代码阅读说明

1. 数据处理部分：
2. 用户评分处理：
   1. 用户评分处理的热度heat采用的是kgdata\_document\_topn\_count表中的count字段，字段count=readCount+likeCount，即点赞数量和喜爱数量的和作为热度。

FAQ:

1. kgdata\_document\_topn\_count用户的阅览数目和点赞数目怎么来的？（问自己，问代码）
2. ~~评分机制怎么来的？~~

~~答：在rs.py文件中的process\_rating\_data函数中。~~

1. 用户画像部分：
   1. 计算用户画像
2. 代码流程图

参见processon

1. 推荐流程
2. 冷启动三种方法：
   1. 推荐热门的item给user，获取用户的反馈
   2. 获取user的个人信息，基于相似用户进行推荐
   3. 获取user的喜好信息，基于相似item进行推荐

2、

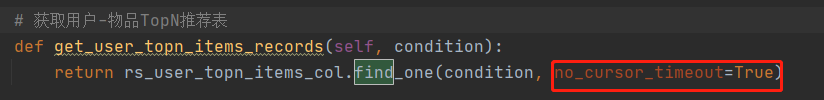
1. 一些设计细节：
2. 用户和文档之间有read、like等各种记录的情况的前提是用户点击过这个文档，所以只要用户记录中有userId\_itemId的记录，则说明用户点击了该文档。

附表：

FAQ Debug记录：

1. Q：评分表抓取mongodb超时问题？

A：加入no\_cursor\_timeout=True

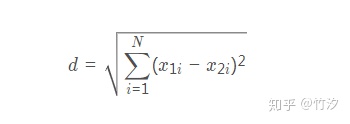


1. Q：相似度计算的方式有哪些？

A：根据数据类型定，如果计算两个集合的相似度，就用jaccard距离。首先，求两个集合的交集/两个集合的并集得到jaccard系数x，然后用1-x得到jaccard距离。

如果计算两个向量的相似度，就用余弦距离，即求两个向量的1-cosθ，用两个向量的内积除以两个向量的模的积得到余弦相似度，再用1减去余弦相似度，得到余弦距离。

和余弦相似度类似，如果是计算空间中两个点的相似度，用欧氏距离，即计算多个维度差异的平方和的平方根。



如果 ，用Pearson系数，用两个变量X和Y的协方差除以两个变量各自的方差，



Pearson系数的相关链接：https://blog.csdn.net/huangfei711/article/details/78456165

1. 代码部署在哪台服务器上？

只在128上部署了。

1. 评价的指标除了RMSE和MAE外，还用了其他评价指标吗？比如hit，ndcg

还用了召回率，其他没有了

1. RMSE和MAE的值均为0.5，怎么理解？

就是正常的计算方式，分数是0.5是计算结果，是否做过归一化，不清楚，具体看一下代码。

1. 文档设计中，看都了有召回模型和排序模型，但是代码中只找到召回模型，排序模型大概的位置没找到0.0

确实没有排序模型，召回模型用的是SVD模型。

1. 目前的模型是不是只包括召回模型的SVD模型，不包括其他模型。

大数据量的模型训练问题待解决。采用的是全量的矩阵，

1. 基于user和基于item的都有，推荐是单独的吗，还是合并结果后进行推荐，有实验测试结果吗，有的话我就不重复造轮子了，哈哈哈

对用户推荐文档是user-based，其他两种方式采用的是item-based，没有混合，都是单独的。

1. 流程图画得是否正确，

流程图没有问题。

1. 确定一个benchmark，然后才能在此基础上优化

benchmark没有，因为之前是9月份去过一次部署，其实和客户的交流并不多。

1. 看到说有存索引文件，索引文件主要是做什么的
2. 之前的用户浏览、点按记录如何获得的，是采用的豆瓣数据吗

是的，这些记录都是豆瓣影评的结果。

1. 代码中的系统冷启动如何进行，HGC的冷启动我考虑的是新进的用户先填写部门信息，先根据部门推荐相关文档。

基于新文档和基于新用户的冷启动基本用的是常规的方式。

基于系统的冷启动可以先用PageRank或者其他的一些基于内容的推荐方式进行。

想到的改进：

1. java端每天用模型进行预测，实时得到推荐列表，并采集用户行为，传到指定表中，供推荐使用。
2. nlp端每天拿到对应的数据进行模型训练，得出新模型发布到到接口处，供java端调用