协同过滤 —— 经典的推荐算法

- 概述
  - 定义
    - 协同大家的反馈、评价和意见一起对海量的信息进行过滤，从中筛选出目标用户可能感兴趣的信息的推荐过程
  - 共现矩阵
  - 优缺点
    - 优点
      - 非常直观、可解释性很强
    - 缺点
      - 不具备较强的泛化能力
- 用户相似度计算
  - 余弦相似度 (Cosine Similarity)
  - 皮尔逊相关系数
    - 通过使用用户平均分对各独立评分进行修正，减小了用户评分偏置的影响
  - 基于皮尔逊相关系数的思路，通过引入「物品平均分」的方式，减少物品评分偏置对结果的影响
- 最终结果的排序
  - 假设
    - 目标用户与其相似用户的喜好是相似的，可根据相似用户的已有评价对目标用户的偏好进行预测
  - User-based CF，基于「用户」的 CF
    - 利用「用户相似度」和相似用户的评价的加权平均获得目标用户的评价预测
    - 最常用的方式
    - 缺点
      - 1、在互联网场景下，用户数往往远大于物品数，导致用户的相似度矩阵的存储开销非常大
      - 2、用户的历史数据往往非常稀疏，对于只有几次购买行为或点击行为的用户来说，找到相似用户的准确度是非常低的
  - Item-based CF，基于「物品」的 CF
    - 基于「物品相似度」进行推荐的协同过滤算法
    - 流程
      - 1、基于历史数据，构建以用户为行坐标，物品为列坐标的 m x n 维的共现矩阵
      - 2、计算共现矩阵两两列向量间的相似性，构建 n x n 维的物品相似度矩阵
      - 3、获得用户历史行为数据中的「正反馈」物品列表
      - 4、利用物品相似度矩阵，针对目标用户历史行为中的正反馈物品，找出相似的 Top k 个物品，组成相似物品集合
      - 5、对相似物品集合中的物品，利用相似度分值进行排序，生成最终的推荐列表
- User-CF 与 Item-CF 的应用场景
  - User-CF
    - 相比于 Item-CF，具备更强的社交特性
      - 用户能够快速得知与自己兴趣相似的人最近喜欢的是什么。即使某个兴趣点以前不在自己的兴趣范围内，也有可能通过「朋友」的动态快速更新自己的推荐列表
    - 非常适用于「新闻推荐」场景
    - 适用于发现热点，以及跟踪热点的趋势
  - Item-CF
    - 更适用于「兴趣变化较为稳定」的应用
      - 电商推荐，用户在一个时间段内更倾向于寻找一类商品
      - 视频推荐，用户观看电影、电视居的兴趣点往往比较稳定
- 协同过滤的下一步发展
  - 优点
    - 非常直观、可解释性很强
  - 缺点
    - 仅利用用户和物品的交互信息，无法有效地引入用户年龄、性别、商品描述、商品分类、当前时间等一系列用户特征、物品特征和上下文特征
      - 逻辑回归，LR
    - 不具备较强的泛化能力
      - 矩阵分解
        - 在 CF 共现矩阵的基础上，使用更稠密的隐向量表示用户和物品，挖掘用户和物品的隐含兴趣和隐含特征
  - 协同过滤无法将两个物品相似这一信息推荐到其他物品的相似性计算上
    - 导致
      - 「热门的物品」具有很强的头部效应，容易跟大量物品产生相似性
      - 「尾部的物品」由于特征向量稀疏，很少与其他物品产生相似性，导致很少被推荐
  - 协同过滤的「天然缺陷」
    - 推荐结果的头部效应较明显
    - 处理稀疏向量的能力弱