派给不同用户不同权限，分管理员用户以及普通用户；记录每次用户的敏感操作，同时对其每次登录留下记录，设置状态码，实现“假删”操作，以便不规范的删除后得以恢复记录。

②核心功能实现

I 成分分析

将接收到的原始信息转化为0-1矩阵，通过基础的矩阵统计得到处方相关基本信息，譬如方剂总数目、药物种类、不同种类药物数量、药物性味归经等。

II 条件(共现)概率分析

将接收到的处方组进行二值化，得到数值化的处方信息；然后依据数理统计的方法，分析这些处方中，每一味中药与同方药物的共现概率。

III 关联度分析

遍历矩阵，求得两两中药节点间的关联度，并列出关联度表进行可视化展示。

IV 社团分析

在网络形成的基础上，通过不断删除边介数最高的边，循环得到分裂树，最终形成划分开的社团组合，并给出划分完成的社团网络，以便于可视化展示。

V 登录注册

重点在于密码的本地hash加密，以及对sql注入攻击预防。

③交互性实现

I UI排布设计思路

采用多种操作类型的控件，以便与集合发挥各个控件的优势；选择控件JComboBox给予用户选择，增加用户操作简便性，同时可以降低bug出现可能性（由于给出了用户标准化选项，可用于规范用户输入）；联合运用menu和panel控件，减少窗体的数量，使得交互流程更加流畅自然。

II 多线程运用

考虑到数据分析操作耗时较长，为了提高交互性，运用多线程实现进度条，以便于用户随时可查看分析进度，不至于陷入“空等”状态；考虑到页面的整体情况以及运算时间的不确定性，使用点式进度条代替条式进度条。

III 多种输入方式

提供两种输入方式，文本输入和直接导入csv表格，以便于满足不同类型用户需求，文本输入时会自动依据分割字符形成表格。以上两种方式最后都会形成一个预览表格。

IV 阈值输入操作

提供文本框供用户自行设定阈值，再进行关联分析，增强交互性，同时满足用户不同需求。

## 4.3 总体设计

采用MVC架构，总共分为三层。

第一层是models，在初期的规划中，一共设计了两大实体：用户和处方，不过在后来的详细设计中发现，处方的相关操作过于零散，不成体系，所以只保留了用户模型。

第二层是views，定义了的视图界面类，包括登陆界面、主界面、各类分析结果界面等。

第三层是viewmodels，将对views层的一部分复杂操作放到viewmodels层，使得views层的逻辑更加简单，易于维护，并且在一定程度上增强了函数的复用性。同时，viewmodels层作为连通views层和models层的中间件，极大程度上避免了逻辑混乱而造成的损失。

功能模块图如下图。

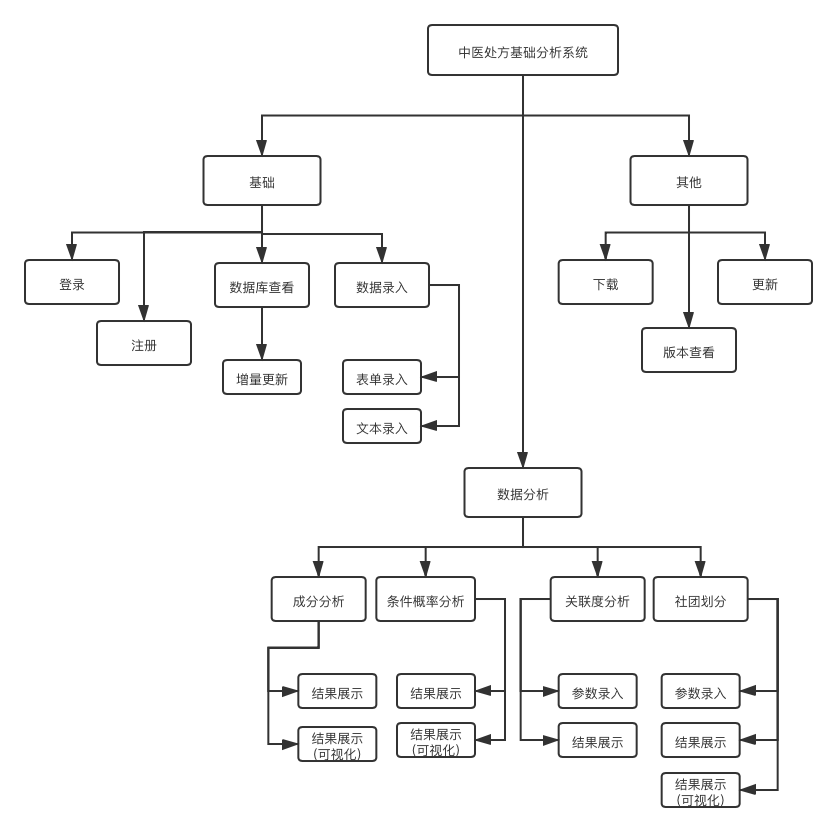


图 3 功能模块图

## 4.4 详细设计

Models:

* User.java
  + 定义了用户模型，变量见数据库设计，方法包括登录校验、更新最后登录时间、注册等
    - 使用date模块完成时间相关功能
    - 使用数据库模块完成成员函数中的登录、注册功能

View:

* Community\_res.java
  + 读取存放在session中的类变量，展示社团分析的结果
    - 使用JSON模块解析ArrayList
* DataImport.java
  + 导入数据，分为文本导入和表单导入两个模块
  + 同时使用jtable将结果显示出来
  + 设置确认和取消两个button
* Login.java
  + 登录界面，提供前往注册界面的入口
* MainFrame.java
  + 主界面，含有Jmenu和5个Jpanel，更具不同的需要展示不同的界面
  + 提供前往数据录入、数据库查询、分析结果查询等功能的入口
  + 使用Java网络编程的相关知识进行与服务器的交互
  + 使用Java多线程的相关知识完成进度条功能
* Query\_sql.java
  + 在每次进行查询之后，数据库中进行增量录入
* Relation.java
  + 读取存放在session中的类变量，展示关联分析的结果
* SignUp.java
  + 注册界面

Viewmodels:

* CsvToJsonUtil.java
  + 使用JSON模块将CSV转化为JSON
* FileUtil.java
  + 提供关于文件操作的一系列函数
* GFG.java
  + 为密码进行hash加密(SHA-256)
* Global\_pool.java
  + 维护一个Session池，内置全局的mutex变量和部分静态变量
* HttpUrlConnectionExample.java
  + 网络编程相关操作的函数集合
* Json2table.java
  + 使用JSON模块将JSON转为JTtable
* MyDocumentLstener.java
  + 自定义的文本监听器
* Mysqlhandler.java
  + 存放了数据库处理相关的函数集合(查询、修改等)

# 5. 中医处方基础分析系统实现

## 5.1 数据库的建立

（user表的建立）

CREATE TABLE `user` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(50) NOT NULL,

`password` varchar(128) NOT NULL,

`email` varchar(60) NOT NULL,

`tel` varchar(11) DEFAULT NULL,

`status` int(1) NOT NULL,

`last\_login\_date` datetime(6) DEFAULT NULL,

`login\_name` varchar(255) NOT NULL,

`create\_date` datetime DEFAULT NULL,

`sex` varchar(255) DEFAULT NULL,

`age` varchar(3) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=20 DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tb\_herb` (

`status` smallint(6) DEFAULT NULL,

`bin\_hid` varchar(32) NOT NULL,

`i\_board\_head` int(11) DEFAULT NULL,

`s\_board\_head` varchar(20) DEFAULT NULL,

`i\_subclass` int(11) DEFAULT NULL,

`s\_subclass` varchar(20) DEFAULT NULL,

`s\_name` varchar(10) NOT NULL,

`s\_abbreviated\_name` varchar(20) DEFAULT NULL COMMENT '首字母缩写',

`s\_usage` varchar(20) DEFAULT NULL COMMENT '用法',

`s\_dosage` varchar(20) DEFAULT NULL COMMENT '用量',

`s\_unit` varchar(10) DEFAULT NULL COMMENT '单位',

`s\_application` varchar(500) DEFAULT NULL COMMENT '应用',

`s\_foundation` varchar(500) DEFAULT NULL COMMENT '基原',

`s\_ancient\_literature` varchar(500) DEFAULT NULL COMMENT '古代文献',

`s\_chemical` varchar(500) DEFAULT NULL COMMENT '化学性质',

`s\_pharmacological\_action` varchar(500) DEFAULT NULL COMMENT '药理作用',

`s\_untoward\_effect` varchar(500) DEFAULT NULL COMMENT '不良反应',

`s\_normal\_remark` varchar(500) DEFAULT NULL COMMENT '普通备注',

`json\_remark` json DEFAULT NULL COMMENT 'Json备注',

`s\_doctor` varchar(20) DEFAULT NULL COMMENT '添加人',

`date` datetime NOT NULL COMMENT '添加日期',

PRIMARY KEY (`bin\_hid`),

UNIQUE KEY `s\_name` (`s\_name`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tb\_prescription` (

`status` smallint(6) DEFAULT NULL,

`bin\_pid` varchar(32) NOT NULL,

`i\_board\_head` int(11) DEFAULT NULL,

`s\_board\_head` varchar(20) DEFAULT NULL,

`i\_subclass` int(11) DEFAULT NULL,

`s\_subclass` varchar(20) DEFAULT NULL,

`s\_name` varchar(20) DEFAULT NULL COMMENT '名字',

`s\_abbreviated\_name` varchar(20) DEFAULT NULL COMMENT '首字母缩写',

`s\_origin` varchar(20) DEFAULT NULL COMMENT '出处',

`s\_usage` varchar(500) DEFAULT NULL COMMENT '用法',

`s\_modified` varchar(500) DEFAULT NULL COMMENT '加减',

`s\_complain` varchar(1500) DEFAULT NULL COMMENT '方解',

`s\_function` varchar(100) DEFAULT NULL COMMENT '功用',

`s\_compatibility` varchar(500) DEFAULT NULL COMMENT '宜忌',

`s\_discussion` varchar(100) DEFAULT NULL COMMENT '各家论述',

`s\_warning` varchar(100) DEFAULT NULL COMMENT '用药禁忌',

`s\_bad\_effect` varchar(500) DEFAULT NULL COMMENT '不良反应',

`s\_clinic` varchar(500) DEFAULT NULL COMMENT '临床应用',

`s\_pharmacological\_action` varchar(500) DEFAULT NULL COMMENT '药理作用',

`s\_prison` varchar(500) DEFAULT NULL COMMENT '毒性实验',

`s\_chemical\_component` varchar(500) DEFAULT NULL COMMENT '化学成分',

`s\_physicochemical\_property` varchar(500) DEFAULT NULL COMMENT '理化性质',

`s\_manufacturer` varchar(50) DEFAULT NULL COMMENT '生产厂家',

`s\_normal\_remark` varchar(1000) DEFAULT NULL COMMENT '普通备注',

`json\_remark` json DEFAULT NULL COMMENT 'Json备注',

`doctor` varchar(20) NOT NULL COMMENT '添加人',

`date` datetime DEFAULT NULL COMMENT '添加日期',

`constitute` varchar(500) DEFAULT NULL COMMENT '组成',

`dose\_list` varchar(50) DEFAULT NULL COMMENT '用量列表',

`s\_indications` varchar(500) DEFAULT NULL COMMENT '主治',

`dosage\_form` varchar(1000) DEFAULT NULL COMMENT '剂型',

`s\_alias` varchar(100) DEFAULT NULL COMMENT '别名',

PRIMARY KEY (`bin\_pid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

## 5.2 数据装载

中药部分的数据使用离线录入，即直接从本地录入数据，数据录入完毕之后锁死，只能由数据库管理员进行修改。

处方部分的数据进行增量更新。用户的每次分析，将符合处方规范的（后续会完善处方检测机制）录入数据库，留待以后进行大数据分析。

用户部分的数据也是增量更新。用户可以自行注册账户并进行使用。

## 5.3 功能实现

### 5.3.1 主要功能对应的关键性代码

将json数据填入jtable，并根据不同的功能生成不同的table

代码如下：

**public** Json2table(String s1,String name) {

JSONObject js = **new** JSONObject();

js = JSON.*parseObject*(s1);

HashSet<String> row =**new** HashSet<String>();

List<String> cols = **new** ArrayList<>(js.keySet());

List<String> rows = **new** ArrayList<>(js.getJSONObject(cols.get(0)).keySet());

a = **new** Object[rows.size()][cols.size()];

**for** (**int** i = 0; i < cols.size(); i++) {

JSONObject eachcol = js.getJSONObject(cols.get(i));

**for** (**int** j = 0; j < rows.size(); j++) {

// System.out.println(eachcol.getString(rows.get(j)));

a[j][i] = eachcol.getString(rows.get(j));

}

}

TableModel model = **new** DefaultTableModel(a,cols.toArray()) {

**public** Class getColumnClass(**int** column) {

Class returnValue;

**if** ((column >= 0) && (column < getColumnCount())) {

returnValue = getValueAt(0, column).getClass();

} **else** {

returnValue = Object.**class**;

}

**return** returnValue;

}

};

tb = **new** JTable(model);

TableRowSorter sorter = **new** TableRowSorter<TableModel>(model);

tb.setRowSorter(sorter);

Container con = getContentPane();

getContentPane().add(**new** JScrollPane(tb),BorderLayout.***CENTER***);

tb.setAutoResizeMode(JTable.***AUTO\_RESIZE\_OFF***);

setTitle(name);

setSize(10000,10000);

setVisible(**true**);

validate();

setDefaultCloseOperation(***HIDE\_ON\_CLOSE***);

}

数据分析部分（二值矩阵、成分分析、条件概率由于分析时间比较短，所以默认开启，关联度和社团分析由于时间较长，交给用户手动开启），开启新线程，并根据不同的任务进度开启进度条计数。

代码如下：

**public** **void** run() {

// **TODO** Auto-generated method stub

**if**(file\_load\_res.isEmpty()) {

inputString += "[";

**for** (**int** i = 0; i < arr.length; i+=2) {

inputString+=("{\"处方名\":\""+arr[i]+"\",\"组成\":\""+arr[i+1]+"\"},");

}

inputString=inputString.substring(0,inputString.length()-1).replace("，", ",")+"]";

System.***out***.println(inputString);

postAndGet(inputString);

} **else** {

postAndGet(file\_load\_res);

}

}

**public** **void** postAndGet(String sj) {

HttpURLConnectionExample poster1 = **new** HttpURLConnectionExample();;

**try** {

AnalizeFrame.*post\_response1* = poster1.*sendPost*(sj,Global\_bool.*url1*);

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

// resultOf01Button.setEnabled(true);

Global\_bool.*laberFlag1* = **true**;

HttpURLConnectionExample poster2 = **new** HttpURLConnectionExample();

**try** {

AnalizeFrame.*post\_response2* = poster2.*sendPost*("",Global\_bool.*url2*);

System.***out***.println("content over");

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

content\_chart.setEnabled(**true**);

content\_table.setEnabled(**true**);

Global\_bool.*labelFlag2* = **true**;

HttpURLConnectionExample poster3 = **new** HttpURLConnectionExample();

**try** {

AnalizeFrame.*post\_response3* = poster3.*sendPost*("",Global\_bool.*url3*);

System.***out***.println("probility over");

} **catch** (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

probility\_chart.setEnabled(**true**);

probility\_table.setEnabled(**true**);

Global\_bool.*labelFlag3* = **true**;

}

### 5.3.2 连接数据库的关键性代码

考虑到代码的复用性，数据库的连接被细分为执行、查询等几个主要函数。

存放在Mysql\_handler类中，这里只展示执行部分的代码。

**public** **static** Boolean sql\_execute(String sql) {

/\*

\* sql执行，传入的sql为标准的sql语句

\* 返回布尔值，表示操作成功或者失败

\*

\* 注意：查询语句与操作语句分用两个函数，因为查询返回的是结果，操作返回的是操作是否完成，注意两者的区别。

\* \*/

Connection con = **null**;

Boolean rs = **true**;

**try** {

Class.*forName*("com.mysql.cj.jdbc.Driver");

con = DriverManager.*getConnection*("jdbc:mysql://39.108.229.166:3306/Java\_sql?useSSL=false","root","root");

PreparedStatement ps = con.prepareStatement(sql);

rs = ps.execute();

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

**return** **false**;

} **catch** (SQLException e) {

e.printStackTrace();

**return** **false**;

} **finally** {

**if** (con!=**null**) {

**try** {

con.close();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

**return** **true**;

}

# 6. 中医处方基础分析系统运行和维护

系统运用Eclipse进行设计，后端数据库支持工具MySQL，效率高，实现了主要功能。

考虑到用户的机器性能参差不齐，所以具体的分析交由服务器来统一实现。

## 6.1主要功能

### 6.1.1条件概率分析

图片包含 文字

描述已自动生成

图 4 条件概率分析

使用JSON模块完成JSON到Array的解析，其结果使用Jtable完成可视化；使用Sort完成Jtable的排序功能。

### 6.1.2成分分析

图片包含 屏幕截图

描述已自动生成

图 5 成分分析

使用JSON模块完成JSON到Array的解析，其结果使用Jtable完成可视化；使用Sort完成Jtable的排序功能。

### 6.1.3主界面

图片包含 屏幕截图

描述已自动生成

图 6 主界面

使用JMenu、JMenuItem完成菜单的设计；使用Jbutton完成按钮设计；使用Jlabel完成文本显示设计。

### 6.1.4数据录入界面

图片包含 屏幕截图

描述已自动生成

图 7 数据录入

使用tabbedPane完成切换格的设计；使用textPane完成文本录入的设计；用文件读写完成表单录入的设计；使用Jtable完成预览功能的设计；使用Jbutton完成交互按钮的设计。

### 6.1.5注册界面

图片包含 屏幕截图

描述已自动生成

图 8 登录注册

使用Jlabel、JtextFeild、JpasswordFeild完成提示输入和输入框设计，使用Jbutton完成交换按钮的设计。

### 6.1.6登录界面

图片包含 屏幕截图

描述已自动生成

图 9 登录界面

使用Jlabel、JtextFeild、JpasswordFeild完成提示输入和输入框设计，使用Jbutton完成交换按钮的设计。

### 6.1.7初始状态的分析结果

由于没有录入数据，所以全都置不可选状态，等到分析完毕后，会恢复为可选状态。

图片包含 屏幕截图

描述已自动生成

图 10 初始状态的分析结果

### 6.1.8社团分析(可视化)

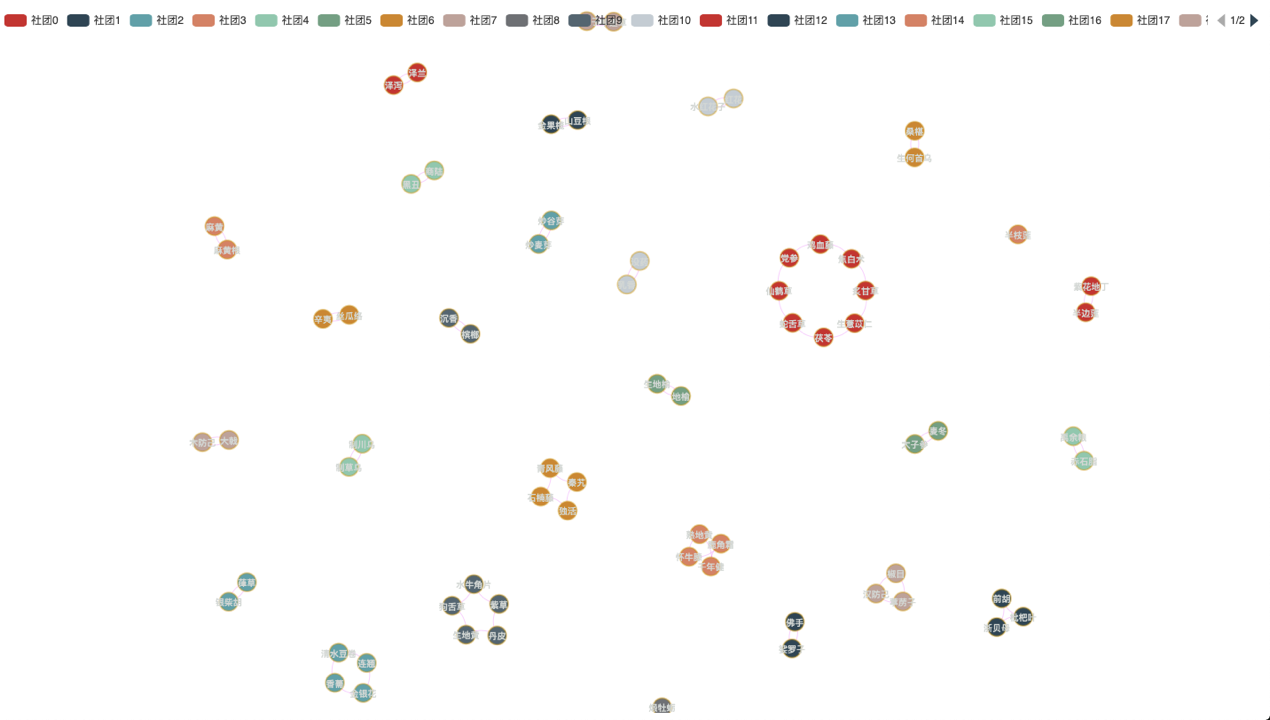


图 11 社团划分(可视化)

使用echarts完成可视化，使用Jinja2完成页面的初始渲染。

### 6.1.9 关联度分析

图片包含 屏幕截图

描述已自动生成

图 12 关联度分析

使用JSON模块完成JSON到Array的解析，其结果使用Jtable完成可视化，使用Sort完成Jtable的排序功能。

### 6.1.10 条件概率分析(可视化)

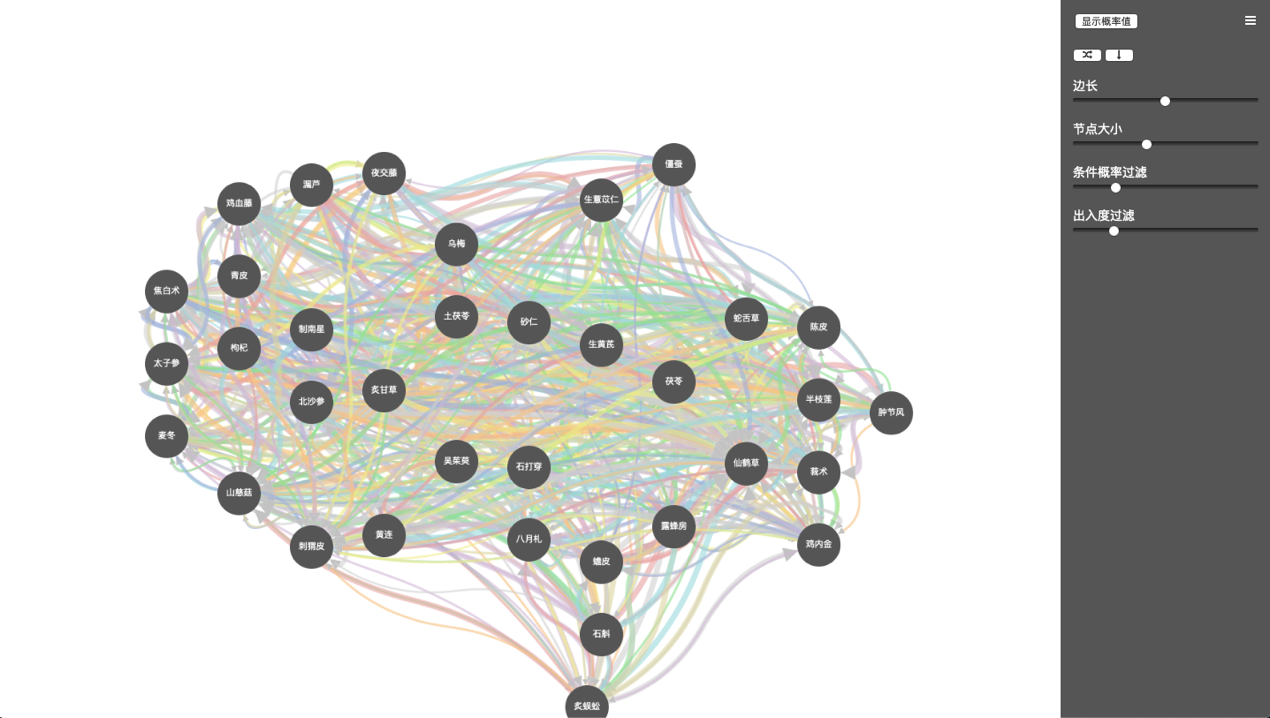


图 13条件概率分析(可视化)

使用echarts完成可视化，使用Jinja2完成页面的初始渲染，使用js完成页面的动态渲染。

### 6.1.11 社团分析(文本)

图片包含 文字, 屏幕截图

描述已自动生成

图 14 社团划分

使用JSON完成回传数据的解析，然后转为String在Jtext中显示。

### 6.1.12 处方成分分析(可视化)

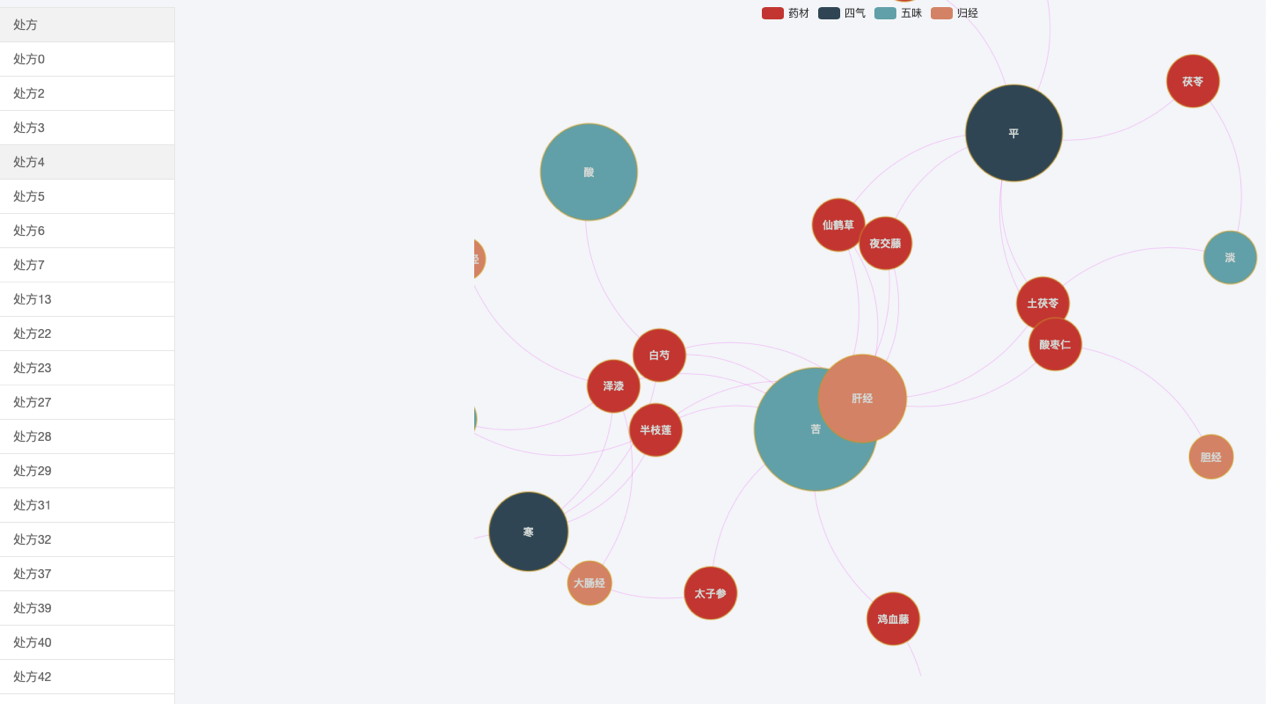


图 15 处方成分分析(可视化)

使用echarts完成可视化，使用Jinja2完成页面的初始渲染，使用js完成页面的动态渲染。

## 6.2 遇到的问题及调试整改

①问题：数据库连接问题

问题描述：在导入java.sql.Connection包并输入连接语句后，提示warning

问题出现原因：mysql5.6 以上的版本要求进行ssl验证

解决方案：在原语句中加上 useSSL=false ，取消验证

问题是否已解决：是

②问题：Java包管理问题

问题描述：开发过程中不可避免地要用到很多的网络上的包，一个一个下载太过凡俗，体验极差

问题解决思路：既然python有pip\conda，那么Java应该也存在自己的包管理工具

解决方案：使用maven

问题是否已解决：是

③问题：多线程问题

问题描述：开发涉及到一个本地和服务器端交互过程，因此想要设计一个进度条，不然交互的那段时间(5s-1min,视用户分析的数据量决定)太过煎熬，且缺乏交互性

问题出现情况：开发过程中考虑到给用户更好体验感

解决方案：引入多线程制作waiting->success进度条

问题是否已解决：是

④问题：json格式数据获取

问题描述：post到网络端需要json格式数据，但生成文本格式为csv

问题出现情况：实现网络端功能时

解决方案：导包直接用csv转json，函数返回的是JSON要求的String类型，再将String转为JSON即可

问题是否已解决：是

⑤问题：中文乱码

问题描述：网络端出现中文乱码情况

解决方案：StringEntity entity = new StringEntity(js,Charset.forName("UTF-8"));

问题是否已解决：是

⑥问题：post请求

问题描述：如何将数据post到服务器端进行分析

问题出现情况：实现网络端功能时

解决方案：创建httpClient、httpPost等实例并进行参数配置

问题是否已解决：是

⑦问题：JTabel列数过多

问题描述：JTabel列数过多，观感体验差

问题出现情况：实现用JTabel可视化展示结果时

解决方案：去掉JTabel自动变换（JTable.AUTO\_RESIZE\_OFF），实现横向滑动

问题是否已解决：是

⑧问题：windowbuilder安装问题

问题描述：手绘编程过于艰难，试图实现可视化编程以方便判断坐标

问题解决思路：在VS stdio上开发c#可以实现可视化，eclipse有对应的GUI开发工具，也就是windowbuilder

解决方案：

S1:安装插件(install new software)(这里注意活用help功能)  
S2:将 <https://download.eclipse.org/windowbuilder/latest/> 复制到work with一栏中  
S3:一路点击next全选安装

问题是否已解决：是

⑨问题：加密方法

问题描述：考虑到安全性，预备采取哈希加密

解决方案：

public static String getSHA(String input) {

MessageDigest md = MessageDigest.getInstance("SHA-256");

byte[] messageDigest = md.digest(input.getBytes());

BigInteger no = new BigInteger(1, messageDigest);

String hashtext = no.toString(16);

while (hashtext.length() < 32) {

hashtext = "0" + hashtext;

}

return hashtext;

}

问题是否已解决：是

⑩问题：string格式化问题

问题描述：数据库中datetime数据要求固定格式：xxxx-xx-xx xx : xx : xx

问题出现情况：需要用到datetime设置时间量时

解决方案：直接用string排出格式xxxx-xx-xx xx : xx : xx

问题是否已解决：是

# 7. 课程设计总结

## 7.1 创新点

首先，本次的选题即具有创新性，因为我选择的是“中药处方基础分析”这个方向，因此需要设计不少数据挖掘类的算法，同时考虑到市面上现存的中医药方面的比较集成的数据挖掘的应用着实不多，因此可以确定我的想法是一个创新点。

其次，本次为了实现社团发现、关联分析等功能的结果可视化，选择了将数据post到集成网络端进行展示，并未将眼界单单局限于GUI界面，因此存在方法上的创新性。

最后，为实现项目，去探索了许多新知识，maven、windowbuilder、post方法、JDBC等等，在课外探索相应知识点，这也是一种创新。

## 7.2 体会

通过本次课程设计，我更加扎实地理解了使用Java进行项目开发相关的知识，在实战中也获得了很大进步，对理论知识的理解也更加深刻。虽然在设计过程中遇到了不少问题，但经过思考，一遍遍的检查与纠错过后最终成功解决了所有出现的问题。但从这点上也可看出，我在理论知识方面尚有所欠缺，且经验还是不足。

本次项目采用了MVC的架构模式，三级划分，使得项目的前端逻辑清晰、脉络完整，无论是后期的维护，功能扩充，还是交付给后来者，都是非常地友好的。

同时，项目采用了前后端分离的体系。在对安全与效率进行了深入的考虑与探究之后，我选择将核心算法的代码放置于服务器，并使用数据处理相对强悍的python去进行分析。而前端则赋予较低的权限，完全作为一个”客户端”来使用。在数据库方面，我分离用户权限，防止串流。这样做的好处是非常明显的。前端、后端、数据库三者有着极低的耦合度，即使作为先锋的前端被攻破了，软件的核心—算法代码和数据库仍然可以维持在一个十分安全的境地。安全方面，其他的hash加密密码等简单环节便不再多做赘述。此外还运用git实现了实时更新等。而前后端分离之后的一个新问题就是交互，这里我采用了RESTFul进行前后端的交互。RESTful的好处，降低了服务的粒度，同时降低了用户与实现的耦合度。

实践出真知，通过亲自动手设计和实现最初的规划，使我课堂内外所掌握的知识不再是纸上谈兵。过而能改，善莫大焉。在课程设计过程中，不断发现错误，不断改正，不断领悟，不断获取。在最终的检测调试环节，本身就是在践行“过而能改，善莫大焉”的知行观。这次课程设计最终顺利完成了，在设计过程中遇到了很多问题，但在老师的指导及不停查阅相关资料、学习新知识的努力之下，终于游逆而解。由此可见，在今后社会的发展和学习实践过程中，需要不懈努力，不能遇到问题就想到要退缩，一定要不厌其烦的发现问题所在，然后一一进行解决，只有这样，才能成功的做成想做的事，才能在今后的道路上劈荆斩棘，而不是知难而退，那样永远不可能收获成功，收获喜悦，也永远不可能得到社会及他人对你的认可！

通过这次课程设计，我掌握了项目设计与软件测试的诸项方法；熟悉了项目总体规划的流程；了解了诸多课堂上未曾学习的知识；扩展了许多不同方面的技能；同时知晓了如何提高系统的性能与交互性等等，掌握了界面设计的方法和技术，通过查询资料，也了解了许多课堂上只是浅略提过的方法原理。我认为，在这学期的实验中，不仅培养了独立思考、动手操作的能力，我在各种其它能力上也都有了提高。更重要的是，在实验课上，我们学会了很多学习的方法。而这是日后最实用的，真的是受益匪浅。要面对社会的挑战，只有不断的学习、实践，再学习、再实践。这对于我的将来也有很大的帮助。以后，不管有多苦，我想我都能变苦为乐，找寻有趣的事情，发现其中珍贵的事情。

回顾起此课程设计，我感慨颇多，Java、Python、MySQL、Linux等等技术第一次实现真正意义上的贯通。从理论到实践，在这段日子里，可以说得是苦多于甜，但是可以学到很多很多的东西，同时不仅可以巩固了以前所学过的知识，而且学到了很多在书本上所没有学到过的知识。通过这次课程设计使我懂得了理论与实际相结合是很重要的，只有理论知识是远远不够的，只有把所学的理论知识与实践相结合起来，从理论中得出结论，才能真正为社会服务，从而提高自己的实际动手能力和独立思考的能力。在设计的过程中遇到问题，可以说得是困难重重，但可喜的是最终都得到了解决。所以总体来说还是十分好的。

## 7.3 尚存不足

想来想去，最主要的问题还是在于如何增强用户观感方面。界面不够友好、交互性不够强、分析速度较为缓慢等等，都会是影响用户观感的问题所在。之后的工作重心，应该还是会放在界面优化以及算法优化上。由于本次课设作品的关键点在于算法方面，因此算法的优化级别实时牵系到分析速度上，算法一旦足够快，用户便也能更快的接收到分析后的结果，因此日后还是会继续优化算法方面。