

推荐算法——基于协同过滤CF

原创 努力给自己看 码小白TM 2019-01-06

前边我们已经介绍了推荐算法里的基于内容的推荐算法CB，今天我们来介绍一下基于协同的推荐算法Collaboration Filtering（CF）。协同过滤是一个利用群体智慧的一个算法，举一个简单的例子，如果放假了你想去北京玩几天，但是不知道北京都有哪些好玩的，你会怎么做？你可能会上网查一下网友的推荐，也可能会咨询一下曾经去过北京的亲朋好友，最后你很可能会采纳你好友的推荐。因为你的好朋友和你相似的爱好和品味。

基于协同的推荐算法又分为两种

- User Based CF

基于当前用户对物品的偏好找到相似偏好的用户，然后把相似偏好用户喜欢的物品推荐给当前用户。例如下图，对于 user1，我们找到了和他有相似偏好的 user3，他们都同时喜欢 Item1 和Item3，并且 user3 还喜欢 Item4，我们就可以把 Item4 推荐给 user1。

user/item	Item1	Item2	Item3	Item4
User1	✓		✓	
User2		✓		✓
User3	✓		✓	✓

- Item Based CF

原理和基于用户的类似，只是在计算时基于物品而不是用户，找到和当前用户喜欢的物品相似的物品，然后推荐给当前用户。例如下图，我们发现喜欢 Item1 的用户都喜欢 Item3，并且还有很多也喜欢 Item4，我们就认为 Item1、Item2 和 Item4 比较相似，所以我们就可以把 Item4推荐给 user1。

user/item	Item1	Item2	Item3	Item4
User1	✓		✓	
User2	✓	✓	✓	✓
User3	✓		✓	✓

努力给自己看

协同过滤的实现，一般通过以下几个步骤

- 收集用户对物品的偏好数据

通常把用户对物品的行为数据收集起来，然后根据不同的行为数据来标志用户对该物品的喜欢程度，然后根据喜欢程度给物品一个打分。比如说一个电商网站，根据用户的浏览，收藏，加购物车，购买，评论等行为数据来进行打分，标志用户对这个物品的喜爱程度。

- 通过计算找到相似的用户或物品

然后像上边举得两个例子那样，通过一些算法，来找到和该用户有相似偏好的用户，或者找到和该用户喜欢的物品相似的物品。

- 然后把相似度较高的n个结果推荐给用户

最后从得到的推荐列表里挑选分数较高的n个结果推荐给用户

基于用户和基于物品的协同过滤有各自的优点和不足，我们使用的时候要根据具体的场景和情况来选择合适的来使用。最后来比较一下两者的不同。

	user based	item based
性能	适用用户较少的场合，如果用户数量较多，计算用户相似矩阵代价太大	适用于物品数量明显小于用户数量的场合，如果物品很多，计算物品的相似矩阵代价太大
领域	时效性强，用户个性化兴趣不太明显的领域	长尾物品丰富，用户个性化需求较强的领域
实时性	用户有新行为，不一定造成推荐结果立即变化	用户有新行为，一定会导致推荐结果的变化
冷启动	在新用户对很少的物品产生行为后，不能马上对他进行个性化推荐，因为用户相似度表是每隔一段时间离线计算的。新物品上线后，一旦有用户对它产生行为，就可以把该物品推荐给和该用户有相似偏好的其他用户	新用户只要对一个物品产生行为，就可以给他推荐和该物品相似的其他物品，但是没办法在不离线更新新物品相似度表的情况下把新物品推荐给用户
推荐理由	很难提供令用户信服的推荐解释	利用用户的历史行为给用户做推荐解释，可以用户比较信服

基于协同的推荐算法也有自己的优缺点

优点	缺点
用户模型刻画用户兴趣需求	推荐精度低（仅仅依赖于物品分析，没有其他分析数据）
推荐形式多样性，具有个性化	马太效应（可能连环看一类的，只能推一类视频）
结果可解释	用户行为稀疏导致覆盖率低

如果觉得对你有帮助，可以点一下 好看，可以推荐和分享给你的朋友
「努力给自己看」