

# 基于协同过滤算法的微信小程序智能助手<sup>①</sup>



刘 勇, 李永杰

(青岛科技大学 信息科学与技术学院, 青岛 266061)

通讯作者: 李永杰, E-mail: 975322135@qq.com

**摘 要:** 随着信息技术和网络的不断迅猛发展, 互联网的信息资源急剧增长. 信息过载问题促进了个性化推荐技术发展. 协同过滤算法通过在用户和信息之间建立联系, 被广泛应用于电子商务各个领域. 本文提出通过利用微信小程序来获取用户的个性化信息数据, 并且通过协同过滤算法, 为用户设计的微信小程序智能助手, 能够为用户推荐符合用户个性化的生活服务信息. 在本文中, 介绍了智能助手的设计方法, 并详细介绍了系统的功能和个性化推荐功能的实现.

**关键词:** 协同过滤; 微信小程序; 移动开发; 推荐系统; 大数据

引用格式: 刘勇, 李永杰. 基于协同过滤算法的微信小程序智能助手. 计算机系统应用, 2019, 28(5): 71-76. <http://www.c-s-a.org.cn/1003-3254/6931.html>

## Intelligent Assistant for WeChat Mini Program Based on Collaborative Algorithm

LIU Yong, LI Yong-Jie

(Information Science and Technology Academy, Qingdao University of Science and Technology, Qingdao 266061, China)

**Abstract:** With the rapid development of information technology and network, the information resources of the Internet have increased dramatically. The problem of information overload has promoted the development of customized recommendation technology. Collaborative filtering algorithm is widely used in various fields of e-commerce by establishing links between users and information. In this study, it is proposed that the user's personalized information data can be obtained by using the WeChat mini program, and through the collaborative filtering algorithm, an intelligent assistant designed for users can recommend customized life service information for users. In this paper, the design method of intelligent assistance is introduced, and the realization of system functions and customized recommendation function introduces elaborated in detail.

**Key words:** collaborative filtering; WeChat mini program; mobile development; recommend system; big data

随着移动互联网的发展, 微信成为目前国内最大的互联网入口之一. 微信每天为用户提供上亿次搜索服务. 随着腾讯对微信小程序的推广, 微信小程序借助微信巨大流量入口迅速崛起. 我们可以通过用户在碎片时间对微信小程序的使用来获取海量的用户习惯和不同用户的个性特点, 并且利用微信小程序可以迅速扫码传播<sup>[1]</sup>, 因此可以迅速获取海量不同类型的用户数据. 因此我们利用协同过滤技术, 微信小程序开放数据

平台, 互联网技术, 构建基于协同过滤算法的微信小程序智能助手, 通过分析用户的行为习惯和个性特点, 智能助手能够在合适的时间和地点为用户推送合适的服务信息, 这样可以节省大量用户搜索的时间.

## 1 系统概述

基于协同过滤算法的微信小程序智能助手借助微

① 收稿时间: 2018-12-04; 修改时间: 2018-12-25; 采用时间: 2019-01-08; csa 在线出版时间: 2019-05-01

信,将用户在微信上的操作行为和兴趣爱好数据上传到平台,然后与协同过滤算法结合.基于地理位置的分析,为用户在提供实时的信息推荐服务.根据用户的具体位置和相关时间点,为用户推送相关信息服务和事件提醒,同时微信小程序智能助手可以根据用户的个性特点为用户推送餐饮和乘车等相关的生活服务信息.

基于协同过滤算法的微信小程序智能助手主要由数据采集层,数据分析层,数据存储层,数据接口层和数据应用层四大部分组成.系统总体架构示意图如图1.

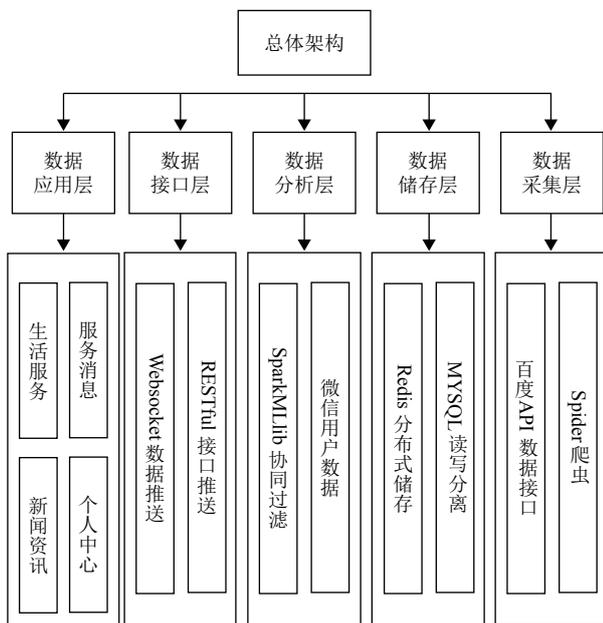


图1 系统总体架构图

### 1.1 数据采集层

数据采集层主要包括三部分.1)利用百度地图接口,获取用户所在地区的公交与地铁的路线资料.同时获取公交站牌地理位置信息与地铁口相关地理位置信息.2)利用Python的Scrapy框架,通过爬虫从网页获取不同地区的餐饮、商店、加油站等相关数据并提交至服务器.3)人工数据录入.对于部分地区或资料无法获取,利用手机小程序端和电脑端将一些数据进行填写并提交.

### 1.2 数据分析层

数据分析层主要通过分析用户数据,利用协同过滤算法对用户推荐相关的个性化信息.其中数据主要包括两部分.1)基于用户在微信的搜索或点击查看记录.同时利用微信提供的数据地理位置相关服务,可以

将用户相关记录与用户所处地理位置同时上传至数据分析层.2)基于用户对推荐服务的反馈.当微信小程序智能助手为用户推荐相关生活服务信息,用户对于结果的反馈数据也将提交至服务器.采用开源框架SparkMLlib框架进行协同过滤<sup>[2]</sup>,该框架支持分布式处理,解决海量数据处理速度过慢的问题.

### 1.3 数据存储层

由于采集的数据与用户个性化数据比较繁杂,其中包括结构化与非结构化数据.因此采用了关系型数据库MySQL与非关系型数据库Redis分别存储不同的数据.对于车站、餐饮以及店铺等信息,由于数据资料比较稳定,不会被频繁修改,将其保存至MySQL数据库.同时由于其数据量较大,数据访问压力较大,我们采用MySQL读写分离技术,这样避免数据库同时读写造成的访问缓慢的问题.而对于用户在使用微信小程序智能助手产生的记录数据,由于需要被频繁更新,且数据量大的特点,则存储于Redis数据库中,并且会在特定时间对数据持久化操作.

### 1.4 数据接口层

基于Laravel框架提供RESTful接口,微信小程序智能助手利用该接口进行数据交互与处理.RESTful原则是客户端与服务器之间交互请求无状态的,多数的HTTP请求都被转移到前端服务器上,降低服务器的负荷,因此十分适合云计算环境<sup>[3]</sup>.同时利用Swoole框架对数据分析层产生的推荐数据进行实时推送.

### 1.5 数据应用层

数据管理层主要包括基于移动终端的微信.微信小程序智能助手可以根据用户实时位置和时间来给用户推送相关生活服务信息,包括公交到站提醒,就餐提醒等.并且,用户可以查看推荐的相关信息,包括基于用户饮食特点的美食等数据.

## 2 系统设计与实现

### 2.1 软件系统总体设计

在数据分析层,采用了跨平台的Java开发语言,使用开源框架SparkMLlib框架进行数据分析处理.在数据接口层与数据展示层,采用PHP语言进行开发,采用开源框架Larvel+Swoole+Nginx.通过Nginx进行负载均衡<sup>[4]</sup>.采用关系型数据库MySQL和非关系型数据库Redis.MYSQL数据库用来存储结构化数据,包括

采集的餐饮等数据,而 Redis 用来存储通过微信小程序智能助手上传的包含用户个性化特点的非结构化数据. 开发工具采用 PhpStorm 和 IntelliJ IDEA 作为主要的开发工具. 系统功能架构如图 2 所示.

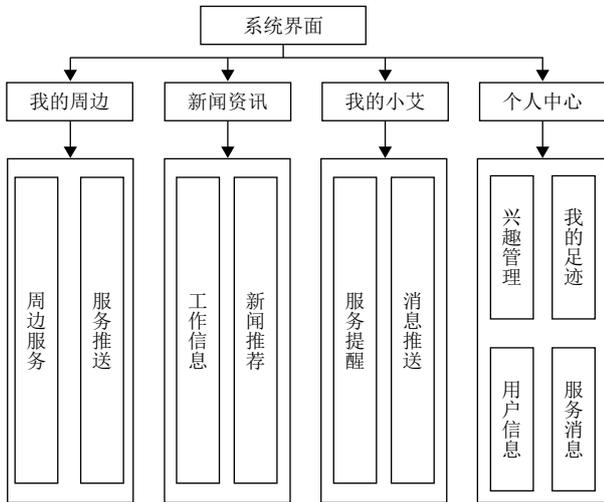


图 2 功能架构图



图 3 个人中心

## 2.2 功能详细设计与说明

### 2.2.1 个人中心

微信小程序智能助手基于微信的开放平台接口,用户可授权微信序获取个人信息,即可完成登录. 在用户注册完成后,可以获取到用户的微信头像,微信昵称. 同时用户可以自己设置消息免打扰时间,用户兴趣爱好以及查看用户日常浏览记录等. 用户个人中心如图 3.

### 2.2.2 服务消息

基于用户在微信内产生的操作数据,同时基于当前用户所处的地理位置,使用协同过滤算法中的 UserCF 来实现个性化推荐并同时推荐数据实时推送给用户. 生活服务推荐数据如图 4 所示.

在工作日期间基于用户历史数据,系统会根据当前用户所处位置<sup>[5]</sup>,通过协同过滤算法,并结合与当前用户有相似饮食特点用户数据,来为用户推送用餐消息,同时基于用户的反馈情况,来不断提高推荐结果的准确度. 服务消息推送效果如图 5 所示.

临近下班时刻,系统会基于用户的乘车记录和相关的公交实时位置,为用户进行公交车班次提醒. 系统会根据用户所处位置,为用户推荐最佳出发时间和乘车线路,确保用户可以准时到达公交车站,解决在公交车站等待时间过长的问题. 微信小程序智能助手推送出行服务消息如图 6 所示.



图 4 生活服务推荐数据

消息推荐采用 Swoole 开源框架, 该框架是一款高性能网络通信框架. 可以基于 Websocket 进行数据推送, 避免了传统的 Ajax 轮询获取数据导致页面卡顿

和手机发热严重的问题. 该框架基于多线程, 其基于 epoll/kqueue 的 Multi-Reactor 模型可以充分利用服务器的多核 CPU 性能.



图5 微信小程序智能助手消息推送



图6 微信小程序智能助手出行消息推送

### 2.2.3 生活服务

基于用户在日常使用微信浏览、搜索、点击等数据信息, 将数据上传到数据分析后台, 通过分析用户数据, 利用协同过滤算法中的 ItemCF 进行信息推荐<sup>[6]</sup>. 在数据采集层采集包括美食、车站、超市、酒店等各类生活服务信息, 因此协同过滤算法会基于用户的个性化特点来推荐给用户相关生活服务信息, 节省大量用户筛选过滤信息的时间. 系统还会根据用户点击、浏览时间、搜索查询等数据来对推荐系统进行不断改进,

这样可以根据用户兴趣爱好随时调整推荐结果, 确保用户能够获取合适的服务消息. 实现界面如图7所示.



图7 生活服务推荐

### 2.2.4 新闻资讯

基于用户的兴趣爱好, 基于协同过滤算法来分析用户习惯并提供用户感兴趣的新闻内容. 系统还会根据用户在个人中心设置的个人兴趣, 结合用户地理位置信息, 为用户提供其有效的新闻资讯. 实现界面如图8.

## 3 系统关键技术分析

### 3.1 数据采集技术

推荐给用户的生活服务信息, 需要通过爬虫采集网络上的相关资讯和信息. 由于数据量较大, 且需要对多个平台进行数据采集, 涉及到多线程采集, 采集的数据需要进行预处理, 方便数据分析层进行分析使用. 因此采用 Python 的开源框架 Scrapy. 该框架是基于多线程的异步处理框架, 能够支持上千个 Url 地址进行并行采集, 降低对采集服务器的压力<sup>[7]</sup>. 该框架具有高可定制性, 可以基于不同的场景, 不同的数据结构来编写不同的采集规则进行处理, 在不同 Item Pipeline 中, 可以对不同层次的数据进行分析过滤. 在采集的过程中会自动进行去重操作, 保证采集的数据不会重复.



图8 新闻资讯推荐

### 3.2 大数据存储技术

在处理协同过滤系统产生的数据的过程中, 由于数据为非关系型结构类型, 因此传统的关系型数据库保存此类数据比较困难. 而且推荐数据要实时为用户推送, 因此从的 MySQL 数据库中读取速度太慢, 存在 IO 瓶颈. 因此采用 Redis 进行数据存储.

Redis 是一个开源的使用 ANSI C 语言编写基于内存的 Key-Value 数据库, 由 Vmware 支持开发<sup>[8]</sup>.

### 3.3 协同过滤算法

协同过滤这一概念首次于 1992 年由 Goldberg Nicols Oki 及 Terry 提出, 他们展示了一种新的推荐思想. 目前主要有两类协同过滤算法: 基于用户的协同过滤算法 UserCF 和基于项目的协同过滤算法 ItemCF.

UserCF 的基本思想为<sup>[9]</sup>: 要为 A 用户推荐物品, 首先需要计算其他用户与 A 用户的用户相似度, 如表 1 所示, 用户 C 与用户 A 的相似度最高, 因此会将用户 C 喜欢的商品推荐给用户 A.

表1 基于用户协同过滤算法

物品	用户			
	A	B	C	D
A	√		√	(推荐)
B		√		
C	√		√	√

ItemCF 的基本思想为: 计算物品 A, B, C 的相似

度, 得到物品 A 与物品 C 相似度高, 那么在用户 C 选取物品 A 后, 为将商品 C 推荐给用户 C. 如表 2 所示.

表2 基于物品协同过滤算法

物品	用户			
	A	B	C	D
A	√		√	√
B	√	√		√
C	√		√	(推荐)

相似度计算方法<sup>[10]</sup>则是将用户评分看作是  $n$  维空间上的向量, 如果用户没有对物品进行评分, 则用户对该物品的评分为 0, 用户间的相似性通过计算向量间余弦夹角. 用户  $i$  和用户  $j$  在  $n$  维物品空间上的评分分别表示为向量  $i, j$ , 则用户  $i$  和  $j$  之间的相似度  $\text{sim}(i, j)$

$$\text{为: } \sin(i, j) = \cos(i, j) = \frac{i \cdot j}{\|i\| \|j\|} = \frac{\sum_{c=1}^n R_{i,c} R_{j,c}}{\sqrt{\sum_{c=1}^n R_{i,c}^2} \sqrt{\sum_{c=1}^n R_{j,c}^2}}$$

分子为两个用户评分向量的内积, 分母为两个用户向量模的乘积. 其中  $R_{i,c}$  和  $R_{j,c}$  分别代表用户  $i$  和用户  $j$  对物品  $c$  的评分.

基于微信的实时定位的特点可以方便的获取用户的地理位置. 因此我们在协同过滤模型中增加地理位置, 在计算用户之间相似度的时把地理位置作为一个距离分布的参数数据来进行分析, 从而使得最后的推荐数据忽略掉距离位置较远数据, 提高推荐算法的准确性<sup>[11]</sup>.

### 3.4 Websocket 网络通信技术

当用户在使用微信小程序智能助手过程中, 系统需要将推荐的信息推送至用户手机. 传统的消息推送是采用 Ajax 进行长连接轮询方式, 该方式采用了请求/响应模型, 通信请求只能由客户端发起, 服务端对请求作出应答处理. 但是这种通信模型有一个弊端: HTTP 协议无法实现服务器主动向客户端发起消息, 而且频繁的 Ajax 轮询导致效率低下, 浪费资源, 并且使得手机耗电量迅速增加.

采用 Websocket 网络通信技术可以解决上述问题. Websocket 连接允许客户端和服务端进行双工通信, 以便任一方可以通过建立的连接将数据推送到另一端. 同时该连接只要建立一次, 就可以一直保持连接状态, 通信效率大大提高, 降低手机的耗电量. 采用基于 Swoole 的 Websocket 框架可以实现通过异步非阻塞、多线程模式进行消息推送<sup>[12]</sup>.

## 4 系统运行结果分析

微信小程序智能助手平稳运行半年, 累积采集数据 7300 条有效数据, 采集数据包括车站信息, 新闻资讯, 餐饮超市等. 数据采集结果如图 9 所示.

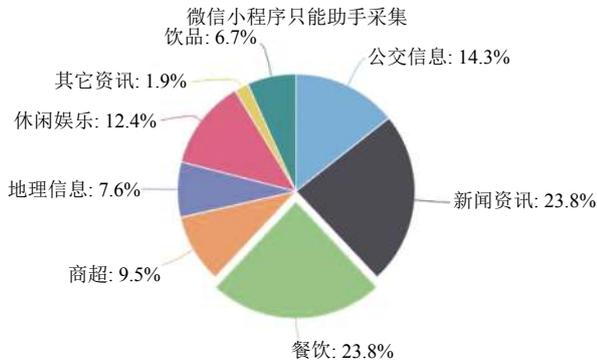


图 9 数据采集结果

微信小程序智能助手为用户生成有效推荐结果大约 6000 次, 其中新闻推荐 3000 多次, 餐饮推荐数据 1600 余次, 公交服务推荐 814 次, 其他相关生活服务推荐约 500 次. 在用户使用过程中, 用户有效反馈数据大约为 3300 次, 通过有效反馈数据, 最终推荐结果提高了 30% 左右. 并且通过用户的评价系统, 推荐结果满意度达到 85%. 运行结果数据分析如图 10 所示.

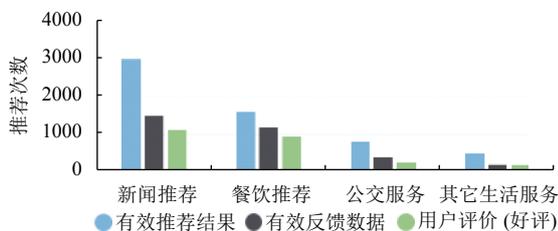


图 10 运行数据分析

## 5 总结

随着互联网的不断发展, 产生的数据信息将会越来越多. 如何能更加高效的分析用户的兴趣爱好和行为习惯, 并基于此为用户从海量的数据中推荐用户最感兴趣的信息将变得更加重要. 而且随着移动设备的不断普及, 用户从传统的 PC 开始不断向移动终端转移, 利用移动终端的快速地理定位、频繁的数据信息交互等特性中充分挖掘用户数据, 分析出有价值的信息, 从而为用户提供更加精准有效的推荐数据.

本文针对都市白领人群对于生活服务信息的需求进行了设计和实现, 在系统设计中遇到的问题进行积

极探索并找到合适的解决方案, 包括数据的采集、数据的分析、数据的存储以及数据的推送的稳定性等.

通过对基于协同过滤算法的微信小程序智能助手的开发与研究, 实现了对资源数据的采集, 利用协同过滤算法挖掘出用户的兴趣爱好并为用户推荐用户感兴趣的数据, 保证用户能够随时获取最新的、最符合用户兴趣的数据, 并且基于用户的日常行为对用户进行合理提醒与建议; 通过研究数据网络通信机制, 为用户提供稳定和高效的数据推送, 减低用户手机电量的消耗. 在开发过程中, 深深体会到了要合理充分挖掘用户行为习惯, 利用协同过滤算法不断优化推荐结果, 实现企业获取稳定用户和流量的重要保障.

## 参考文献

- 许婉韵. 关于微信小程序与原生 APP 使用偏好性的研究. 农家参谋, 2018, (21): 216-217.
- Xiao T, Shi YJ. Medical health data analysis based on spark MLlib. International Conference on Artificial Intelligence and Engineering Applications. Hong Kong, China. 2016. 116-125.
- 周绍景, 应杰, 潘宏斌, 等. RESTful 架构的应用研究. 数字技术与应用, 2018, 36(5): 59-60.
- 刘卓, 张向利. 基于 Nginx 的负载均衡集群设计与实现. 桂林电子科技大学学报, 2017, 37(6): 490-493. [doi: 10.3969/j.issn.1673-808X.2017.06.012]
- 郭兰杰, 梁吉业, 赵兴旺. 融合社交网络信息的协同过滤推荐算法. 模式识别与人工智能, 2016, 29(3): 281-288.
- 王晨阳, 刘垣, 郭李华, 等. 融合位置相似性度量的快消品电商网站推荐算法. 福建工程学院学报, 2017, 15(6): 586-590. [doi: 10.3969/j.issn.1672-4348.2017.06.016]
- 邓万宇, 刘光达, 董莹莹. 一种基于 Scrapy-Redis 的分布式微博数据采集方案. 信息技术, 2018, (11): 59-62. [doi: 10.3969/j.issn.1674-2117.2018.11.021]
- 朱亚兴, 余爱民, 王夷. 基于 Redis+MySQL+MongoDB 存储架构应用. 微型机与应用, 2014, 33(13): 3-5, 9. [doi: 10.3969/j.issn.1674-7720.2014.13.002]
- 冯宇, 李爱萍, 段利国. 融合社交关系和位置影响的地点推荐算法. 计算机工程与设计, 2018, 39(9): 2934-2940.
- 李晓瑜. 协同过滤推荐算法综述. 商丘师范学院学报, 2018, 34(9): 7-10. [doi: 10.3969/j.issn.1672-3600.2018.09.002]
- Sun SH, Kong GS, Zhao CW. Collaborative filtering methods based on user relevance degree and weights of recommend-items. Proceedings of 2011 International Conference on Multimedia Technology. Hangzhou, China. 2011. 12314458.
- 王佃来, 宿爱霞, 安晏辉. 基于 WebSocket 的消息推送系统. 计算机系统应用, 2017, 26(9): 87-92. [doi: 10.15888/j.cnki.csa.005973]