**面向互联网医疗的用户个性化健康服务推荐系统的研究与实现**

**作者姓名**

**指导教师姓名、职称 教授**

**申请学位类别 工学硕士**

**面向互联网医疗的用户个性化健康服务推荐系统的研究与实现**

**作者姓名：**

**一级学科：**软件工程

**二级学科：**软件工程

**学位类别：**工学硕士

**指导教师姓名、职称：** 教授

**学　　院：**计算机学院

**提交日期：**2017年4月

**西安电子科技大学**

**硕士学位论文**

**学　号**

**密　级　 公开**

**学校代码 10701**

**分类号 TP311**

By

Supervisor: Title: Professor

April 2017

A thesis submitted to

XIDIAN UNIVERSITY

in partial fulfillment of the requirements

for the degree of Master

in Software Engineering

**Research and implementation of personalized health service recommendation system based on E-health**

**西安电子科技大学**

**学位论文独创性（或创新性）声明**

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名： 日 期：

**西安电子科技大学**

**关于论文使用授权的说明**

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，结合学位论文研究成果完成的论文、发明专利等成果，署名单位为西安电子科技大学。

保密的学位论文在 年解密后适用本授权书。

本人签名： 导师签名：

日 期： 日 期：

摘要

个性化医疗推荐服务能够为患者提供有针对性、符合患者个人特性的个性化医疗服务。然而，已有的个性化医疗推荐服务需要成本造价高的特定穿戴设备作为数据采集终端，且仅考虑患者的生理指标，其数据维度单一、无法在复杂多样的医疗资源中为患者提供可靠的个性化推荐结果。因此，本文针对面向患者的个性化医疗推荐服务，以广泛普及的智能手机端作为医疗数据采集平台，结合患者生理指标、地理位置等多维数据，研究用户个性化健康服务推荐系统，以提高医疗服务质量，降低患者的医疗负担。

首先，本文开始介绍了个性化医疗健康服务推荐的研究背景以及研究意义，明确了本文的研究目标以及需要研究的内容，接着对国内外对该领域的研究近况进行了分析，接着对本文中需要应用的相关理论和技术进行了介绍，为文章的后续提供了基础。然后，提出了一种个性化健康服务推荐方法，该方法包括数据的采集、数据的处理以及推荐方法的设计。数据采集对数据采用了JSON的表达方式，具有良好的意图表达性，和传输便利性；数据处理对不同类型的数据的空值以及噪声值的处理方式做了探讨，利用分隔符的方式把数组型的数据整合成了一个数据字段来适应结构化的存储；最后，在推荐方法的设计中，对不同维度的数据进行的聚类处理，采用的推荐算法，各算法的参数的设计都做了详细的阐述。接着，本文对个性化健康服务推荐系统的设计与实现做了介绍。从需求分析的角度出发设计了系统的功能与架构，对系统的核心模块的具体实现做了介绍。为了验证推荐方法的有效性和推荐系统的各个功能的有效性，设计了相关的测试验证并对结果进行了分析。最后，对本文做了总结与展望，对本文工作中的不足做了分析与展望。

为实现研究目标，本文首先对当前国内外个性化医疗健康服务推荐现状进行了深入研究，并以此为基础，结合聚类分析等数据挖掘技术，提出了一种个性化健康服务推荐方法，该方法涉及数据的采集、数据的处理以及推荐方法的设计三个方面。其中，数据采集对数据采用了JSON的表达方式，具有良好的意图表达性和传输便利性；数据处理对不同类型的数据的空值以及噪声值的处理方式做了探讨，利用分隔符的方式把数组型的数据整合成了一个数据字段来适应结构化的存储；推荐方法的设计对不同维度的数据进行的聚类处理，采用的推荐算法，各算法的参数的设计都做了详细的阐述。接着，本文对个性化健康服务推荐系统的设计与实现做了介绍。从需求分析的角度出发设计了系统的功能与架构，对系统的核心模块的具体实现做了介绍。为了验证推荐方法的有效性和推荐系统的各个功能的有效性，设计了相关的测试验证并对结果进行了分析。最后，对本文做了总结与展望，对本文工作中的不足做了分析与展望。

**关键词：** 个性化医疗， 推荐系统， 医疗数据处理

ABSTRACT

Personalized medical referral service can provide personalized medical service for patients with targeted and individual characteristics. However, some personalized recommendation service for the medical cost of the high cost of specific wearable devices as data acquisition terminal, and only considering the physiological data of single dimension patients, they unable to provide reliable personalized for patients in various medical resources in the recommended results. Therefore, according to the personalized recommendation service for medical patients, with widespread intelligent mobile phone terminal of the medical data acquisition platform, combined with the physiological indexes, location of multidimensional data, this paper research the personalized service recommendation system of health, in order to improve the medical service quality, reduce the medical burden of patients.

Firstly, this paper begins with the introduction of personalized health care services to recommend the research background and research significance. Then, the research goal of this paper and the research content, and present research state are analyzed, and then introduces the theory and technology related to the application of this paper, and those provides the basis for the article the subsequent. Then, a personalized health service recommendation method is proposed, which includes data collection, data processing and the design of recommendation method. The data collection uses the JSON expression to the data, has the good intention expression, and the transmission convenience; For different types of data and the null value of noise processing method of data processing is discussed, using the delimiter data integration array into a data field to structured storage; Finally, in the design of the recommended method, the clustering of different dimensions of the data, the use of the proposed algorithm, the parameters of the algorithm design are described in detail. Then, this paper introduces the design and implementation of personalized health service recommendation system. From the point of view of requirement analysis, the function and structure of the system are designed. In order to verify the effectiveness of the proposed method and the effectiveness of the various functions of the recommendation system, the relevant test verification is carried out and the results are analyzed. Finally, the final work of this paper summarizes the work of some of the need to improve and continue in-depth research directions are suggested.

**Keywords**: Personalized medicine, Recommendation system, Medical data processing

插图索引

[图1.1 大数据的个性化医疗服务体系 2](#_Toc479477307)

[图1.2 淘宝推荐 6](#_Toc479477308)

[图1.3 网易云音乐推荐 6](#_Toc479477309)

[图2.1 K均值算法描述 9](#_Toc479477310)

[图2.2 凝聚层次聚类算法描述 10](#_Toc479477311)

[图2.3 整体式混合推荐算法 13](#_Toc479477312)

[图2.4 并行式混合推荐算法 13](#_Toc479477313)

[图2.5 流水线式混合推荐算法 14](#_Toc479477314)

[图2.6 JSON键值对格式 14](#_Toc479477315)

[图2.7 JSON数组格式 14](#_Toc479477316)

[图3.1 数据收集流程 19](#_Toc479477317)

[图3.2 数据分隔符整合工作方式 24](#_Toc479477318)

[图3.3 推荐过程 25](#_Toc479477319)

[图4.1 系统使用者用例图 30](#_Toc479477320)

[图4.2 管理人员用例图 31](#_Toc479477321)

[图4.3 系统架构图 32](#_Toc479477322)

[图4.4 数据清洗算法流程图 34](#_Toc479477323)

[图4.5 数据清洗模块类图 35](#_Toc479477324)

[图4.6 数据整理算法流程图 37](#_Toc479477325)

[图4.7 数据整理模块类图 38](#_Toc479477326)

[图4.8 聚类分析模块交互流程图 40](#_Toc479477327)

[图4.9 聚类分析模块类图 40](#_Toc479477328)

[图4.10 协同过滤模块交互流程图 42](#_Toc479477329)

[图4.11 协同过滤模块类图 42](#_Toc479477330)

[图5.1 基础测试用例 46](#_Toc479477331)

[图5.2数据采集过程 47](#_Toc479477332)

[图5.3 测试用例1执行结果 48](#_Toc479477333)

[图5.4 测试用例2执行结果 48](#_Toc479477334)

[图5.5 测试用例3执行结果 49](#_Toc479477335)

[图5.6 测试用例4执行结果 49](#_Toc479477336)

[图5.7 测试用例5执行结果 50](#_Toc479477337)

[图5.8系统整体响应时间 51](#_Toc479477338)

[图5.9 主要模块工作效率 52](#_Toc479477339)

[图5.10 系统压力响应 53](#_Toc479477340)

表格索引

[表2.1 Mahout主要算法集 15](#_Toc479474939)

[表3.1 医疗大数据特性 17](#_Toc479474940)

[表3.2 收集的数据类型 18](#_Toc479474941)

[表3.3 最终生成JSON数据格式 20](#_Toc479474942)

[表3.4 数据转换映射 23](#_Toc479474943)

[表4.1 数据清洗模块主要函数功能 36](#_Toc479474944)

[表4.2 数据整理模块主要函数功能 39](#_Toc479474945)

[表4.3 聚类分析模块主要函数功能 41](#_Toc479474946)

[表4.4 协同过滤模块主要函数功能 43](#_Toc479474947)

[表5.1 功能测试用例 47](#_Toc479474948)

[表5.2 系统响应时间测试用例 50](#_Toc479474949)

[表5.3 各主要模块工作效率测试用例 51](#_Toc479474950)

[表5.4 系统压力响应时间测试用例 52](#_Toc479474951)

符号对照表

符号 符号名称

缩略语对照表

|  |  |  |
| --- | --- | --- |
| 缩略语 | 英文全称 | 中文对照 |
| HIS | Hospital Information System | 医疗信息系统 |
| EMR | Electronic Medical Record | 电子病历系统 |
| PACS | Picture Archiving and Communication Systems | 医学影像系统 |
| APP | Application | 应用程序 |
| JSON | JavaScript Object Notation | 对象标记 |

目录

[摘要 I](#_Toc479503696)

[ABSTRACT III](#_Toc479503697)

[插图索引 V](#_Toc479503698)

[表格索引 VII](#_Toc479503699)

[符号对照表 IX](#_Toc479503700)

[缩略语对照表 XI](#_Toc479503701)

[第一章 绪论 1](#_Toc479503702)

[1.1 研究背景与意义 1](#_Toc479503703)

[1.2 国内外研究现状 2](#_Toc479503704)

[1.2.1 个性化医疗服务 2](#_Toc479503705)

[1.2.2 推荐系统 4](#_Toc479503706)

[1.3 研究目标与内容 7](#_Toc479503707)

[1.4 论文组织结构 8](#_Toc479503708)

[1.5 本章小结 8](#_Toc479503709)

[第二章 相关理论与技术 9](#_Toc479503710)

[2.1 聚类分析 9](#_Toc479503711)

[2.1.1 基于划分的聚类方法 9](#_Toc479503712)

[2.1.2 基于层次的方法 10](#_Toc479503713)

[2.2 推荐算法 10](#_Toc479503714)

[2.2.1 基于内容的推荐 11](#_Toc479503715)

[2.2.2 协同过滤推荐算法 11](#_Toc479503716)

[2.2.3 基于知识的推荐算法 12](#_Toc479503717)

[2.2.4 混合推荐算法 12](#_Toc479503718)

[2.3 JSON 14](#_Toc479503719)

[2.4 Mahout 15](#_Toc479503720)

[2.5 本章小结 16](#_Toc479503721)

[第三章 个性化健康服务推荐方法的分析与设计 17](#_Toc479503722)

[3.1 医疗数据采集 17](#_Toc479503723)

[3.2 医疗数据处理 21](#_Toc479503724)

[3.2.1 医疗数据清洗 21](#_Toc479503725)

[3.2.2 医疗数据整理及结构化存储 22](#_Toc479503726)

[3.3 推荐方法设计 25](#_Toc479503727)

[3.4 本章小结 27](#_Toc479503728)

[第四章 推荐系统的设计与实现 29](#_Toc479503729)

[4.1 个性化健康服务推荐系统需求分析 29](#_Toc479503730)

[4.1.1 非功能性需求 29](#_Toc479503731)

[4.1.2 功能需求 30](#_Toc479503732)

[4.2 个性化健康服务推荐系统总体设计 31](#_Toc479503733)

[4.2.1 架构设计 32](#_Toc479503734)

[4.3 个性化健康服务推荐系统核心模块具体实现 34](#_Toc479503735)

[4.3.1 数据清洗模块 34](#_Toc479503736)

[4.3.2 数据整理模块 37](#_Toc479503737)

[4.3.3 聚类分析模块 39](#_Toc479503738)

[4.3.4 协同过滤模块 41](#_Toc479503739)

[4.4 本章小结 43](#_Toc479503740)

[第五章 运行与测试 45](#_Toc479503741)

[5.1 测试环境 45](#_Toc479503742)

[5.2 测试过程 46](#_Toc479503743)

[5.2.1 功能测试 46](#_Toc479503744)

[5.2.2 性能测试 50](#_Toc479503745)

[5.3 测试结果分析 53](#_Toc479503746)

[5.4 本章小结 54](#_Toc479503747)

[第六章 总结与展望 55](#_Toc479503748)

[参考文献 57](#_Toc479503749)

[致谢 61](#_Toc479503750)

[作者简介 63](#_Toc479503751)

# 绪论

## 研究背景与意义

随着人们生活质量的提高，医疗服务需求日益增长，大部分病患往往选择具有先进技术的大医院就诊，导致该类医院出现床位短缺、病患爆满等现象，医疗信息资源的短缺，使其难以服务广大人民群众，越来越多的人们出现无处看病的情况，与此同时，地方医院医疗卫生资源不能有效利用，造成了一定程度的医疗资源浪费，加剧了人民群众和医疗卫生单位及从业人员之间的矛盾，医患关系日渐紧张，“看病难，看病贵”问题成为医疗行业必须解决的首要问题。

随着互联网、物联网、云计算等新技术的不断创新与应用，合理的利用信息技术成为了解决以上问题的关键途径之一。

互联网、物联网、云计算等新技术的不断创新与应用，使得利用信息技术解决以上问题成为了可能。因此，在此大环境下，我国发布了一系列的政策文件来推动其发展，其中《关于促进和规范健康医疗大数据应用发展的指导意见》政策提出，标志着互联网那医疗在线服务，如在线咨询服务，网上门诊预约服务，医院检查结果的在线查询服务等应用服务已经形成合理的流程规范形，规范化的管理已经建成，同时对各层级的医疗资源的整合也开始实施，物联网、互联网和医疗应用程序在医疗行业的应用也开始起航[1]。各地方政府与部门也相继的出台了一系列的文件与政策来推动医疗信息化的发展。

由此，逐渐激起了互联网医疗，数字医疗等医疗信息化建设的发展的高潮。

由此，互联网医疗、数字医疗等医疗信息化建设迎来了新的曙光。

借鉴与信息处理技术在零售业、金融业、房地产业、交通、物流等各个行业的快速发展。医疗领域在信息化建设中也备受关注，各个医院开始建设自身的信息化平台、如HIS信息系统、EMR信息系统、PACS信息系统等，在这些系统中，存储了大量的医生数据、患者数据、疾病数据、诊断数据等，这些原始医疗数据对于解决“看病难、看病贵”问题，具有十分重要的意义。一方面，以数据驱动为手段，分析已有的大量医疗数据，挖掘有价值信息，并结合患者的具体病情，能够帮助医生做出更加合理的诊断，在医疗活动各个环节给予患者针对性极高的指导与建议，降低医疗风险，为医生的临床决策提供有力支持[2]。另一方面，通过医疗信息化建设，实现医疗数据的整合与共享，能够避免患者重复检查治疗，节约了医疗成本和资源，提高了医疗资源利用率。

近年来，数据挖掘技术和机器学习技术的蓬勃发展，为各领域的大规模的数据进行分析提供了方法支撑，从而实现了针对个体的个性化服务，如购物平台可以根据用户的浏览和购买记录等数据来给用户推荐一些他感兴趣的商品；音乐网站可以根据用户的听的音乐，收藏的音乐以及他的朋友听的音乐来给用户推荐它感兴趣的音乐。但是，对于医疗领域来说，由于医疗领域的知识特殊性和数据的结构多样性，使得传统方法在医疗领域应用上面对着诸多的问题。因此，如何针对医疗领域的特性，对大量的复杂的多结构的医疗数据进行快速准去的分析与计算，从而为患者提供由针对性的个性化健康服务具有一定的实际意义。

鉴于上述，本文提出一种针对医疗领域独有数据特点的个性化健康服务推荐方法。该（方法）利用协同过滤的思想，从纷繁复杂的医疗数据中提取整合出有效的数据，利用其它（其他）患者的历史数据，寻找与当前使用患者相似的其他患者，并根据这些其他患者的病历信息，地理位置信息等，

寻找与当前患者具有相似症状的其他患者，并根据其病历信息，地理位置信息等，

结合患者自身状况，为当前患者提供个性化的健康服务。为了实现数据的高效计算，本文计划设计并实现了一个高效的，具有一定鲁棒性的个性化健康推荐系统，提高医疗服务质量。

## 国内外研究现状

### 1.2.1 个性化医疗服务

个性化医疗又叫精准医疗，个性化医疗的基本工作方式是结合了个人基因信息、蛋白质信息、代谢信息等相关人体生物学信息，以这些条件为基础，寻找出符合病人自身的最佳的治疗方案，并且该治疗方案可以对该病人起到最佳的治疗效果，且可以把治疗风险降到最低。简单的说，就是把以前根据症状来治疗改变成根据人的特点来治疗，这就是个性化医疗的“个性”含义。这种“个性”，医用在疾病预防、分析、诊断和治疗等各个医疗活动阶段里[3]。近些年来，大数据技术的迅猛发展，使得大数据的分析技术可以有效的应用在个性化医疗领域中。大数据分析技术可以对基因数据、个人基本数据、病历数据等数据进行分析、计算以及可视化，从而形成有针对性的个性化诊疗方案[5]。传统的大数据分析的个性化医疗服务体系架构如图1.1所示：



图1.1 大数据的个性化医疗服务体系

推动个性化医疗的发展具有重大意义，首先它可以帮助提升医疗质量，个性化医疗以患者为中心，结合了患者自身的情况，为患者提供了符合自身特点的健康服务，对于患者来说，可以提高他的健康水平，使得医疗服务更加高效。

个性化医疗的优势在于，它可以利用现有的优秀的信息领域的先进技术，如大数据分析技术，物联网技术和云计算等相关技术，对医疗数据的采集、处理和分析提供了稳定又成熟的保障，对于患者来说，可以获得更加优质的医疗服务。个性化医疗充分的考虑了患者的个体性差异，它可以根据患者的独特的个人情况，为患者提供了符合其自身健康的服务，提高了患者对医疗服务的满意度。个性化医疗可以很好的利用互联网带来的便利，应用互联网和移动APP等工具，可以为患者提供更加全面的医疗服务，并且优化了患者获取医疗健康服务的流程，有效的缓解了我国医疗资源分布不均衡的问题。其次个性化医疗有助于控制医疗费用：个性化医疗有助于实现最佳的治疗策略，通过精准的定位，减少无效的治疗，提高医疗各个阶段的准确性，对患者来说，降低了他的医疗费用。

个性化医疗近些年来在国内外都得到了广泛的重视与发展，在国外，随着个性化医疗研究的日益广泛，各种优秀的生物技术等相关技术的迅速发展，欧美等西方国家已经开始发力于个性化医疗领域，并启动和推出了相关的国家性计划[5]。“精确医疗”的概念是在2011年被美国首次提出的，随后美国在2015年又启动了“精确医疗计划”，并且发布了《精确医疗项目集群——建立21世纪医学研究基金会》白皮书，进一步指明了个性化医疗的发展道路。截止2015年，美国个性化医疗市场价值已经达到了420亿美元，并且仍以每年10%的速率在迅速增长[6]。在日本，日本的主要着力点在于对人类基因的研究，2004年日本建立了全世界最大的人类基因库，为了构建如此庞大的基因库，累计有12个国家，超过40个机构和120名研究人员参与了建设，该基因库收录了约2万多个人类基因，以此为基础，日本建立了结合患者电子病历等信息的基因解析结果数据库，该数据库的建立，标志着日本在个性化医疗领域的初见成果，该数据库的价值在于，可以利用该数据库的数据，可以为每个患者提供最佳的医疗服务体验，同时，它对人类疾病的研究、新型药物的开发和生命科学的探索都具有重大意义[5]。

个性化医疗在我国的发展也同样得到了关注与重视。在国家政策层面上，我国相继发布了一些国家政策来推动个性化医疗的发展，《当前优先发展的高技术产业化重点领域指南》文件中提出，对医疗信息系统的建设，以及对重大疾病的数据库的建设是我国个性化医疗发展的的重点建设方向。在“十三五”规划中，也将启动“精准医疗重点科技研发计划”。2015年我国在全国疾病预防控制工作会议上确定了相关工作重点，其中一点指出，以慢性病为试点，提供全程的管理防治的个性化医疗服务是当前工作的一项重点，以上的国家政策会进一步的推动个性化医疗在我国的发展与进步[7]。

在数据科学的研究领域中，国内外的众多学者也在对个性化医疗的研究发挥着各自的作用。Jonghun Kim等人通过上下文感知技术，建立隐马尔可夫模型来感知上下文，对医生过去所做出的医药知识（包括药剂等）评分预测和推荐的准确性程度为其建立信用档案并根据该档案进行医生之间的医药知识个性化推荐，该方法在一定程度上解决了用户偏好缺失问题，在一定程度上为以后的研究提供了思路[8]。Duan等人利用数据分析和数据挖掘的方法，提出了一种针对临床诊疗方案的个性化推荐方法，该方法对医疗数据提出了一种度量方法，同时，该方法提出了一种树状的数据结构来保存相应的医疗数据，以提高数据的检索效率，最终达到为患者做出个性化医疗方案的推荐目的[9]。Lin等人提出了一个实时的个性化医疗推荐框架，该框架可以有效的结合相关的可穿戴设备以及移动设备，实时的获取患者的相关医疗体征，接着利用数据挖掘等相关方法为患者提供个性化的医疗方案[10]。

### 1.2.2 推荐系统

推荐系统最开始的研究就是为了解决日益增长的信息过载问题[4]，近些年来，随着互联网的超快速发展，各类信息系统的成熟与完善，每天会产生大量的信息，人们难以在这些大量的信息中找到合适自己的信息[11]，而推荐系统的出现，可以有效的帮助人们在纷繁复杂的信息中，筛选出符合自己需求的信息。推荐系统可以利用推荐算法，并结合用户的行为数据、喜欢、习惯等兴趣为其推荐相应的信息，推荐系统以用户为中心，可以满足用户的个性化需求，并且可以根据用户的不断变化的需求，动态的改变推荐的内容，让用户可以得到符合自身爱好、习惯的服务。

推荐系统最开始被定义为使用者把自己的某件事的推荐作为系统的输入，系统会对这些输入进行相应的整合，并把这些推荐直接发送给合适的受众。而随着研究的不断深入与扩大，推荐系统的定义也逐渐有了变化，现在推荐系统被认为是一个可以对个体产生个性化影响或者可以指导个体对其感兴趣的事务作为输出的系统，这样的系统可以应用在一个信息的数量大大的超过了任何人可以掌控的环境里[12]。推荐系统的出现时间并不长，从推荐系统的发展历程来看，可以把推荐系统的发展历程分为三个阶段，第一阶段是推荐系统形成的初期，第二阶段是推荐系统商业应用期，第三阶段是研究爆发期。下面以这三个时期详细阐述推荐系统的发展。

第一阶段是推荐系统形成的初期，这个时期是推荐系统的萌芽时期，这个时期的主要工作就是对推荐系统的探索，在这个时期，基于协同过滤的推荐系统和基于知识的推荐系统首次亮相，在对这两个推荐进行了相关的使用验证后，表明了推荐系统研究的具有一定的可行性与可探索性，这对研究人员是一件欢欣鼓舞的事，使得研究人员对推荐系统的科学研究和商业应用上都充满着积极性。在这一时期，Xerox Palo Alto研究中心设计了名为 Tapestry的推荐系统[13]，该系统主要是帮助用户过滤邮件，解决邮件过载的问题，该系统采用了当时相对新颖的基于其他用户显式反馈（评分和注释）的思想。这也是最早提出“协同过滤”（collaborative filtering，CF） 这一思想的。Tapestry能够实现较好的推荐效果，但它要求使用者具备书写复杂SQL语句的能力，这限制了它的使用和推广。1994 年，第一个可以被称作推荐系统的名为GroupLens的系统[14]被研发了出来。该系统最开始被应用在对文档的过滤与筛选，和 Tapestry不同之处在于，Tapestry只能在某一特定的系统内部进行推荐工作，而GroupLens则可以在不同系统中，通过网络的方式，实现跨网的计算和推荐。1997 年 “推荐系统”（recommender system，RS）一词被Resnick 等人提出[15]，他们认为该词比“协同过滤”更合适来描述推荐技术。自此，“推荐系统”的概念被正式确定，该词被广泛的关注与传播，使得推荐系统开始在信息处理领域成为一个重要方向。

第二阶段是推荐系统的商业应用期，在这一时期，推荐系统首次进去了商业领域，并且取得了不错的效果。在这一时期，研究人员的研究重点主要集中在如何解决超越实验室数据规模的情况下带来的技术难题，研究新的算法来降低计算时间，研究新的系统体系结构来支撑系统的快速响应。在这一阶段，出现了很多优秀的商业应用，最著名的就是电子商务平台亚马逊，顾客浏览了一些商品后，网页中就会出现其他商品的推荐列表。随后，Linden等人公布了在亚马逊电商平台中使用的推荐方法，该方法的优秀之处在于它可以对大规模的评分数据进行高效的处理，并且在计算过后可以实现良好的推荐，这大大提高了亚马逊的知名度和营业额，据统计，推荐系统对亚马逊的营业额的贡献率在20%~30%之间[16]。在国内，也涌现了诸如淘宝一类的电商平台，这类电商平台会根据用户的浏览记录，来适当的为其推荐感兴趣的内容。在一些社交平台上，推荐系统也有着较为突出的应用，比较成功的应用场景就是在FaceBook，Weibo等社交平台上的广告推荐，该类社交平台，会分析用户的喜欢和社会关系，为用户推荐相关的广告。

第三阶段则是研究的爆发期，在这个阶段，新型的算法以及应用被不断的提出。2000年至今，随着应用的不断推广与深入和各个学科内的研究人员的参与，推荐系统得（得去掉）的发展开始逐渐的趋向于稳定和高速。推荐系统在各行各业的应用以及其他相关技术领域的发展与创新，都为推荐系统的研究提供了新的思路和方法。同时相关应用的推广与深入又产生了大量的数据，这些数据又可以推动算法研究方面的进步，这样推荐系统的研究进入了一个良性循环的阶段。2005年基于内容的推荐系统、协同的推荐系统和混合的推荐系统在Adomavicius 等人的综述论文[17]被定义为推荐系统的三个类别，该综述文章中还阐述了推荐系统未来可能的会遇到的瓶颈与研究的方向。这篇文章获得了极高的赞誉，对推荐系统的发展起到了承上启下的作用。而近年来，随着关于推荐系统的会议举办的越来越多，诸如KDD、SIGIR、WWW，ICDM，RecSys等会议，越来越多研究人员开始投身于相关的研究中去，这些会议为广大的研究人员提供了一个国际舞台，这些研究人员可以在这个舞台上尽情的展示推荐系统的新的研究成果。

现如今，无论是在成熟的商业领域，还是在实验研究领域，推荐系统都已经有着广泛的应用与不错的实例。在商业领域中，较为常见的应用领域有电商、电影、音乐等娱乐领域[18]。如在电子商务领域的应用中，淘宝网会根据使用者的历史浏览记录和消费记录，向用户推荐相关产品，如图1.2所示；网易云音乐的则利用了用户的历史行为数据，根据用户的行为数据，对用户进行划分，并结合协同过滤的相关算法，对用户进行音乐的推荐。其推荐结果如图1.3所示。

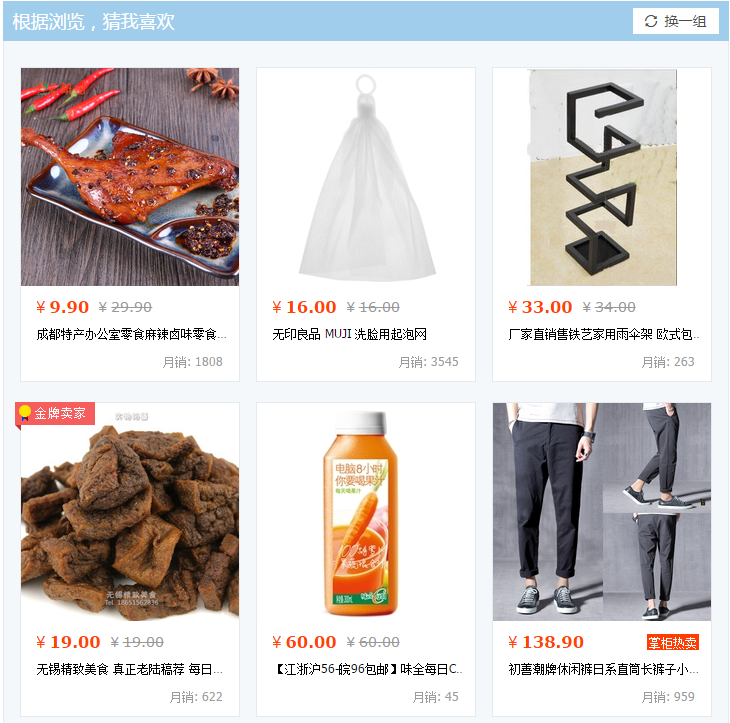


图1.2 淘宝推荐



图1.3 网易云音乐推荐

在实验研究领域，也有着一些重量级的推荐系统，如上海交大Apex实验室开发的推荐系统SVDFeature，它可以很高效的对大矩阵进行计算分解。顾名思义，SVDFeature具有对SVD、SVD++等算法优良的拓展性, SVD系列算法是单模型推荐算法中推荐精度最高的一个[19][40]。SVDFeature的代码实现简单，在大规模的单机矩阵分解运算中，对与计算机的内存消耗较少；台湾国立大学开发的LibMF，它在矩阵分解并行化方面有着突出的贡献，它提出了一种矩阵分解的高效算法FPSGD（Fast Parallel SGD），该方法可以根据计算节点的个数来对评分矩阵进行划分，并为算法的执行分配计算节点[20][40]，该方法对SGD(随机梯度下降)优化方法在并行计算中的锁问题和内存碎片问题有着较好的针对性；明尼苏达大学的GroupLens团队开发的Lenskit[21][40]以及Apache开源的Mahout等。

综上所述，国内外虽然已经有了个性化医疗推荐方面的相关研究，但其医疗数据的采集大都利用较为昂贵的穿戴设备，成本造价较高，且关于面向患者的个性化医疗推荐服务仅考虑了患者的生理指标，数据维度单一，无法在复杂多样的医疗资源中形成可靠的推荐结果。因此，本文以广泛普及的智能手机端作为医疗数据采集平台，降低数据采集成本，并结合患者生理指标、地理位置等多维数据，研究面向互联网医疗的用户个性化健康服务推荐系统，以患者为中心，提供智能、精准的医疗推荐服务，提高医疗服务质量。

## 研究目标与内容

本文结合医疗领域的数据特点，设计并实现了一种以数据为驱动的个性化医疗服务的推荐方法及推荐系统。

推荐方法综合考虑了患者的地理位置信息，兴趣信息和病历信息，首先根据患者的地理位置信息对患者进行一次简单的群组划分，将相同地域的患者划分在同一组里。其次，对同一地域群组里的用户进行兴趣划分，得到相同兴趣的用户群组。最后再根据协同过滤里的基于用户的协同过滤思想，进行相应的推荐。这样经过多次的群组划分与聚类，可以在大量的数据中逐渐缩小范围，使得推荐的准确更加的准确，可以得到更好的推荐效果。

推荐系统在系统层面上分为三层。首先系统会从数据源中接收到数据，接收到的数据可能是多结构多格式的，系统会对这部分数据进行相应的处理，将不同结构和格式的数据统一的进行结构化的表示，并存储于数据库当中，并对缺失的错误的数据进行相应的补全与修正。而后，会利用系统的计算能力，采用个性化医疗服务推荐方法对这些患者进行相应的结果推荐，最后的结果推荐的是医疗服务资源，这里的医疗服务资源主要指的是患者可以使用的医疗服务单位，比如，医生，医院，药品等等。最终再把这些医疗服务资源通过网络协议发送给患者。

## 论文组织结构

文着重针对互联网环境下，个性化健康服务推荐方法和个性化健康服务推荐系统进行研究。全文共六章，其组织结构如下：

第一章从研究背景和研究意义出发，阐述了面向互联网医疗的用户个性化健康服务推荐的研究背景与研究意义，进而阐述了研究目标以及详细的研究内容。然后围绕个性化医疗和推荐系统两个方面阐述了相关领域的研究现状。

第二章主要介绍了本文在研究过程中需要用到的相关的理论与技术，主要包括聚类分析的相关理论，推荐算法的相关内容，JSON数据格式以及Mahout算法库，为后续的章节的阐述提供了理论基础和技术支撑。

第三章主要介绍了对医疗数据的解析过程以及推荐方法，医疗数据的解析主要包括数据的提出了医疗数据清洗的方法，以及相关的结构化整理，并提出了一种依据医疗数据进行的个性化推荐方法。

第四章主要从实际需求出发，搭建了推荐系统的整体框架，并对框架中的主要模块给出了相应的设计与实现。

第五章主要为了验证推荐方法的准确性与有效性，以及系统主要功能的可运行性与完善性，设计了一系列实验测试，并对实验测试结果进行了分析。

第六章对本文工作进行了总结，分析了本文工作的缺点与不足，并对后续工作的一些待完善与继续深入研究的方向进行了展望。

## 本章小结

本章从论文的研究背景和研究意义作为出发点，阐述了面向互联网医疗的用户个性化健康服务推荐的研究意义与研究价值，通过对个性化医疗和推荐系统的国内外现状分析，进一步确立了研究目标以及详细的研究内容。本章作为第一章，是全篇论文的基点，本章的阐述为后续章节的展开进行了铺垫。

# 相关理论与技术

## 聚类分析

聚类分析将数据集合划分成有价值或者有用的新的数据组（簇）。聚类分析往往是数据处理的第一步，即聚类分析的最终结果往往是一些算法或者方法的第一步，它要做的仅仅是数据处理简单但又实用的第一步。现在，各领域都喜欢用聚类分析来解决该领域内的问题。这些领域包括有统计学、心理学、生物学、建筑学等各个工程方向，还有诸如模式识别、信息检索、机器学习和数据挖掘等研究方向[22]。经过聚类分析后得到的簇，每个簇中的对象的相似性都可以保证最大，簇外的对象的相似度都可以保证最小。目前聚类算法经过长时间的发展，已经提出了很多稳定且高效的算法，这些算法大致可以划分成以下几类[23]：（1）基于划分的聚类方法；（2）基于层次的聚类方法；（3）基于密度的聚类方法；（4）基于网格的聚类方法；（5）基于模型的聚类方法；（6）模糊聚类方法；（7）基于图论的聚类方法；（8）基于分型的聚类方法；（9）复杂网络聚类方法；（10）仿生聚类法；（11）核聚类方法等。

### 2.1.1 基于划分的聚类方法

基于划分的聚类方法有很多，其中流传最为广泛的是K均值算法[24]，K均值算法是一种出现时间较早的、发展时间较长的、使用最广泛的聚类算法。K均值算法的思想比较简单易懂，算法第一步先指定K个初始质心，其中K代表的是质心的个数，这个参数代表的意义是希望聚类的簇的数量。在选定K后，会把每个点都分配到离这个点最近的质心，这些被分配点形成一个簇。然后，更新这些簇的质心。重复以上步骤，当所有的簇中的点都不再发生分配，或者说是簇中质心不再发生变化。K均值算法描述如图2.1所示：

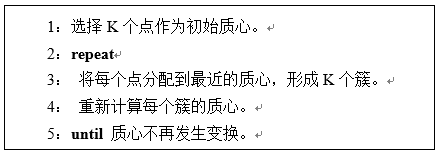


图2.1 K均值算法描述

### 2.1.2 基于层次的方法

基于层次的聚类方法也是一项重要的聚类方法[25]。与K均值一样，基于层次的聚类也是一项发展较早的聚类方法，但是这些方法仍然被广泛的应用。一般来说，层次聚类方法又可以被分为两种，第一种是凝聚的方法，所谓凝聚的方法的含义是是从单个点作为簇的开始，每次合并两个距离最近的簇。这种方法要求定义簇之间的邻近性。第二种是分裂的方法，所谓分裂的方法指的是从所有点出发，首先把所有点集看成一个整体，然后逐渐的分裂成为一个个单独的簇，直到不再分裂簇。应用这种方法，需要确定如何分裂，以及分裂的原则步骤是什么。以目前的应用情况来说，凝聚的层次聚类方法是使用最多的方法，也是研究最为广泛的方法。

凝聚的层次聚类技术一般的算法步骤如下：从点集中的个体点作为簇的开始，合并两个距离最近的簇，知道剩下最后一个簇。其算法描述如图2.2所示：

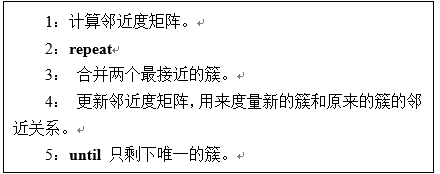


图2.2 凝聚层次聚类算法描述

凝聚的层次聚类的难点之一就是对两个簇的邻近性的定义，现在较为常见的定义两个簇之间的邻近度的方法一般都是基于图的方法，比如有MIN，MAX和组平均。MIN方法又被称为单链方法（single link），它的做法是，遍历两个簇中的所有点，找到两个簇中的最近的两个点，作为簇的距离，也可以理解为在不同点子集中取所有点，最近的两个点即为距离。MAX方法则和MIN方法相反，MAX方法又叫做全链（complete link），全链的具体做法和单链相反，它选择的是两个簇中的最远的两点作为簇的距离，也就是说不同点子集中取所有点，选择两个最远的点作为距离。另一种方法组平均（group average）也是一种基于两个距离的方法。它的做法是两个簇中所有两个点的距离的平均值作为整个簇的距离。除了以上几种方法外，还有Ward等常见的方法[22]。

## 推荐算法

目前，应用较多的几种推荐算法有基于内容的推荐算法、协同过滤推荐算法、混合推荐算法，基于知识的推荐算法和基于人口统计信息的推荐等[26]。

### 2.2.1 基于内容的推荐

基于内容的推荐算法[27]的核心是对物品的描述（既可以是人工描述也可以是自动抽取）和对这些特征的记录，建模与分析。在基于内容的推算法中，最主要是要解决如下三个问题：一、如何持续的获取并更新用户的行为记录。二、如何确定用户对哪类物品感兴趣。三、如何用算法来描述物品的信息，从而减少人工的描述。基于内容的推荐算法主要有以下两大优点，首先，此方法不依赖用户数据，不需要大量的用户数据就可以实现推荐。其次，此方法对计算的需求不大，在得到了物品属性后，就可以立刻为使用者进行推荐。

### 2.2.2 协同过滤推荐算法

协同过滤推荐算法[28]的主要思想是，对已有的大量的用户群体过去的行为进行记录，利用对这些记录的计算与分析来预测当前用户对某些事物的喜爱程度。这种方法可以在用户离线的情况下进行计算，是用户在隐式地与其他人进行相互的协作。目前，此项技术已经得到了广泛的业界应用，尤其是在零售业，社交媒体中应用的较为广泛，在过去的十几年中，人们对此方法进行的广泛的研究，提出了各种算法和技术，这些算法的优点、缺点、时间复杂度和应用的场景都广为人知。基于用户的协同过滤和基于物品的协同过滤是协同过滤推荐算法的两种基本算法。

基于用户的协同过滤[29]算法的基本思想如下：对当前用户的对某些物品的评分矩阵为算法的输入，利用距离度量方法找出与当前用户有着相似兴趣的其他用户，这些其他用户又可以称为邻近用户；然后，根据这些邻近用户对某些项的评分，来预测这些项是否是当前用户所感兴趣的，从而形成推荐。至于如何寻找有相似偏好的其他用户，有很多方法，下面介绍推荐系统中较为通用的方法Pearson相关系数。首先确定一个对物品的评分矩阵R，以用户p和用户q为例，为了描述他们的相似度，定义sim(p,q)，表示为用户p和用户q的相似度，用具体的数学形式化表示如公式3-1所示。其中符号表示的是用户p对物品的平均评分，t表示某一个物品。

 (3-1)

由公式可知，Pearson相关系数的取值区间为[-1,1]，其值越接近1，代表用户之间相似度越大；其值越接近-1，代表用户相似度越小。

基于用户的协同过滤算法已经被应用于许许多多的领域中了，但是，在一些用户量以百万计的应用场景中，基于用户的协同过滤算法还面临着较为严峻的挑战，比如在一些大型电子商务领域的应用中，由于电子商务的使用用户量非常庞大，在确定用户的近邻关系时，需要耗费大量时间在计算上，这时基于用户的协同过滤算法就难以达到实时的推荐效果了。为了解决以上问题，研究人员提出了另一种推荐算法，基于物品的协同过滤[30]。与基于用户的协同过滤不一样，基于物品的协同过滤的主要做法是计算物品之间的相似度，以物品作为度量，而不是以用户之间作为度量，去计算用户的的相似度。因此，为了度量物品之间的是相似度，往往需要一种数学上的表达，经过研究人员的努力，余弦相似度的计算方法被证实是一种效果较为精确的度量方法。余弦相似度用两个n维向量之间的夹角来计算相似度。将两个物品a和b用相对应的评分向量和来表示，其相似度的计算公式3-2如下：

 (3-2)

余弦相似度的计算结果在0到1之间，余弦相似度的值越大，表明两个物品越相似。

### 2.2.3 基于知识的推荐算法

以上的两种方法在一定的方面都有着不足之处，基于内容的推荐只能推荐和曾经喜欢过的物品的相似物品，不能推荐新的物品项。协同过滤推荐算法必须要存在这大量的用户，并且这些用户还得对相应物品进行过打分，存在着冷启动的问题。为了解决上述问题，基于知识的推荐被提了出来。基于知识的推荐既不需要评分数据，也不依赖用户信息，它有两种基本方法，分别是基于约束的推荐和基于实例的推荐[31]，这两种推荐方法的思想比较相似：用户必须指定自己的想要什么，然后系统会根据用户的需求给出相关结果。如果无法给出推荐结果，用户必须重新制定自己想要什么。两种方法之间的差别主要是，前者会特定的计算相似度的公式来寻找符合用户需求的物品，而后者则需要规则集合来完成推荐[32]。

### 2.2.4 混合推荐算法

以上介绍的三种方法都可以很好的完成个性化推荐的处理，但它们在不同的应用场景中的应用效果各有千秋，协同过滤是利用大量的群体用户的评分信息经过计算，来生成推荐列表；基于内容的方法则需要对物品本身的特征进行描述；而基于知识的算法则需要根据知识领域模型进行推理。对于一个应用系统来说，会产生很多的数据输入，比如可能有用户模型和上下文信息、群体数据、产品数据和知识模型等，然而，以上三种方法只能利用部分的系统输入数据，没有一种方法可以完全用到所有这些数据。因此，如果能构建出一种新的推荐系统，它可以结合不同算法和模型的优点，就可以克服它们的不足与缺陷，因此，混合推荐算法被提出来了，混合推荐算法是可以将一种或者多种推荐算法组合在一起的技术[33]。

混合推荐算法一般有三种设计思路[34]，分别是整体式、并行式和流水线式。整体式混合推荐算法是指将各个推荐策略集中到一个推荐模块中。如图2.3所示，整体式混合推荐算法结合了多个推荐算法的共同计算的结果，它只包含一个推荐模块，但它可以通过对各个算法的调整以及权重的设置，使得各个算法可以整合到一起进行工作，这属于对算法的内部结构进行微调，从而可以利用不同类型的输入数据。



图2.3 整体式混合推荐算法

并行式混合推荐算法至少需要实现两种不同的推荐算法，多种算法以并行的方法进行计算，互相之间不干扰；根据输入的数据，不同的推荐算法之间相互独立运行，分别产生最终的推荐列表，如图2.4所示，最后根据不同的策略，这些推荐列表将会被更新为最终的推荐结果。



图2.4 并行式混合推荐算法

流水线式混合推荐算法同样需要至少实现两种不同的推荐算法，如图2.5所示，流水线式是将多个不同的推荐算法按照流水线的工作方式连接起来，上一个算法的输出部分作为下一个算法的输入部分。



图2.5 流水线式混合推荐算法

## JSON

JSON是一种轻量级的数据格式，它具有较好的数据表达及数据传输的能力。JSON可以让使用者们很容易进行数据编写与阅读，同时也方便了计算机进行生成与解析，同时可以有效的提升网络传输效率。JSON一般有两种结构[35]：

(1)“键值”对的集合，在不同的编程语言中，“键值”队可以被理解成不同的表达意思，比如对象，记录，结构体，字典，哈希表，有键列表，关联数组等。“键值”对是一个无序“键”和“值”组成的集合，在JSON中，一个“键值”对以“{”符号开始，以“}”符号结束。每个“键”后面跟一个“：”用来分隔“键”和“值”。图2.6是“键值”对的格式。



图2.6 JSON键值对格式

(2)数组的集合，数组是值的集合，一个JSON数组用“[”符号表示数组的起始，用“]”符号表示数组的结束，具体值之间使用“，”符号进行分隔。值需要使用双引号括起来，它可以是字符串、数值、布尔值、空值、“键值”对或者是数组，同时，这些值也可以被嵌套使用[36]。图2.7是数组的格式。



图2.7 JSON数组格式

## Mahout

Mahout是Apache基金会立项启动的一个项目，它开始是Lucene项目的一个子项目出现的。Lucene项目是一个著名的开源的全文搜索引擎工具包，它在搜索引擎领域的相关探索是超前的，处于领先地位的。由于在全文搜索的过程中，需要对大量的文本进行聚类或者分类，而最常用的聚类或者分类方法往往都是机器学习的方法，这样一来，有些项目的贡献者把精力投入到了对机器学习方法的研究上来，从而逐渐脱离出来形成了Mahout项目[37]。。

Mahout作为一个著名的，被Apache基金会所扶持的项目，它的算法主要归属于机器学习领域的算法，目前来说，Mahout项目主要实现推荐、聚类和分类三个领域的算法。Mahout具有良好的可扩展性，这极大方便的研究人员，研究人员在使用它时，可以针对不同的数据规模，来选择不同的处理模式，当所处理的数据规模远大于单机处理能力时，可以将Mahout和Hadoop组合使用，形成分布式计算框架[39]，并且，这些可扩展的实现都是用Java语言实现的，可以很方便的让开发人员进行编程实现。表2.1是Mahout实现的部分算法。

目前Mahout实现的主要算法集[38]，如下表2.1所示:

表2.1 Mahout主要算法集

|  |  |  |
| --- | --- | --- |
| **算法类** | **算法名** | **中文名** |
| 分类算法 | Logistic Regression | 逻辑回归 |
| Bayesian | 贝叶斯 |
| Support Vector Machines | 支持向量机 |
| Perceptron and Winnow | 感知器算法 |
| Neural Network | 神经网络 |
| Random Forests | 随机森林 |
| Restricted Boltzmann Machines | 有限玻尔兹曼机 |
| 聚类算法 | Canopy Clustering | Canopy聚类 |
| K-Means Clustering | K均值算法 |
| Fuzzy K-Means | 模糊K均值 |
| Expectation Maximization | EM聚类 |
| Mean Shift Clustering | 均值漂移聚类 |
| Hierarchical Clustering | 层次聚类 |
| Dirichlet Process Clustering | 狄里克雷过程聚类 |
| Latent Dirichlet Allocation | LDA聚类 |
| Spectral Clustering | 谱聚类 |
| 关联规则挖掘 | Parallel FP Growth Algorithm | 并行FP Growth算法 |
| 回归 | Locally Weighted Linear Regression | 局部加权线性回归 |
| 降维 | Stochastic Singular ValueDecomposition | 奇异值分解 |
| Principal Components Analysis | 主成分分析 |
| Independent Component Analysis | 独立成分分析 |
| Gaussian Discriminative Analysis | 高斯判别分析 |
| 推荐/协同过滤 | Non-distributed recommenders | Taste(UserCF , ItemCF,  SlopeOne) |
| Distributed Recommenders | ItemCF |
| 向量相似度计算 | RowSimilarityJob | 计算列间相似度 |
| VectorDistanceJob | 计算向量间距离 |

## 本章小结

本章从论文相关领域的理论和技术出发，首先详细介绍了数据处理中的聚类操作，对聚类的基本方法进行了简单的介绍。接下来介绍了常见的几种推荐方法，以及它们各自的特点以及原理，最后介绍了开源的机器学习软件库Mahout，为后续章节的展开提供基本的概念和知识。

# 个性化健康服务推荐方法的分析与设计

本章研究个性化健康服务推荐方法，介绍了对医疗数据的采集，以及医疗数据在运行时和传输时的存储格式；同时介绍了对医疗数据的清洗和整理及结构化的工作；并提出了一种推荐方法，该方法合理的运用聚类算法和协同过滤算法，来为患者生成最终的推荐结果。

## 医疗数据采集

医疗大数据的出现，对传统的医疗数据需求带来了冲击与挑战。如表3.1所示，结合大数据固有的 5V 特性对医疗大数据进行了分析。

表3.1 医疗大数据特性

|  |  |  |
| --- | --- | --- |
|  | **特点说明** | **特点体现** |
| Volume | 量级提升，TB跃升到PB | 单个医疗机构数据存储量平均为1-50TB，部分达到100TB以上 |
| Variety | 数据结构多样、数据内容多样、存储形式多样 | 结构化数据：HIS、EMR等系统的关系数据资源 |
| Value | 大数据包含大价值，需要发掘 | 多元化、精细化医疗决策支持服务 |
| Velocity | 高效性，实时性 | 医疗数据实时检索、查询 |
| Veracity | 数据真实性、数据可靠性 | 实际医疗环节产生的真实、有效、可靠的数据 |

通过对表3.1的分析可知，从传统的医疗信息系统中采集数据，需要对大量的数据进行过滤，且要对不同结构、不同存储形式的数据进行一定的处理，此外还要针对不同的系统采取不同的采集处理。因此，从传统的医疗信息系统中采集数据具有难度大，可执行度不高的特点。为了解决以上困难，本文设计一个基于移动设备的应用，该应用既可以作为医疗数据的采集工具，又可以作为数据分析工具，通过对采集的数据处理，返回最终的推荐结果给用户。为了合理的选择需要的数据，在采集工具中设定了用户输入数据，具体收集的数据类型如表3.2所示：

表3.2 收集的数据类型

|  |  |
| --- | --- |
| **数据类型** | **数据名称** |
| 个人基本数据 | 姓名 |
| 年龄 |
| 性别 |
| 地理位置信息 | 区域 |
| 病历数据 | 就诊过的医院 |
| 就诊过的医生 |
| 所患疾病 |
| 使用过的药物 |
| 兴趣数据 | 关注的医院 |
| 关注的医生 |
| 关注的疾病 |

用户在使用此工具时，在不同的阶段根据自己输入或者由系统为用户自动产生所需数据。个人基本数据主要是由用户根据自身的情况进行输入；地理位置信息由移动设备根据GPS信息自动生成；病历数据需要用户根据自己的既往病史来输入；而兴趣数据则既能通过用户的显性输入同时也可以根据用户的历史行为来产生。

在生成了相应的数据后，需要把这些数据合理高效的采用一定的格式进行组织，以便高效的发送给后台服务器进行保存，在本系统中，采用了JSON数据格式来对这些医疗数据进行表示与传输，JSON数据格式生成方案如图3.1所示：



图3.1 数据收集流程

在数据采集过程中，按顺序对个人基本数据、地理位置数据、病历数据和兴趣数据进行组建，分别的判断该项数据是否齐全，如果齐全，则执行数据组建操作。如果数据不全，执行补全策略，具体来说，补全策略根据数据段的数据类型进行相应的“补零处理”，对于字符串类型以空字符串进行“补零”，对于数值类型则以数字零进行“补零”。对于个人基本数据组建、地理位置数据组建、病历数据组建和兴趣数据组建的方法，具体来说，该方法首先会生成一个名为person的JSON键值对，此键值对采用嵌套的结构，来存储个人基本数据，在此键值对中将定义三个键值对，用于保存当前用户的姓名值、年龄值与性别值。以name作为姓名的键值，实际姓名作为值；以age作为年龄的键值，实际年龄作为值；以sex作为性别的键值，以性别为值，在这里规定男性的值为0，女性的值为1；接着会分别生成两个键值对，一个以latitude为键值，纬度为值和以longitude为键值，经度为值的键值对，分别表示地理位置信息中的纬度和精度。然后会生成一个名为history\_hospital的JSON数组，这个数组里主要保存曾经就诊过的医院的键值对，以name为键值对的键值，以医院名称为值。接着会生成一个名为history\_doctor的JSON数组，这个数组里主要保存曾经就诊过的医院的键值对，以name为键值对的键值，以医生名称为值。接着会生成一个名为history\_disease的JSON数组，这个数组里主要保存从过去到现在所患的疾病的键值对，以name为键值对的键值，以疾病名称为值。接着会生成一个名为history\_drugs的JSON数组，这个数组里主要保存从过去到现在所使用过的药品的键值对，以name为键值对的键值，以药品名称为值。最后分别生成名为interest\_hospital、interest\_doctor和interest\_disease的JSON数组，这个JSON数组里主要采用键值对的方式保存的是用户所感兴趣的医院、医生和疾病数据，同样的，分别以name为键值对的键值，以医院名称、医生名称和疾病名称作为值。最后生成以后的JSON数据如下所示：

表3.3 最终生成JSON数据格式

|  |  |
| --- | --- |
| {“person”:{  “name”:”李云”,“age”:36,“sex”:1},  “latitude”:1,  “longitude”:1  “history\_hospital”:[  {“name”: ”西安唐都医院”},  {“name”: ”西京医院”}  ],  “history\_doctor”:[  {“name”:”范浩”}  ],  “history\_disease”:[  {“name”:”肺炎”}  ], | “history\_drug”:[  {“name”:” 第二代头孢菌素”}  ],  “interest\_hospital”:[  {“name”:”西京医院”}  ],  “interest\_doctor”:[  {“name”:”王德林”}  ],  “interest\_disease”:[  {“name”:”肺炎”}  ]} |

在生成JSON数据后，移动设备将通过HTTP协议将JSON数据传回给服务器，服务器接收到数据后，将进行进一步的处理。

## 医疗数据处理

本节主要介绍医疗数据的处理方法及流程，医疗数据的处理主要包括医疗数据的清洗，整理以及结构化存储。

### 3.2.1 医疗数据清洗

在移动端收集到数据以后，会通过网络传输的方式，把数据发送到服务器端。由于数据的收集采用了用户自主输入和移动客户端自动填充的混合方式，因此在数据的完整性和正确性上存在着一些潜在问题。由于在移动客户端无法完整的校验用户的输入，因此用户难免会因为记错或者是错别字的原因导致输入结果的不合法；移动客户端本身同样也存在着数据出错的隐患，在网络状况不佳时移动客户端的产生的地理位置信息容易出现空值的情况，而不同厂商的移动客户端在字符串的编码上，GPS模块配置上都存在着差异，因此在数据的组建上也容易出现不易发现的错误。因此，在服务器端接受到数据以后，需要对这些数据进行相应的清洗，整理操作，以保证为后续的计算提供正确和可用的数据。对数据的清洗，主要集中在以下几个方面：

1. 由于用户的主观输入，难免会出现用户的拼写错误，记忆错误，项目简写或不填写等异常情况，造成输入的异常值。
2. 由于设备的原因，造成地理位置的数据字段出现空值或者是与实际情况差距较大的值。

对于第1种由用户主观输入造成的异常情况，更进一步又分为两种：一种是空值异常值，另一种是噪声异常值，下面先对这两种异常情况的处理做介绍。

空值异常处理：空值异常有许多的处理办法，可以分为以下几类处理方法：1.忽略该数据；2.删除该属性；3.人工填补空缺值;4.使用系统默认值;5.使用该数据的平均值;6.使用同类的平均值;7.预测可能的值。由上一节可知，空值异常可能出现的数据类型是个人基本信息数据，地理位置数据，病历数据和兴趣数据。由于这四类数据都具有客观性与特定性，因此对这几类的空值处理无法采用人工填补、默认值、平均值和预测值的办法，如果采用这些方法处理空值，极易影响数据的真实可靠性，造成数据的不可靠。因此对于空值的处理全部采用弃用的办法，这也是对数据可靠和正确的最好的保证办法。

噪声异常值处理：噪声异常值产生的原因是由于用户不小心的输入形成的异常数据，主要表现在用户个人信息中的姓名，年龄和性别上，对于这三类数据出现了主观性的输入错误是无法检测的，因此对于这种情况的数据，采取不处理不抛弃的原则。

对于第二种情况，由于设备原因造成的数据异常，主要的存在形式地理位置数据的空值异常。与用户主观输入异常不同的是，设备原因造成的数据异常是有一定的办法对其进行预测补全的。其补全算法的基本思想就是利用其病历数据中的曾经就诊过医院记录，查找数据库中与之相似的曾经就诊过的医院记录的人，并用这些人的地理位置信息的均值对其进行预测补全。具体算法步骤描述如下：

Step1.将当前用户U的数据写入内存。

Step2.if(当前用户U的地理位置数据为控制&当前用户U存在病历数据){

读取当前用户U的病历数据中的曾经就诊过的医院记录，并封装成String类型的数据historyHospital。

查找数据库，对historyHospital进行字符串匹配，提取匹配成功的数据项，并把其依次加入到线性数据结构List中。

}else{

执行Step5。

}

Step3.while(List不为空){

取出List中的数据项的地理位置数据值，进行累加，得到Sum。

}

Step4.将累加后的Sum结果取平均值，写回到当前用户U的地理位置数据中。

Step5.算法结束。

至此，对于数据的空值异常和噪声异常的处理就已经全部完成，下一步要做的就是对数据的合理的整理并结构化存储到数据库当中。

### 3.2.2 医疗数据整理及结构化存储

在上一小节中，对数据的空值异常和噪声异常进行了处理，接下来需要对这些数据进行一定的整理，并将整理后的数据存储到传统的关系型数据库中去，这里采用的是MySQL作为数据库。由上节可知，数据是以JSON格式接收并存储在内存当中的，而对传统的关系型数据库来说，无法直接格式化存储JSON格式的数据，传统的关系型数据库的数据表达方式一般使用的是关系模型，关系模型一般指的是二维表格模型。因此，为了使数据可以符合二维表的存储格式，需要对数据进行进一步的整理。

首先，需要对JSON格式的数据进行转换，使其可以形成二维表的格式。最基本的原则就是把JSON的键作为二维表中的字段，而把JSON中键对应的值作为二维表中属性对应的值。由于JSON数据格式允许嵌套，且每个嵌套分支下的键值对互相不冲突，所以这种情况下，极易出现在不同的键值对嵌套中，出现同一个键值，而在关系型数据库中，任何的字段名称必须是唯一的。因此，为了实现数据可以完整的存储于数据库中，必须对相应字段按一定的规则进行转换映射，具体的转换映射关系如表3.4所示：

表3.4 数据转换映射

|  |  |  |
| --- | --- | --- |
| **JSON键** | **字段映射** | **字段数据类型** |
| person.name | Person\_name | varchar(10) |
| person.age | Person\_age | int |
| person.sex | Person\_sex | bit |
| latitude | Latitude | float |
| longitude | Longitude | float |
| history\_hospital.name | History\_hospital\_name | text |
| history\_doctor.name | History\_doctor\_name | text |
| history\_disease.name | History\_disease\_name | text |
| history\_drug.name | History\_drug\_name | text |
| interest\_hospital.name | Interest\_hospital\_name | text |
| interest\_doctor.name | Interest\_doctor\_name | text |
| interest\_disease.name | Interest\_disease\_name | text |

本映射规则主要是将JSON数据中嵌套结构的重名键名映射成唯一的字段名，这样就符合关系型数据库的存储规则，根据不同字段的使用场景，将其定义成数据库中合适的数据类型。

由于在原本的JSON数组中部分键值对中又嵌套着JSON数组，如history\_hospital、history\_doctor等。这些JSON数组的值往往不止一个，所以需要对这些值进行整合，使之可以存储到数据库中的唯一字段中去，这里有两种方案可以选择，一种是通过设置外键的方式，即将这些数据存放在另一张二维表中，通过设置外键的方式进行数据上的关联；另一种方式采用的是通过分隔符的方式，将多个数据组合在一起，从而可以存入同一字段中去。两种方式各有优缺点：通过设置外键的方式，需要形成一张新的数据表，往往会带来数据的冗余和查询效率的降低，但在逻辑表达上却更清晰也更易于理解；通过分隔符的方式，需要制定创建于解析分隔符的规则，增加了编码的工作量，同时也增加了流程步骤，但通过分隔符的方式，不会产生数据冗余，同时也具有较好的查询效率。本文采用了分隔符的方式，一方面是因为虽然说分隔符的方式一定程度上增加了工作量，但相比于不产生数据冗余和提高查询效率来说，这点代价是可以接受的，并且根据实际情况来说，数据会有一种持续增量的表现，所以采用分隔符的方式也可以很好的适应数据的增量。其工作方式如图3.2所示。



图3.2 数据分隔符整合工作方式

数据输入后，首先会产生分隔符，然后会用分隔符对输入数据进行包装后再存入数据库，同样的，从数据库读取数据后，也会经过分隔符的解析与包装再作为输出。

分隔符的产生主要由分隔符产生器执行，分隔符产生器的工作流程是依次顺序的读入输入数据，每次读入一个数据后，分隔符产生器会在数据的末尾添加一个分隔符，分隔符的字符，则通过配置文件完成，当分隔符产生器读到最后一个输入数据时则不在末尾添加分隔符，到此，分隔符产生器的工作流程结束。

分隔符的解析主要由分隔符解析器执行，分隔符解析器的工作流程和产生器相反，首先会读入带有分隔符的初始字符串数据，依次检索每一个字符，每次遇到分隔符字符则将之前的字符组装成新的字符串作为输出，直到初始字符串中的每一个字符全部检索完。

至此，数据的清洗、数据整理以及结构化的存储已经全部完成，数据已经具备了一定完整性与正确性。

## 推荐方法设计

在经过数据清洗、数据整理以及结构化存储后，获取的数据已经具备一定的完备性与正确性，并且也符合了相关的业务需求。本节主要讨论如何利用这些已有的数据来生成相似用户群组，并且结合相应的推荐算法，实现医疗资源的推荐。在这里，本文将医疗资源被定义为可直接或者间接的为人们所享用的具有医疗特性的资源、如医生等医疗从业人员、医院等医疗机构、药品等医疗物资等。

本文提出的方法可以分三个步骤，首先是根据聚类的方法，将相同地域的用户聚成一簇；第二步是在聚类的基础上，对簇中的具有相同的兴趣的用户再一次进行群组划分，形成新的群组。最后一步，是在新的群组中，根据协同过滤中的基于用户的推荐方法，形成最终的推荐列表。该方法的推荐过程如图3.3所示：



图3.3 推荐过程

**第一步，**用户地理位置信息的聚类。用户的地理位置信息包含经度和纬度两个值，其数据本身就是二维的，因此对于大多数聚类算法，可以不进行相关的预处理操作，可以直接将数据应用到算法中。在聚类算法中，比较适用于地理位置信息聚类的算法有K-means聚类算法和DBSCAN聚类算法。

K-means聚类算法的基本思想是随机确定K个对象作为聚类的中心，计算每一个对象距离K个聚类中心的距离，并把每个对象分发到这K个中心所在的簇中去，当对象被全部分配完毕后，重新计算聚类中心并分配对象，直到聚类中心不再发生变化。在地理位置信息聚类的应用中，由于地理位置信息的获取不具有先知性，用户可能产生的地理位置信息不可预知，因此对于K-means聚类算法中的K的选择具有一定的难度。

DBSCAN聚类算法的基本思想是利用基于密度的概念，将具有高密度的对象划分到同一簇中。与K-means聚类算法相比，它不需要输入划分的个数K，但同样的，它也需要输入另外的参数半径（Eps）和中心邻域内最少点的数量（MinPts），这两个参数一个是用来确定中心的圆形领域的范围，另一个用来判断该点是否应该加入到簇中。而对于Eps参数和MinPts参数的设置，往往是根据人为的经验所设置，设置的好坏直接影响着聚类的效果。

由于我国国情的特殊性，我国的医疗资源呈现出具有区域性的特点，即在不同地域中，总有某一个小的地域具有较为多且优秀的医疗资源，因此，可以把我国的医疗资源分为七个区域，分别为北部、东北、东部、东南、中部、西北和西南。因此，对比K-means聚类算法和DBSCAN聚类算法，K-means聚类算法的参数K值就可以确定为七，而DBSCAN聚类算法的两个参数，却无法从业务需求的角度进行确定。综上所述，从系统的业务需求出发，对于地理位置信息的聚类，本文采用的是K-means聚类算法。

在这里，使用了开源软件库Mahout里面实现的K-means聚类算法。

**第二步，**用户兴趣的分类。在第一步地理位置信息聚类的基础上，根据用户的兴趣对用户进行分类操作，以达到对用户群组更加精细的划分，使群组内的用户的相似度更高。用户兴趣主要包含三个方面，分别是感兴趣的医院、感兴趣的医生和感兴趣的疾病。对于用户感兴趣的这三个方面，可以对这三方面进行模糊化，让这三方面统一作为用户的兴趣，而判断两个用户的兴趣是否相同，只需要判断两个用户的兴趣是否具有较高的相似度，从而可以把问题抽象成相似度计算的问题。

由于一个用户可能有很多兴趣，每个用户的兴趣也不完全一样，因此用户的兴趣往往是多维的，而且这两个用户的兴趣集合并不完全吻合，同时，对于某一个用户来说，其他用户的与当前用户的关系只有相似用户和非相似用户两种，下面针对业务需求提出一种判断用户是否相似的方法，其步骤如下：

Step1.将用户的感兴趣的医院、感兴趣的医生和感兴趣的疾病抽象成三个兴趣集合A，B，C。

Step2.对两个用户的兴趣集合，若或 或 ，则认为这两个用户时相似用户。

在经过上述方法进行用户兴趣分类后，数据准备完毕。

**第三步，**协同过滤形成推荐。经过第一步和第二步的地理位置信息聚类和用户兴趣分类，已经形成了更加精细的用户群体。下一步，将通过协同过滤中的基于用户的协同过滤形成推荐。在由于在医疗服务领域的特殊性，不像其他领域一样，在其他领域中，用户可以对某项商品或者服务进行自己的打分，从而对协同过滤算法来说，就可以利用分数来做相应的推荐。而在医疗服务领域里，由于政策的特殊性以及其发展扔处于萌芽阶段，所以用户难以用分数来衡量医疗资源质量的好与坏。因此，在这里系统统一把用户使用过的医疗资源的打分全部设为5，对用户为使用的医疗资源打分全部设为1。为了形成更加丰富的医疗资源推荐，把用户曾经就诊过的医院、曾经就诊过的医生和曾经使用过的药品抽象成物品，分别对这三类物品使用协同过滤的算法，生成三类推荐列表，并把最终的推荐结果返回给用户。

同样，这里使用了开源软件库Mahout提供的基于用户的协同过滤算法。

## 本章小结

本章是论文工作的主要部分之一，详细的介绍了本文对个性化健康服务推荐方法的研究。从个性化健康服务推荐方法的三个步骤做了详细的介绍。首先介绍了数据的采集方法，以及对数据的表达及传输的格式。然后介绍了数据的处理中的清洗数据中的空值及噪声的方法，和对数据的整合以及结构化的方法进行了介绍。最后，对推荐方法中的流程设计做了介绍。本章是个性化健康服务推荐系统的核心部分，对个性化健康服务推荐系统起着重要作用。

# 推荐系统的设计与实现

上一章着重介绍了个性化健康服务推荐方法的过程与方法，结合实际应用场景介绍了个性化健康服务推荐方法的过程，包括其数据采集过程、数据处理过程和推荐方法。本章开始介绍个性化健康服务推荐系统的设计与实现，会在已有的推荐系统上进行扩充和整合，使其达到医疗领域的使用要求。本章首先会对个性化健康服务推荐系统进行需求分析，接着会给出系统的总体设计，最后会介绍系统的核心模块的具体实现过程。

## 个性化健康服务推荐系统需求分析

需求分析是整个软件工程里的最重要的一环，需求分析的质量的高低，直接影响着软件的质量的高低，本节主要对个性化健康服务推荐系统进行需求分析，将从系统非功能性需求和功能性需求来进行描述。

### 4.1.1 非功能性需求

对于一个软件系统的开发来说，首先要确定这个软件的主要使用人群以及他们的使用场景，具体到本文所探讨的个性化健康服务推荐系统来说，使用本系统的人群主要是患者群体以及关注身体健康的一部分人群，这部分人群主要有以下几个特点：

1. 信息需求量大。这部分人群的目的性往往比较明确，希望获取较为权威的且有效的医疗资源。
2. 年龄分布较为集中。大部分年龄在40岁以下的人，对自身的健康的关注度往往不高，而对健康服务关注较多的人群，年龄段大多集中在40岁以上。
3. 实时性要求高。这类用户往往需要快速的获取信息，而不应该具有长时间的等待。

因此，对于这部人群的特点，首先系统要满足性能上的需求，这部分需求包括具有快速响应的能力，在用户进行相关的输入后，该系统可以在可接受的时间范围内对用户的需求做出响应，系统对于大规模的访问需求具有一定的处理能力，当遇到访问峰值的时候，系统在保障响应的同时，对于系统资源的消耗，应有一定的应对策略。其二，系统应该可以有一定的可靠性。当遇到大量的访问请求，以及计算请求时，如果系统的性能上已经不能及时响应了，那么系统必须具有相应的任务调整策略，执行滞后响应，以此来缓解请求压力过大，防止系统崩溃。

### 4.1.2 功能需求

图4.1是患者群体以及关注身体健康的一部分人群的用例图。用户这类用户属于系统的最外层的使用用户，是系统的输入者和输出者，这类用户往往接受和使用的功能是经过系统底层功能层层包装后的功能，所以根据用户的实际使用需求，可以这类用户的功能划分为输入模块和输出模块两部分。

输入模块：主要包含数据组建、数据传输和数据解析三个模块，这三个模块即使用户必要的模块也是整个系统在数据预处理方面的核心模块，这三个模块的功能运行，直接关系到系统对数据处理和计算的质量。

输出模块：主要是最后推荐结果的输出，主要包含医院资源推荐、医生资源推荐和药品资源推荐模块。这三个模块不仅仅是作为给用户的输出，同时也是系统经过一系列的处理与运算所形成的最终结果，这三个模块是系统最终计算结果的展现。



图4.1 系统使用者用例图

除了一般的使用人群外，系统还需要有系统级的管理人员，图4.2是系统管理人员的用例图。管理人员的主要作用是对系统的维护以及监控，这类用户的主功能要包括配置管理、系统监控和日志管理。

配置管理：配置管理主要是用相关的配置文件来对系统中的各个功能的运行规则进行规定，包含对数据库连接池的配置；计算功能模块中的线程池的配置；对任务调度方式的配置；以及对数据输入输出方式的配置。

系统监控：系统监控主要是对系统运行时的状态进行监控，主要包含对系统日志的监控、对系统网络状态的监控以及对系统运行时的内存、CPU的使用情况的监控。

日志管理：日志管理主要是对系统工作时产生的日志进行管理，包括运行时日志，错误日志，存储日志等，包括日志的存储方式以及存储位置的管理；对日志的导入导出设置；以及对日志的分析。



图4.2 管理人员用例图

对于推荐系统来说，大部分的功能都是属于后台工作模块，这些工作模块基本上都是为用户的功能提供基础支撑的。因此对于系统大部分功能来说，都是以一种后台的工作模式来运行的。

## 个性化健康服务推荐系统总体设计

根据4.1节的需求分析，本节将主要介绍个性化健康服务推荐系统的总体设计，包括架构设计和具体的功能设计。

### 4.2.1 架构设计

推荐系统的架构一直是业界的一个讨论热点，相比于其他系统的架构，推荐系统有着它自身的特殊性，首先作为一个优良的推荐系统，得具备良好的推荐能力，即可以根据用户的实际使用情况，做出合理并且优秀的推荐结果，这部分的能力，主要是凭借成熟并且优秀的推荐方法和数据分析来提供保障的。其次，它还应该具备稳定运行与快速响应的能力，即可以保证长时间的运行不宕机也可以快速对相关事件做出相应的计算，并且可以实时的反馈给用户，来保证用户使用上的流畅性，而对于这一部分的保障，则是靠着稳定并且优秀的系统架构所决定的。

系统的架构主要采用三层架构，分别是请求层，业务处理层以及运行保障层，系统架构图如图4.3所示，请求层是最高层，负责对用户的请求做处理，当请求层将用户的请求处理完成后，会将任务分配给业务处理层；业务处理层接收到任务会，会根据任务的类型，执行不同的功能模块；运行保障层是最底层，这一层主要是和数据库做交互，为上一层的业务处理层做数据支撑。接下来会对系统的每一层做详细说明。



图4.3 系统架构图

请求层：请求层是用户与系统的交互入口和出口，这层的主要工作是负责接收和返回用户的请求。其包含的主要功能模块有请求解析模块、请求分配模块和数据通道模块。

请求解析模块：该模块的主要功能是对用户发送来的请求做解析，判断用户发送的请求类型，然后该模块会把请求分配给相应的业务模块。该模块首先会获得用户的请求消息，然后会进行字符串解析，和字典中的请求类型进行对比判断，得到的判断后的请求类型并把此请求交给请求分配模块处理。

请求分配模块：该模块的主要作用是根据不同的请求类型，调用不同的功能模块执行相应的任务。

数据通道模块：当请求中携带了数据时，该模块会被调用，该模块会把这些数据进行反序列化，形成能被程序设计语言理解的对象数据，这些对象数据可以方便在功能模块中的传递。

业务处理层：业务处理层是系统的核心层，在这一层里主要有两方面的作用。第一方面主要是对于数据的接入，当新用户或者老用户做数据更新时，经过请求层的解析，将任务分配到数据处理模块后，在该模块中会做相应的数据处理工作。另一方面是做数据的对外输出，当用户需要获取推荐内容时，经过请求层的解析，将任务分配到计算模块后，在该模块中进行相应的整理与计算后，将最终的计算结果通过网络传输协议返回给用户。业务处理层主要包含数据清洗模块、数据整理模块、结构化处理模块，聚类分析模块，协同过滤模块，结果呈现模块，计算线程池模块。

数据清洗模块：该模块主要功能是清洗数据中的空值与异常值。

数据整理模块：该模块的调用主要是在数据清洗模块的调用之后，该模块主要的功能是对清洗后的数据进行整理，是数据更加的规范化。

结构化处理模块：该模块是数据处理的最后一步，该模块的调用在数据整理模块后，该模块的主要功能是将整理后的结构化并且存入数据库。

聚类分析模块：该模块的调用是生成推荐的第一步，该模块的主要功能是调用第三方软件库Mahout中的聚类算法对地理位置信息和兴趣信息进行聚类。

协同过滤模块：该模块的调用是生成推荐的第二部，该模块的主要功能是调用第三方软件库Mahout 中的协同过滤算法来生成最后的推荐列表。

结果呈现模块：该模块的调用是生成推荐的最后一步，该模块的主要功能是对协同过滤模块产生的推荐列表进行数据包装，使之可以适应网络传输的要求，并通过网络传输给相应的用户。

计算线程池模块：该模块为整个业务处理层提供计算支持，采用线程池的方式实例化线程，并用配置文件控制线程池的配置与工作方式。

运行保障层：运行保障层是整个系统架构的最底层，它的主要功能是与数据库打交道，通过数据库连接池的作用，对数据库中进行数据的读取与数据的写入。运行保障层主要包含数据库连接池模块。

数据库连接池模块：该模块为上层的业务模块的数据需求提供支持，该模块向外提供接口，开发人员可以根据相应的业务需求来实现接口；同时它维护了数据库连接对象，通过配置文件进行相关的配置。

## 个性化健康服务推荐系统核心模块具体实现

在上一节中，对系统的总体架构进行了介绍。本节将对系统中的核心模块的实现做详细介绍。

### 4.3.1 数据清洗模块

数据清洗模块是数据处理的第一步，它主要的功能是对数据进行清洗工作。本小节将从实现的角度详细的介绍清洗模块，包括其流程图、类图和各函数详细说明等。

#### 数据清洗算法流程

根据第三章对数据清洗的分析可知，数据清洗可以对空值及噪声值进行清洗，因此设计的数据清洗算法流程如图4.4所示：



图4.4 数据清洗算法流程图

算法开始是由请求分配模块新建一个清洗算法对象，使清洗算法实例进入执行状态；第二步是依次的读入需要处理的数据字段；第三步是判断该数据是否是地理位置数据，如果不是，则根据第三章的分析可知该数据是个人主观输入的数据，噪声异常予以忽略，只需再次判断是否是空值异常，如果是空值异常，则抛弃这条数据。如若不是，则加入到最终的数据集合中去。如果是地理位置数据，则这条数据字段不存在噪声异常，则只需判断是否存在空值异常，如果存在则根绝第三章的分析，执行异常处理算法；第四步则把清洗后的数据加入到数据集合中，让后续的模块可以继续使用这些数据；第五步判断是否有下一个数据字段，如果有则重复第二步，如果没有，则结束算法流程。

#### 数据清洗模块类图

数据清洗的实现类主要有清洗接口，实现接口的抽象类，具体实现类和清洗运行环境类。数据清洗模块的实现类图如图4.5所示。



图4.5 数据清洗模块类图

数据清洗模块中的清洗接口为clearData，设计该接口的主要目的是为了增加编程的灵活性以及系统的可拓展性，该接口定义了两种行为，这两种行为规定了实现类的行为准则，让其实现类不会出现行为上的偏差。AbClearData是实现了clearData接口的抽象类，该类的设计满足了面向对象的编程思想，该类中实现了clearData接口中的方法，定义了通用的行为方法。LocationClear是AbClearData抽象类的实现类，该类的主要作用是对地理位置数据进行清洗，由于地理位置信息只需要对空值异常进行清洗，因此，在该类中只重写了清洗空值的函数。ClearContext类是清洗模块中的环境类，该类的主要作用是响应系统的指令，同时将清洗模块中的其他类进行整合，在ClearContext中使用多态的方法来灵活调用清洗的类，运用设计模式中的工厂模式，来创建清洗类，使清洗模块对于系统来说只有唯一接口。

总的来说，清洗模块的实现灵活运用了面向对象编程思想中的多态与继承的思想，使得逻辑清晰。同时，合理的运用设计模式，使得清洗模块的可拓展性与可读性得到了大幅提升。

#### 数据清洗模块主要函数功能

表4.1详细介绍了数据清洗模块各主要函数的功能。

表4.1 数据清洗模块主要函数功能

|  |  |  |
| --- | --- | --- |
| **类名** | **方法名称** | **方法含义** |
| clearData | clearNull () | 接口中定义行为的函数 |
| clearNoise() | 接口中定义行为的函数 |
| AbClearData | clearNull () | 抽象类中的实现接口行为的函数，定义了通用的清洗空值的方法 |
| clearNoise () | 抽象类中的实现接口行为的函数，定义了通用的清洗噪声值的方法 |
| LocationClear | clearNull () | 具体实现类，定义了清洗地理位置数据的方法 |
| findUser() | 获取数据库中和当前用户相似的全部用户，用来为clearNull方法服务 |
| getAva() | 得到用户经度和纬度的均值，用来为clearNull方法服务 |
| ClearContext | startClear () | 开始数据清洗 |
| choseClear() | 根据不同的数据类型，选择相应的清洗类 |
| getResult() | 数据清洗结束后的返回 |
| isNull() | 判断当前数据字段是否为空值 |
| addResult() | 将清洗后的数据字段加入到结果列表中 |

### 4.3.2 数据整理模块

数据整理模块是数据处理的第二步，该模块主要的功能一是对数据进行映射二是对数据进行整合，来适应数据结构化的需求。本小节将从实现的角度介绍本模块，主要包括整合的流程、实现的类图以及各主要函数的详细说明。

#### 数据整理算法流程

根据第三章对数据整理的分析，设计的数据整理算法流程如图4.6所示。



图4.6 数据整理算法流程图

第一步，对数据进行映射，这个操作主要是对JSON进行解析，把JSON中的字符串数据转换成内存中的对象，同时和数据库中的属性对应起来，映射的方法主要是根据预设好的映射表来实现对比映射。第二步是在数据映射结束后，从内存中依次读入数据，判断是否需要进行数据整合，如果不需要，则判断是否有下一个数据，如果需要则从分隔符产生器中读取分隔符。第三步，利用读取的分隔符进行字符串拼接操作，使得多个数据整合成一个数据，整合结束后，判断是否还有下一个数据，如果有则继续执行第二步，如果没有，则结束流程。

#### 数据整理模块类图



图4.7 数据整理模块类图

数据整理模块类图如图4.7所示，数据整理模块中的环境类为IntegrationContext类，该类是和系统交互的接口，该类中组合了ExecuteMap类和Integration类，该类可以控制这两个类的执行。ExecuteMap类的主要作用是用来做字段映射的，它可以根据Mapped类提供的映射规则来生成相应的对象。Mapped类主要是从静态类MappedRules中读取映射规则，而MappedRules类主要是和配置文件进行交互，它从配置文件中读取配置的映射规则，再交给Mapped类。SpeartorCreator类是用来产生分隔符的类，它可以根据配置文件来产生相应的分隔字符。Integration类可以读取Separator类产生的分隔符，然后读取相应的数据并与分隔符相结合做进一步整合。

#### 数据整理模块主要函数功能

表4.2详细列举了数据整理模块各主要函数的功能。

表4.2 数据整理模块主要函数功能

|  |  |  |
| --- | --- | --- |
| **类名** | **方法名称** | **方法含义** |
| MappedRules | getRules () | 从配置文件中读取相关的数据映射规则 |
| Mapped | readMapped () | 从MappedRules对象中读取数据映射规则 |
| isMap () | 判断是否可以映射 |
| getMap() | 得到映射后的数据类型 |
| ExecuteMap | startMap () | 开始执行数据映射操作 |
| CreatePersonalInformation() | 创建映射后的个人数据信息类 |
| SeparatorCreator | createSeparator() | 根据配置文件中的分隔符，并且创建此分隔符 |
| readConfigureFile() | 读取相关的配置文件，把配置文件中的分隔符读取到内存中 |
| Integration | startIntegration() | 开始数据整合操作 |
| readSeparator() | 读取SeparatorCreator类中的分隔符，并把分隔符应用到数据整合中 |
| IntegrationContext | startExecuteMap() | 创建ExecuteMap对象，并使ExecuteMap对象开始工作 |
| startIntegartion() | 创建Integartion对象，并使Integartion对象开始工作 |

### 4.3.3 聚类分析模块

聚类分析是生成推荐的第一步，该模块利用了Mahout开源软件库提供的聚类算法，对数据库中的相关数据进行了聚类分析，并保存了聚类的结果，并把聚类的结果提交给它的上一层进行进一步的处理。本小节将从与Mahout的交互流程、模块实现类图和主要函数功能进行介绍。

#### 聚类分析交互流程图

要与Mahout进行交互进行聚类操作，首先要满足Mahout定义的数据结构以及它的运行规则。为了使用Mahout提供的聚类算法，首先得将数据表达成Mahout提供的向量(Vector)数据结构，Mahout提供的向量数据结构中的每一个域都是浮点型(double)的符合结构。Mahout提供了许多Vector的实现类，比如DenseVector类，RandomAccessSparseVector类和SequentialAccessVector类；除此之外，还可以实现Mahout提供的Vector抽象类，来自定义业务所需的向量类。在满足了Mahout的向量数据结构后，就可以调用Mahout提供的聚类算法，其交互流程如图4.8所示。



图4.8 聚类分析模块交互流程图

#### 数据分析模块类图



图4.9 聚类分析模块类图

聚类分析模块的类图如图4.9所示，AbCluster是聚类算法的执行抽象类，该类的主要作用定义通用的算法调用模板。MyCluster是继承自AbCluster类，该类是具体的实现类，该类调用了Mahout中的Kmeans算法，实现了K均值聚类的实现。VectorFactory是向量创建类，这个类的主要功能是对数据进行向量化，使数据可以适应算法。ResultSave类是结果保存类，这个类的主要作用是对聚类的结果进行展示与保存。

#### 聚类分析模块主要函数功能

表4.3详细介绍了聚类分析模块各主要函数的功能

表4.3 聚类分析模块主要函数功能

|  |  |  |
| --- | --- | --- |
| **类名** | **方法名称** | **方法含义** |
| ResultSave | print () | 对得到的聚类结果进行格式化的控制台打印 |
| saveAsFile () | 将结果保存成文件 |
| setFileType () | 设置保存文件的文件类型 |
| MyCluster | setK() | 设置K均值算法中的K值 |
| setMaxIter() | 设置最大迭代次数 |
| setDistanceMeasure() | 设置距离计算方法 |
| setCluster() | 设置K均值聚类器 |
| startCluster() | 开始k均值聚类算法 |
| AbCluster | setCluster () | 设置通用的聚类器 |
| startCluster() | 开始通用的聚类算法 |
| VectorFactory | createVetcor() | 创建相应的向量转换器 |
| getType () | 得到向量转换器的类型 |
| getPointVector() | 得到二维点向量转换器 |

### 4.3.4 协同过滤模块

协同过滤模块是形成推荐的第二步，该模块利用上一步聚类分析产生的数据，调用Mahout提供的算法，通过对数据的分析与计算，来生成最终的推荐结果。本小节将从与Mahout的交互流程、模块实现类图和主要函数功能进行介绍。

#### 协同过滤模块交互流程图



图4.10 协同过滤模块交互流程图

协同过滤模块与Mahout 的交互较为简单，其流程如图4.10所示。首先，为了满足Mahout执行的数据模型，需要用Mahout提供的数据模型接口，会根据相应的接口来适配数据，这里Mahout提供了数据库和文件的数据接口。第二步，在建立好数据接口后，选择是用基于物品的推荐算法还是基于用户的推荐算法，根据第三章的分析，本文选择基于用户的推荐。第三步则调用Mahout执行算法，并得到最终的结果。

#### 协同过滤模块类图



图4.11 协同过滤模块类图

协同过滤模块的类图如图4.11所示。在协同过滤模块中ModelFactory类的主要作用是生成和使用相关的数据模型，该数据模型主要是为了适配Mahout中的数据模型；ResultInterface的主要功能是结果的对外输出，由于协同过滤模块基本上是生成推荐的最后一步，所以产生的结果应该对外输出；RecommderContext类是协同过滤环境类，该类是和系统交互的接口，控制着协同过滤算法的开始。

表4.4详细介绍了协同过滤模块各主要函数的功能

#### 协同过滤模块主要函数功能

表4.4 协同过滤模块主要函数功能

|  |  |  |
| --- | --- | --- |
| **类名** | **方法名称** | **方法含义** |
| ModelFactory | createModel () | 创建数据模型对象 |
| getModelType () | 得到数据模型的类型 |
| getModel () | 得到模型实例 |
| RecommderContext | getModel() | 得到模型实例 |
| startRecommder() | 开始协同过滤算法 |
| setSimilarity() | 设置相似度算法 |
| ResultInterface | saveAsFile() | 将最终结果保存成文件 |
| setFileType() | 设置保存文件的类型 |
| print() | 将最终结果进行控制台打印 |
| output() | 将结果传给上一层，准备做、网络传输发送给用户 |

## 本章小结

本章是论文的主要工作之一，本章从具体实现的角度详细阐述了个性化健康服务推荐系统的功能。本章首先从需求分析出发，确定了系统应该具有的功能。然后对系统的总体进行了设计，包括对系统的功能模块进行的讨论和对系统的架构设计进行了分析与阐述。最后介绍了系统的核心模块的具体实现，包括数据清洗模块、数据整理模块、聚类分析模块和协同过滤模块的设计与实现。

# 运行与测试

本文第三章介绍了一种个性化健康服务推荐方法，第四章介绍了推荐系统的设计与实现。为了验证方法的正确性以及系统运行的正确性，本章将设立有针对性的测试。本节将会验证方法的准确性以及系统的可运行性，本节将会对系统的各个模块和系统的性能进行测试，通过测试产生的数据来对方法中的数据采集方案，数据清洗方案以及整理方案进行验证，然后会验证推荐方案的实施；然后会对系统的性能进行测试，包括测试整体系统的响应速度，主要模块的执行效率，压力响应能力。

## 测试环境

本章的测试对象是个性化健康服务推荐系统，测试分为两大部分。

第一部分是对功能上的测试，功能上的测试通过对系统中的数据清洗模块、数据整理模块、聚类分析模块和协同过滤模块等模块的测试，验证系统的运行是否良好以及方法执行的是否正确。根据第四章中的介绍，首先本章通过客户端传输数据到服务器，验证服务器是否可以解析请求和分发请求任务；接着会对清洗模块进行测试，判断是否可以完成清洗功能；然后会整理模块进行测试，验证其是否可以完成数据结构化的功能；接着会对聚类分析模块做测试，验证其是否可以很好的形成簇关系；最后测试协同过滤模块，来验证是否可以推荐。

第二部分是对系统性能上的测试，性能上的测试又主要是对系统的响应时间、各模块的工作效率和压力响应能力测试。根据第四章的需求分析，首先会对系统整体响应时间和各个模块的响应时间做测试，通过模拟用户的请求，来测试系统的整体的响应时；接着会在系统正常工作的情况下，测试系统主要模块的工作效率；最后会对系统的压力响应能力做测试，通过模拟大量用户的同时请求，来测试系统在大压力的情况下，是否仍旧可以响应请求。

本文所述的方法与系统，已经应用在实验室项目“易医”平台中，“易医”平台是一个面向医生和普通人群（患者和大众）的健康服务平台，以下测试全部是基于“易医”平台完成的。下面对实验的测试环境做说明。

客户端运行环境

* 客户端型号：Honor FRD-AL10
* 处理器：Hisilicon Kirin 950
* 运行内存：4.0GB
* 操作系统：Android 7.0

服务器运行环境

* 处理器：Intel(R) Core(TM) i5-3470 CPU @ 3.20G
* 系统内存：4GB(DDR3 1600MHz)
* 网卡：10/100M 自适应以太网卡
* 操作系统：Microsoft Windows 7(64位)
* 数据库：MySQL2005
* 网络容器：Tomcat 7.0.62

## 测试过程

### 5.2.1 功能测试

功能测试主要包括对数据清洗模块、数据整理模块、聚类分析模块以及协同过滤模块功能完整性的测试，以此来验证功能是否可用及完善。设计一个基础测试用例，它的具体形式如图5.1所示：

|  |  |
| --- | --- |
| {“person”:{  “name”:”李云”,“age”:36,“sex”:1},  “latitude”: 34.346482,  “longitude”: 108.950905,  “history\_hospital”:[  {“name”: ”唐都医院”}  ],  “history\_doctor”:[  {“name”:”范浩”}  ],  “history\_disease”:[  {“name”:”肺炎”}  ], | “history\_drug”:[  {“name”:” 第二代头孢菌素”}  ],  “interest\_hospital”:[  {“name”:”西京医院”}  ],  “interest\_doctor”:[  {“name”:”王德林”}  ],  “interest\_disease”:[  {“name”:”肺炎”}  ]} |

图5.1 基础测试用例

为了验证数据采集的正确性，采用Android客户端对数据进行采集，采集过程如图5.2所示。

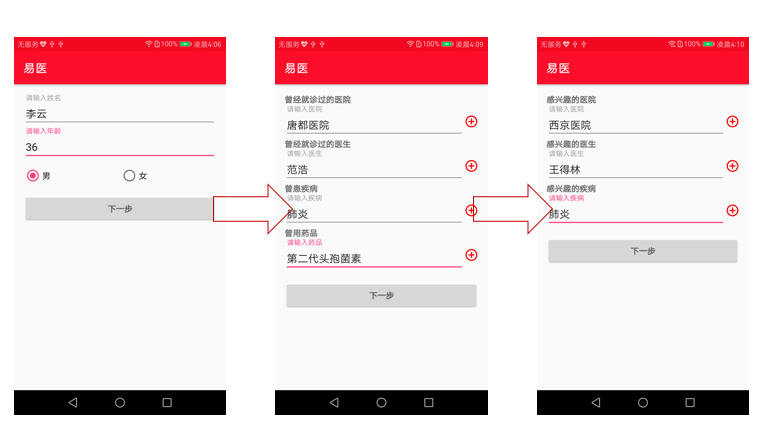


图5.2数据采集过程

通过在数据采集过程中的输入变化，对这个基础测试用例的变化，来验证各个功能模块的功能性需求，具体测试用例变化如表5.1所示。

表5.1 功能测试用例

|  |  |  |
| --- | --- | --- |
| 用例编号 | 对基础用例的修改处 | 说明 |
| 用例1 | 无修改 | 以正确的数据，来验证数据清洗模块和数据整理模块工作的正确性 |
| 用例2 | 修改person中的name为马云，修改latitude和longitude的字段值为0 | 验证数据清洗模块是否可以补全latitude和longitude的值 |
| 用例3 | 修改person中的name为李阳，修改person中的name的字段值为0 | 验证数据清洗模块是否会抛弃无法清洗的数据 |
| 用例4 | 修改person中的name为李东，修改history\_hospital数组，增加键值对：“name”: ”西京医院” | 验证数据整理模块是否会进行数据整理及结构化处理 |
| 用例5 | 增加大量和基础测试用例相似的数据用例 | 验证聚类分析模块及协同过滤模块工作的正确性 |

测试过程如下，首先对测试用例1进行测试，测试结果如图5.2所示，该图表明，当输入数据是正确无异常的时候，数据清洗模块和数据整理模块的工作正常，测试用例可以被数据存入数据库。

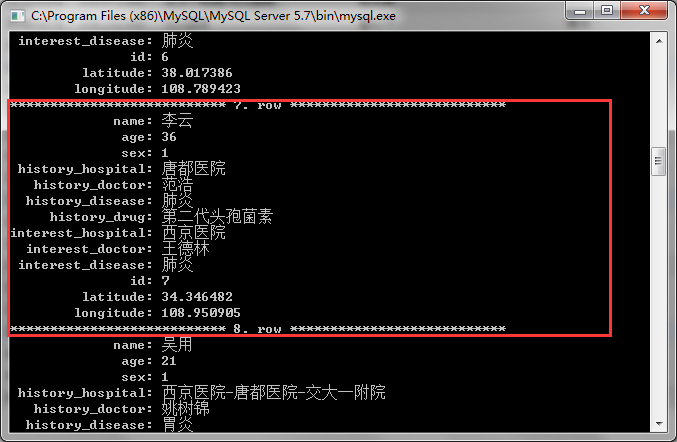


图5.3 测试用例1执行结果

接着对测试用例2进行测试，测试结果如图5.3所有，该图表明，当输入的地理位置信息数据存在空值异常时，数据清洗模块可以正常的工作，测试用例中的空值被进行了补全处理，并存入数据库中。

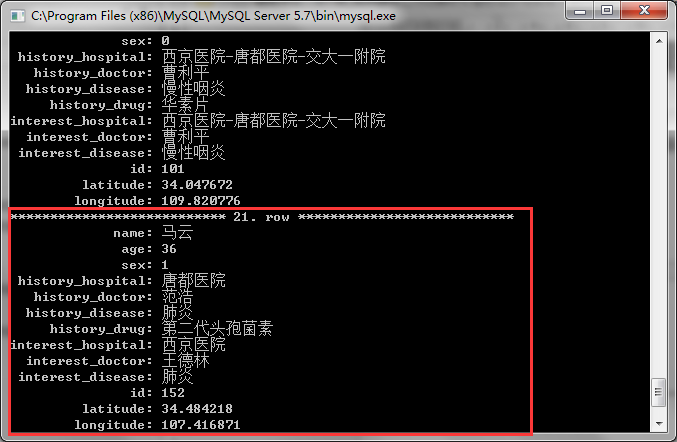


图5.4 测试用例2执行结果

对测试用例3进行测试，测试结果如图5.4所示，该图表明，当输入的其他数据出现空值异常时，数据清洗模块工作正常，测试用例被抛弃该，这与预期的效果一致。

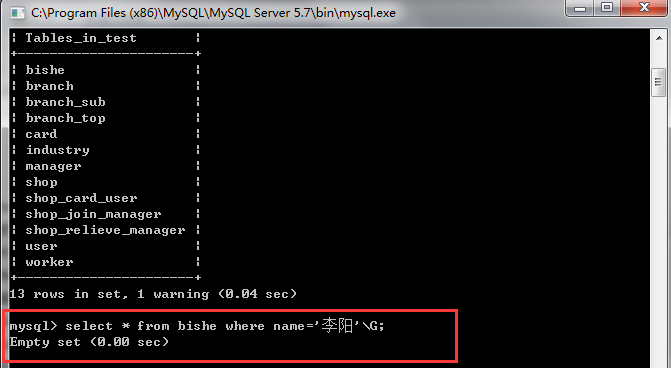


图5.5 测试用例3执行结果

对测试用例4进行测试，测试结果如图5.5所示，该图表明，当输入的数据出现JSON数组型的数据时，数据整理模块工作正常，测试用例中的history-hospital字段会被进行结构化处理，并存入数据库中。

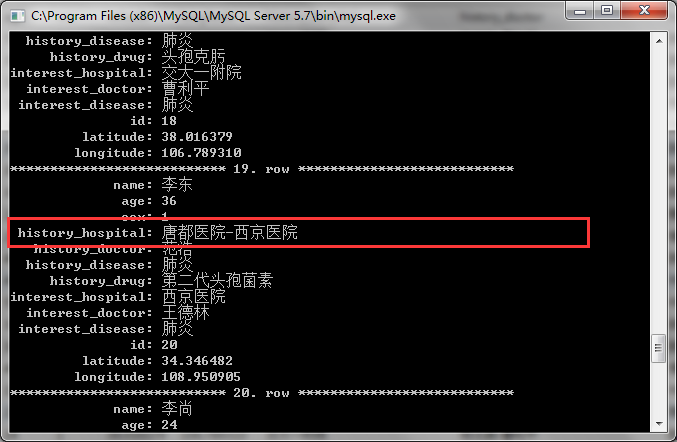


图5.6 测试用例4执行结果

对测试用例5进行测试，测试结果如图5.7所示，该测试结果表明，聚类分析模块及协同过滤模块工作正常，产生的推荐结果包括医生推荐和医院推荐，并且可以返回给客户端显示。

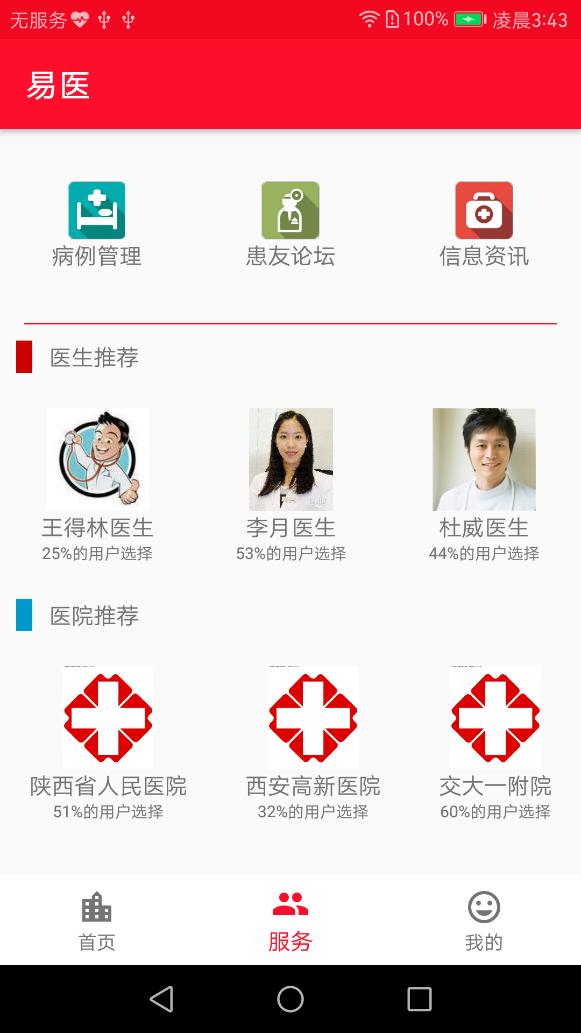


图5.7 测试用例5执行结果

### 5.2.2 性能测试

性能测试主要测试的系统在性能上的功效，本节对系统的性能测试主要包含以下两个方面，分别是系统响应时间的测试，系统压力响应时间的测试。

首先为了测试系统的响应时间，综合考虑用户的两种请求类型，以及用户使用的网络状况，设计测试用例如表5.2所示。用例的设计方法采用的是控制变量法，即固定某一条件，变换另一条件来测试。

表5.2 系统响应时间测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例 | 请求类型 | 环境条件 | 测试次数 |
| 用例1 | 数据提交请求 | 使用4G网络，且网络条件好 | 100 |
| 用例2 | 数据提交请求 | 使用3G网络，且网络条件好 | 100 |
| 用例3 | 数据提交请求 | 使用2G网络，且网络条件好 | 100 |
| 用例4 | 数据提交请求 | 网络条件较差 | 100 |
| 用例5 | 获取推荐请求 | 使用4G网络，且网络条件好 | 100 |
| 用例6 | 获取推荐请求 | 使用3G网络，且网络条件好 | 100 |
| 用例7 | 获取推荐请求 | 使用2G网络，且网络条件好 | 100 |
| 用例8 | 获取推荐请求 | 网络条件较差 | 100 |

这里使用程序来模拟请求，系统整体响应时间结果如图5.7所示。测试结果表明，对于不同的网络条件以及不同的请求类型，系统的响应时间有所不同。在网络条件相同时，获取推荐请求的用时要比数据提交请求长；而在请求类型相同时，使用的网络制式越好，响应时间越快。而对于在网络条件较差的测试时，大多数的请求响应时间都超过了10秒(一般来说，超过10秒的请求，都算请求失败)。即除了网络环境较差的极端情况下，系统的响应时间表现尚可，基本上可以满足使用需求。

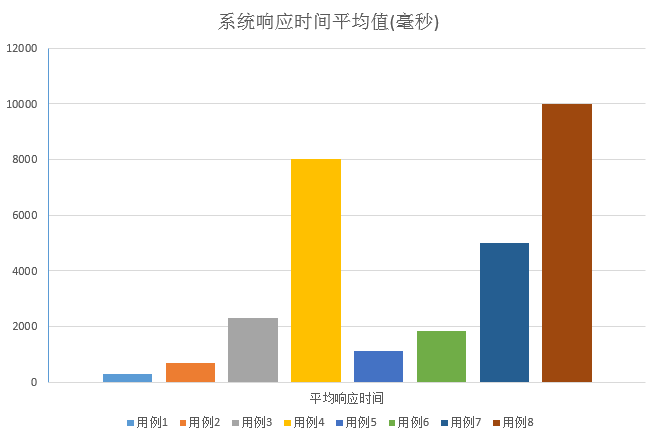


图5.8系统整体响应时间

接着，会对系统中各主要模块的工作效率进行测试，采用的测试用例如表5.3所示。根据实际的业务需求情况，设计了两种请求类型的测试用例，分别测试各个模块在少数据量和大数据量下的工作效率。

表5.3 各主要模块工作效率测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例 | 请求类型 | 测试模块 | 环境条件 | 测试次数 |
| 用例1 | 数据提交请求 | 数据清洗、数据整理模块 | 提交的数据约1kb | 100 |
| 用例2 | 数据提交请求 | 数据清洗、数据整理模块 | 提交的数据约10kb | 100 |
| 用例3 | 数据提交请求 | 数据清洗，数据整理模块 | 提交的数据约50kb | 100 |
| 用例4 | 获取推荐请求 | 聚类分析、协同过滤模块 | 10条用户数据 | 100 |
| 用例5 | 获取推荐请求 | 聚类分析、协同过滤模块 | 100条用户数据 | 100 |
| 用例6 | 获取推荐请求 | 聚类分析、协同过滤模块 | 500条用户数据 | 100 |

采用程序模拟的方式对系统进行测试，测试结果如图5.8所示。测试结果表明，数据清洗和数据整理模块的在不同数据的测试条件下，工作时间较短，效率较为稳定，基本上是线性增加。而聚类分析和协同过滤模块在不同数据的测试条件下，工作时间在可接受的范围内，但工作效率以非线性增长。

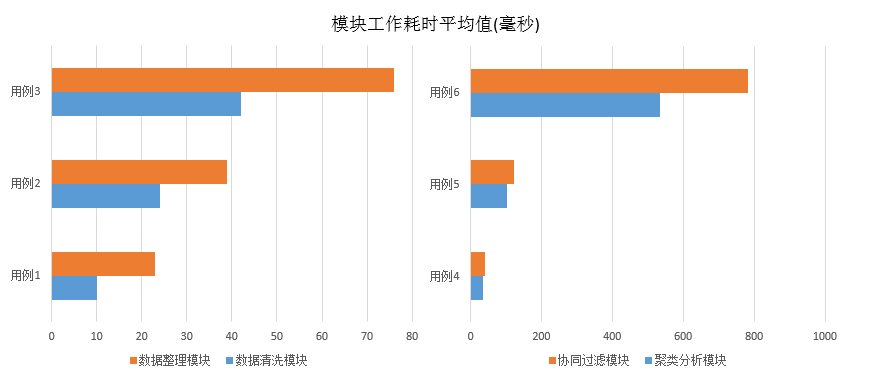


图5.9 主要模块工作效率

最后，对系统压力响应进行测试。系统压力响应指的是，当系统在某一时间段内遇到大量的请求时，系统可以对响应请求的数量。这里用程序来模拟一段时间内的大量请求。设计测试用例如表5.4所示。因为该测试主要是压力测试，因此选择较好的网络，排除因网络原因引起的测试误差，对每种请求类型设计从低到高的三种压力请求。

表5.4 系统压力响应时间测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例 | 请求类型 | 测试条件 | 环境条件 | 测试次数 |
| 用例1 | 数据提交请求 | 10/秒 | 4G网络，网络环境较好 | 100 |
| 用例2 | 数据提交请求 | 100/秒 | 4G网络，网络环境较好 | 100 |
| 用例3 | 数据提交请求 | 1000/秒 | 4G网络，网络环境较好 | 100 |
| 用例4 | 获取推荐请求 | 10/秒 | 4G网络，网络环境较好 | 100 |
| 用例5 | 获取推荐请求 | 100/秒 | 4G网络，网络环境较好 | 100 |
| 用例6 | 获取推荐请求 | 1000/秒 | 4G网络，网络环境较好 | 100 |

测试执行结果如图5.9所示，结果表明，在相同请求类型的情况下，每秒的请求次数越多，系统响应的请求越少，在测试用例3和测试用例6的测试中，系统无法响应大量的请求，甚至出现系统崩溃的情况。在测试条件相同的情况下，数据提交请求的响应个数要比获取推荐请求多，这是因为数据提交请求的请求量小，不需要系统进行复杂的计算，而获取推荐数据传输量大，系统也需要大量的计算。结果表明，对于超过前次的请求来说，系统会出现大量的请求失败的情况。但是，基于实际使用情况来看，处理百次级的请求在可接受的范围内。

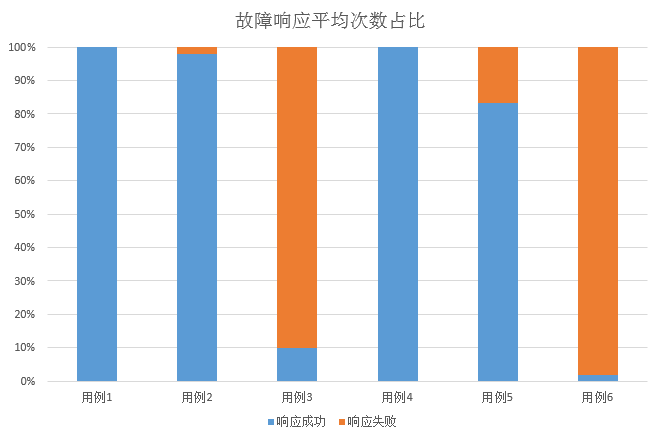


图5.10 系统压力响应

## 测试结果分析

对个性化健康服务推荐系统的功能测试结果表明，该系统具有完成相关的数据清洗功能，对空值的处理，以及噪声的处理与预期相符合；在数据处理模块中，对数据的整理以及结构化的处理与预期相符；同时，该系统具备聚类分析和协同过滤计算的能力，可以对大量的数据进行分析与计算。

对个性化健康服务推荐系统性能的测试结果表明，该系统在系统响应时间上具有优良的效果，既在网络条件尚可的情况下，可以在2秒内完成响应。同时，该压力请求的情况也，也有着一定的能力，在每秒请求不上千的条件下，响应成功率基本在80%以上。

综合测试结果来看，本系统基本完成了第四章需求分析中的要求，系统的功能完整性良好，系统的性能也在可接受的范围内。

## 本章小结

本章节对本文之前提出的个性化健康服务推荐方法和系统做出了测试，本章节首先介绍了相关的实验环境，接着分析了需要测试的方面，即功能测试和性能测试，然后分别设计了不同的测试用例和测试方案，最后对测试的结果做了一定的分析。本章节的内容证明了本文提出的个性化健康服务推荐方法和系统真实有效，在实际使用过程中具有一定的价值。

# 总结与展望

本文主要针对当前医疗资源分配不均衡，信息过载问题严重，人们对医疗资源的选择上有一定困难的现象，提出了一种利用机器学习、数据挖掘等方法的解决方案。在医疗信息化的发展越来越迅猛的前提下，利用相关的医院、医生患者等医疗信息数据，通过对这些相关的医疗信息数据的分析与处理，形成一种良好的医疗资源的推荐效果，来帮助人们有针对性的选择自己所需要的医疗资源。

本文提出了一种应用于健康服务领域的推荐方法，对医疗数据的采集过程，处理过程和推荐方法的设计都进行了详细的讨论与研究，在医疗数据采集过程做了一定的研究，以及在数据采集完成后的数据表示也应用了相关成熟的表示方式；在数据的整理过程中，对采集到的数据存在的空值、异常值的情况进行了介绍，并且给出了对这些类情况的处理方案，然后在数据的结构化存储过程中，也详细的介绍了数据的结构化方法；最后再推荐方法的设计上，本文综合考虑了数据的多维情况，提出了一种二次聚类并结合协同过滤算法的方法，并且利用相关成熟的软件库做了实现。其次，本文为了满足实际业务情况的需求，设计并实现了一个稳定高效的推荐系统方案。该推荐系统采用三层架构，并有着较高的拓展性，可以适应较为复杂的软件运行环境，第一层作为请求层，是用来处理用户请求并且根据请求分发任务的，通过解析用户的请求来调用相关的功能模块；第二层是业务处理层，是系统的核心执行部分，该层利用计算的线程池来做数据的处理业务和数据的计算业务；第三层是运行保障层，这一层主要是和数据库直接打交道，采用数据库连接池的方式，负责向上层提供需要的数据。在本文的最后，本文以实际场景，分别从数据采集、数据整理以及推荐方法上，对本文提出的健康服务推荐方法做了验证。然后，对系统的运行做了验证。验证的结果表明，该方法是有效的，系统的运行也是良好的，性能上也基本上可以达到实际使用的需求。

在取得了预定的成果时，仍旧有着不足与待改进的地方。首先，在推荐方法的设计上仍旧有着改良的空间，在数据的采集中，本方法采用的是JSON的数据表达方法，对于医疗数据来说，这存在着数据冗余与表达方式不清的问题，在后续工作中对医疗数据的表达仍旧有着探索的空间，在对医疗数据的空值和异常值的处理中，对空值和异常值采用了抛弃的办法，这在存在着一定的数据遗漏问题，可能会对最后的推荐结果造成一定的影响，在结构化的处理中，本文利用分隔符的方案，使得数据可以很好的适应二维表的结构，然而用分隔符的方案，容易造成解析异常，即分隔符与实际字符冲突的情况。在最后的推荐中，对与医疗资源无评分这一问题仍旧有着研究的必要，本文采用的默认评分的模式多多少少会影响着推荐的准确性。

其次，在推荐系统层面上，系统的计算模式上仍有改进的空间，本系统采用的是在线计算的方法，然而在实际情况中，往往可能由于数据量过大，导致计算不及时影响用户的体验，没有提供在线和离线两种计算模式的选择。其次，本系统对数据缓存的开发不足，所有的数据读取或写入都是直接操作数据库完成的，而对数据库的操作往往又是耗时的，这将影响系统的相应速度。

本文主要的工作集中在个性化健康推荐方法和系统的研究上，由于本人的水平有限，在某些方面未能进行深入的探讨与研究，存在着谬误也在所难免，敬请各位评委老师指导。

参考文献

1. 魏婉. 关于促进和规范健康医疗大数据应用发展的指导意见[J]. 科研, 2016(09):00306-00307.
2. Huang Z, Lu X, Duan H, et al. Collaboration-based medical knowledge recommendation[J]. Artificial Intelligence in Medicine, 2012, 55(1):13.
3. 海川. 大数据助推个性化医疗[J]. 新经济导刊,2014,(09):42-46.
4. 王国霞, 刘贺平. 个性化推荐系统综述[J]. 计算机工程与应用, 2012, 48(7):66-76.
5. 刘宁,武琼,陈敏. 个性化医疗服务类型及相关数据资源研究[J]. 中国卫生信息管理杂志,2016,(01):93-98.
6. 新华网.美国总统奥巴马拟推动发展个性化医疗[EB/OL][2015-01-31]. http://news.xinhuanet.com/tech/2015- 01/31/c\_1114203224.html
7. 中国慢病管理网.建立慢病全程防治管理模式:2015 年十一项疾控重点工作发布 [EB/OL][2015-01-16]. http://www.ncd.org.cn/Article/index/ id/3989.
8. J. H. Kim, D. S. Lee, K. Y. Chung. Item recommendation based on context-aware model for personalized u-healthcare service [J]. Multimedia Tools and Application, 2014, 71(2): 855-872.
9. L. Duan, W. N. Street, E. Xu. Healthcare information systems: data mining methods in the creation of a clinical recommender system [J]. Enterprise Information Systems, 2011, 5(2): 169-181. DOI:10.1080/17517575.2010.541287.
10. Lin Y F, Shie H H, Yang Y C, et al. Design of a real-time and continua-based framework for care guideline recommendations.[J]. International Journal of Environmental Research & Public Health, 2014, 11(4):4262-4279.
11. 杨杰. 个性化推荐系统应用及研究[D]. 中国科学技术大学, 2009.
12. Burke R. Hybrid Recommender Systems: Survey and Experiments[J]. User Modeling and User-Adapted Interaction, 2002, 12(4):331-370.
13. Goldberg D. Using collaborative filtering to weave an information tapestry[J]. Communications of the Acm, 1992, 35(12):61-70.
14. Resnick P, Iacovou N, Suchak M, et al. GroupLens: an open architecture for collaborative filtering of netnews[C] ACM Conference on Computer Supported Cooperative Work. ACM, 1994:175-186.
15. Resnick P, Varian H R. Recommender systems[J]. Communications of the ACM, 1997, 40(3): 56-58.
16. Linden G, Smith B, York J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering[J]. IEEE Internet Computing, 2003, 7(1):76-80.
17. Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge & Data Engineering, 2005, 17(6):734-749.
18. W. Y. Zhang, S. Zhang, Y. G. Chen, X. W. Pan. Combining social network and collaborative filtering for personalized manufacturing service recommendation [J]. International Jouranl of Production Research, 2013, 51(22): 6702-6719
19. Chen T, Zhang W, Lu Q, et al. SVDFeature: a toolkit for feature-based collaborative filtering[J]. Journal of Machine Learning Research, 2012, 13(1):3619-3622.
20. Chin W S, Yuan B W, Yang M Y, et al. LIBMF: a library for parallel matrix factorization in shared-memory systems[J]. Journal of Machine Learning Research, 2016, 17.
21. Ekstrand M D, Ludwig M, Konstan J A, et al. Rethinking the recommender research ecosystem: reproducibility, openness, and LensKit[C]. ACM Conference on Recommender Systems, Recsys 2011, Chicago, Il, Usa, October. DBLP, 2011:133-140.
22. PANG-NINGTAN, MICHAELSTEINBACH, VIPINKUMAR. 数据挖掘导论:完整版[M]. 人民邮电出版社, 2011.
23. 金建国. 聚类方法综述[J]. 计算机科学, 2014, 41(S2):288-293.
24. Hartigan J A, Wong M A. Algorithm AS 136: A K-Means Clustering Algorithm[J]. Applied Statistics, 1979, 28(1):100-108.
25. Cai Y, Sun Y. ESPRIT-Tree: hierarchical clustering analysis of millions of 16S rRNA pyrosequences in quasilinear computational time[J]. Nucleic Acids Research, 2011, 39(14):e95.
26. 许海玲, 吴潇, 李晓东,等. 互联网推荐系统比较研究[J]. 软件学报, 2009, 20(2):350-362.
27. Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]. International Conference on World Wide Web. ACM, 2001:285-295.
28. Papagelis M, Plexousakis D. Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents[J]. Engineering Applications of Artificial Intelligence, 2004, 18(7):781-789.
29. 刘建国, 周涛, 汪秉宏. 个性化推荐系统的研究进展[J]. 自然科学进展, 2009, 19(1):1-15.
30. Balabanovi, Marko, Shoham Y. Fab: content-based, collaborative recommendation[J]. Communications of the Acm, 1997, 40(3):66-72.
31. 张丙奇. 基于领域知识的个性化推荐算法研究[J]. 计算机工程, 2005, 31(21):7-9.
32. 詹尼士, 赞克, 弗里德里克. 推荐系统[M]. 人民邮电出版社, 2013.
33. 张驰, 陈刚, 王慧敏. 基于混合推荐技术的推荐模型[J]. 计算机工程, 2010, 36(22):248-250.
34. Breese J S, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering[C]// Fourteenth Conference on Uncertainty in Artificial Intelligence. 2013:43--52.
35. ECMA-404 the JSON Data Interchange Standard.[EB/OL]. http://www.json.org/json-zh.html.
36. 高静, 段会川. JSON数据传输效率研究[J]. 计算机工程与设计, 2011, 32(7):2267-2270.
37. SEAN OWEN. MAHOUT实战(图灵程序设计丛书)[M]. 人民邮电出版社, 2014.
38. Mahout算法集.[EB/OL]. http://blog.csdn.net/tanxinwhu/article/details/7091452.
39. Barrachina A D, O’Driscoll A. A big data methodology for categorising technical support requests using Hadoop and Mahout[J]. Journal of Big Data, 2014, 1(1):1-11.
40. TOP 10 开源的推荐系统简介.[EB/OL].http://www.oschina.net/news/51297/top-10-open-source-recommendation-systems唐绪军. 报业经济与报业经营[M] .北京: 新华出版社, 1999: 117-121.

致谢

作者简介

##### 基本情况

XXX，男，四川眉山人，1991年9月出生，西安电子科技大学计算机学院软件工程领域2014级硕士研究生。

##### 教育背景

2010.08～2014.07南京邮电大学，本科，专业：软件工程

2014.08～ 西安电子科技大学，硕士研究生，专业：软件工程

##### 攻读硕士学位期间的研究成果

###### 发表学术论文

###### 申请（授权）专利

###### 参与科研项目及获奖

1. “麦芒杯”第一届全国研究生移动终端应用设计传新大赛. 全国三等奖, 2016.8