### 协同过滤算法

协同过滤简单来说是利用某[兴趣](https://baike.baidu.com/item/%E5%85%B4%E8%B6%A3/5720174" \t "https://baike.baidu.com/item/%E5%8D%8F%E5%90%8C%E8%BF%87%E6%BB%A4/_blank)相投、拥有共同经验之群体的喜好来推荐用户感兴趣的[信息](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF/111163" \t "https://baike.baidu.com/item/%E5%8D%8F%E5%90%8C%E8%BF%87%E6%BB%A4/_blank)，个人通过合作的机制给予信息相当程度的回应（如评分）并记录下来以达到过滤的目的进而帮助别人[筛选](https://baike.baidu.com/item/%E7%AD%9B%E9%80%89/10883707" \t "https://baike.baidu.com/item/%E5%8D%8F%E5%90%8C%E8%BF%87%E6%BB%A4/_blank)信息，回应不一定局限于特别感兴趣的，特别不感兴趣信息的纪录也相当重要。

## ****基于用户的协同过滤算法（UserCF）****

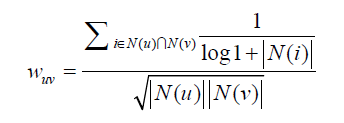
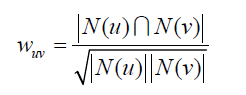
算法核心：当一个用户A需要个性化推荐时，可以先找到他有相似兴趣的其他用户，然后把那些用户喜欢的、而用户A没听过的物品推荐给A。

a) 找到和目标用户兴趣相似的用户集合

b) 找到这个集合中的用户喜欢的，且目标用户没有听说过的物品推荐给目标用户。

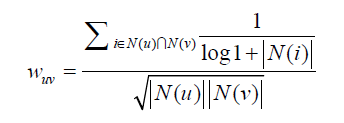
步骤(a)的关键就是计算两个用户的兴趣相似度。这里，协同过滤算法主要利用行为的相似度计算兴趣的相似度。给定用户u和用户v，令N(u)表示用户u曾经有过正反馈的物品集合，令N(v)为用户v曾经有过正反馈的物品集合。那么，我们可以通过如下的Jaccard公式简单地计算u和v的兴趣相似度或者通过余弦公式：

             jaccard                                                     余项公式：

[](https://img2018.cnblogs.com/blog/918077/201811/918077-20181103141813841-55180140.png)            [](https://img2018.cnblogs.com/blog/918077/201811/918077-20181103141814408-1227794214.png)

上述算法很简单但是计算量较大，因为需要所有用户之前的复杂度  n（n-1）/2。

计算用户相似度的改进算法：

[](https://img2018.cnblogs.com/blog/918077/201811/918077-20181103141555872-1924358063.png)

**分子中的倒数惩罚了用户u和用户v共同兴趣列表中热门物品对他们相似度的影响**。N(i)是对物品i有过行为的用户集合，越热门，N(i)越大