这个程序用Mysql数据库，php，实现需要送货服务的客户的信息发布，送货人的接单，并且运用，基于SVD隐语义模型智能推荐算法，基于送货人喜好进行货物类别推荐，优化了用户体验。

使用Mysql数据库，运用了六张表，建表语句为：

//建立用户表

create table users(

id int auto\_increment comment'内部ID可用来计数',

user\_name varchar(20) not null comment '微信昵称',

gender varchar(3) comment '性别',

tele varchar(15) not null comment '电话号码' ,

credit int not null comment '信用星级',

addtime varchar(20) not null comment '添加时间',

convey\_success int not null comment '匹配成功次数',

primary key(id)

)charset=utf8;

//创建物品类别表

create table items(

id int auto\_increment comment'物品类别ID可用来计数',

item\_class varchar(20) not null comment'物品类别',

primary key(id)

)charset=utf8;

//未接收订单表

create table unreceive(

user\_name varchar(20) not null comment '发货人微信昵称',

item\_name varchar(20) not null comment '物品名称',

item\_class varchar(20) not null comment '物品类别',

time\_start varchar(20) not null comment '期待发货时间',

time\_end varchar(20) not null comment '期待到达时间',

send\_address varchar(40) not null comment'发送地址',

get\_address varchar(40) not null comment'接受地址',

convey\_class varchar(10) not null comment'快件种类' ,

weight float not null comment '重量',

price float not null comment '价格'

)charset=utf8;

//已接收未完成订单表

create table received\_uncover(

user\_name varchar(20) not null comment '发货人微信名称',

convey\_name varchar(20) not null comment '送货人微信名称',

item\_name varchar(20) not null comment '物品名称',

item\_class varchar(20) not null comment '物品类别',

time\_start varchar(20) not null comment '实际发货时间',

time\_end varchar(20) not null comment '期待到达时间',

send\_address varchar(40) not null comment'发送地址',

get\_address varchar(40) not null comment'接受地址',

convey\_class varchar(10) not null comment'快件种类' ,

weight int not null comment '重量',

price float not null comment '价格'

)charset=utf8;

//已完成订单表

create table completed(

user\_name varchar(20) not null comment '发货人微信名称',

convey\_name varchar(20) not null comment '送货人微信名称',

item\_name varchar(20) not null comment '物品名称',

item\_class varchar(20) not null comment '物品类别',

time\_start varchar(20) not null comment '期待发货时间',

time\_end varchar(20) not null comment '实际到达时间',

send\_address varchar(40) not null comment'发送地址',

get\_address varchar(40) not null comment'接受地址',

convey\_class varchar(10) not null comment'快件种类' ,

weight int not null comment '重量',

price float not null comment '价格'

)charset=utf8;

//物品：物品类别，运送物品的用户，评分，对该类物品评分次数

create table useritem(

convey\_name varchar(20) not null comment'用户微信昵称',

name\_ID int not null comment'用户ID',

item\_class varchar(20) not null comment'物品类别',

item\_ID int not null comment'物品类别ID',

stars int not null comment'物品评价星级',

stars\_num int not null comment'给该类物品评分次数'

)charset=utf8;

/\*

用品表:

手机数码，

鞋包

饰品

户外运动

母婴童装

服装

家装家纺

食品保健

家电

美妆

\*/

对于数据库调用，用了两种技术：

对于 未接收订单表，已接收未完成订单表，已完成订单表， 使用PHP进行处理，并且由于前端平台为微信小程序，其对于后台调用要求较高，需要用HTTPS协议，并且需要服务器挂载有域名，所以后台处理均使用自己写的API，后台接收前端数据后判断需求，然后返回前端需要的数据（当然，返回为一个JSON对象）

对于 用户表 ， 用户评价表，物品表 ， 使用FLASK技术进行处理，这是由于FLASK框架基于python实现，而python拥有更好的大数据处理能力，这样对于我们实现基于隐语义模型的智能推荐算法有很大的便利。同样，我们用FLASK也需要将其做成一个存在服务器端的API，原因与上方的PHP相同，此不赘述。我们自己实现的这个基于隐语义模型的智能推荐算法将在下面详细讲解。

基于隐语义模型的协同过滤推荐算法：

数据集中行代表用户user，列代表物品item，其中的值代表用户对物品的打分。基于SVD的优势在于：用户的评分数据是稀疏矩阵，可以用SVD将原始数据映射到低维空间中，然后计算物品item之间的相似度，可以节省计算资源。

整体思路：先找到用户没有评分的物品，然后再经过SVD“压缩”后的低维空间中，计算未评分物品与其他物品的相似性，得到一个预测打分，再对这些物品的评分从高到低进行排序，返回前N个物品推荐给用户。

具体代码如下，主要分为5部分：

第1部分：加载测试数据集；

第2部分：定义三种计算相似度的方法；

第3部分：通过计算奇异值平方和的百分比来确定将数据降到多少维才合适，返回需要降到的维度；

第4部分：在已经降维的数据中，基于SVD对用户未打分的物品进行评分预测，返回未打分物品的预测评分值；

第5部分：产生前N个评分值高的物品，返回物品编号以及预测评分值。

优势在于：用户的评分数据是稀疏矩阵，可以用SVD将数据映射到低维空间，然后计算低维空间中的item之间的相似度，对用户未评分的item进行评分预测，最后将预测评分高的item推荐给用户。

其中，对于K的确立，我们从PCA（主成分分析）中获得启示，参照SVD中的sigma矩阵规模，按照前k个奇异值的平方和占总奇异值的平方和的百分比来确定k，并且，前10%甚至1%的奇异值的和就占了全部的奇异值之和的99%以上了。，所以。我们这里选了0.8作为度量（越大算法运行越慢，但是效果越好，我们需要做一个取舍，取得最大性价比）。