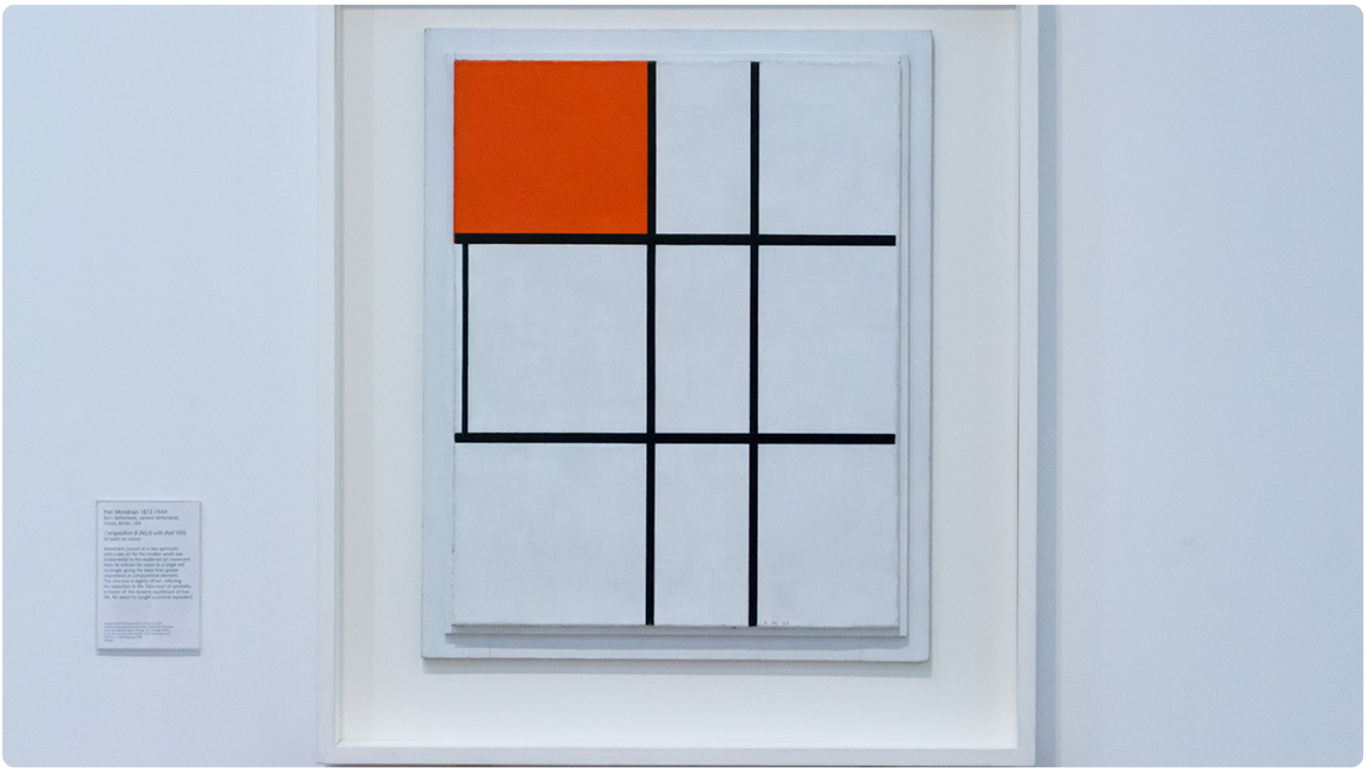
064 | 简单推荐模型之二：基于相似信息的推荐模型

2018-02-28 洪亮劼

AI技术内参 进入课程 >



讲述：初明明
时长 08:04 大小 3.70M

这周我们开始讲推荐系统。周一的文章中，我们聊了一个最基本的推荐模型：基于流行度的推荐模型。这是一种简单且实用的推荐系统搭建方式，那就是需要对每一个物品的流行度进行估计。

今天，我们来看另外一种简单但很有效果的推荐模型：**基于相似信息的推荐模型。**

什么是相似信息的推荐模型

相似信息的推荐模型又叫**“临近”（Neighborhood）模型。**顾名思义，就是我们希望利用临近、或者相似的数据点来为用户推荐。

临近模型的内在假设是推荐系统中著名的“**协同过滤**”（Collaborative Filtering）。什么意思呢？就是说，我们认为，**相似的用户可能会有相似的喜好，相似的物品可能会被相似的人所偏好。**于是，如果我们能够定义怎么寻找相似的用户或者相似的物品，那么我们就可以利用这些类别的人群或者物品来给用户进行推荐。

例如，对于一个电影推荐的场景来说，有一个用户 A 观看了电影《战狼 2》，我们希望根据这个信息来为用户进行推荐。很显然，如果我们仅仅知道用户 A 观看过《战狼 2》，这个信息是非常有限的。但是，假设有一个用户 B 也观看过《战狼 2》，并且最近还观看过《红海行动》。那么， 我们可以根据 B 的信息来对 A 进行推荐，也就是说，我们认为用户 A 也有可能喜欢《红海行动》。

这里面，我们其实经历了这么两个步骤。

第一，联系用户 A 和用户 B 的是他们都看过《战狼 2》。这就帮助我们定义了 A 和 B 是相似用户。

第二，我们的假定是，相似的用户有相似的观影偏好，于是我们就直接把 B 的另外一个观看过的电影《红海行动》拿来推荐给了 A。

这两个步骤其实就很好地解释了“协同过滤”中“协同”的概念，意思就是相似的用户互相协同，互相过滤信息。

“协同过滤”从统计模型的意义上来讲，其实就是“借用数据”，在数据稀缺的情况下帮助建模。掌握这个思路是非常重要的建模手段。

在用户 A 数据不足的情况下，我们挖掘到可以借鉴用户 B 的数据。因此，我们其实是把用户 A 和用户 B “聚类”到了一起，认为他们代表了一个类型的用户。当我们把对单个用户的建模抽象上升到某一个类型的用户的时候，这就把更多的数据放到了一起。

基于相似用户的协同过滤

刚才我们简单聊了聊什么是基于相似信息的推荐系统。相信到现在，你已经对这个概念有了一个最基本的认识。

那么，如何才能够比较系统地定义这样的流程呢？

首先，问题被抽象为我们需要估计用户 I 针对一个没有“触碰过”（这里指点击、购买、或者评分等行为）的物品 J 的偏好。

第一步，我们需要构建一个用户集合，这个用户集合得满足两个标准：第一，这些用户需要已经触碰过物品 J，这是与用户 I 的一大区别；第二，这些用户在其他的行为方面需要与用户 I 类似。

现在我们假设这个集合已经构造好了。那么，接下来的一个步骤，就是根据这个相似的用户集，我们可以对物品 J 进行一个打分。这个打分的逻辑是这样的。首先，我们已经得到了所有和 I 相似的用户对 J 的打分。那么，一种办法就是，直接用这些打分的平均值来预估 J 的评分。也就是说，如果我们认为这个相似集合都是和用户 I 相似的用户，那么他们对 J 的偏好，我们就认为是 I 的偏好。显然这是一个很粗糙的做法。

我们可以针对这个直接平均的做法进行两个改动。

第一，采用加权平均的做法。也就是说，和用户 I 越相似的用户，我们越倚重这些人对 J 的偏好。

第二，我们也需要对整个评分进行一个修正。虽然这个相似集合的人都对 J 进行过触碰，但是每个人的喜好毕竟还是不一样的。比如有的用户可能习惯性地会对很多物品有很强的偏好。因此，仅仅是借鉴每个人的偏好，而忽略了这些用户的偏差，这显然是不够的。所以，我们需要对这些评分做这样的修正，那就是减去这些相似用户对所有东西的平均打分，也就是说，我们需要把这些用户本人的偏差给去除掉。

综合刚才说的两个因素，可以得到一个更加合适的打分算法，那就是，用户 I 对物品 J 的打分来自两个部分：一部分是 I 的平均打分，另外一部分是 I 对于 J 的一个在平均打分之上的补充打分。这个补充打分来自于刚才我们建立的相似用户集，是这个相似用户集里每个用户对于 J 的补充打分的一个加权平均。权重依赖于这个用户和 I 的相似度。每个用户对于 J 的补充打分是他们对于 J 的直接打分减去他们自己的平均打分。

相似信息的构建

我们刚才讲了一下相似用户协同过滤的一个基本思路。那么，这里面有几个要素需要确定。

第一，我们怎么来定义两个用户是相似的？一种最简单的办法，就是计算两个用户对于他们都偏好物品的“**皮尔森相关度**”（Pearson Correlation）。这里当然可以换做是其他相关

信息的计算。

具体来说，皮尔森相关度是针对每一个“两个用户”都同时偏好过的物品，看他们的偏好是否相似，这里的相似是用乘积的形式出现的。当两个偏好的值都比较大的时候，乘积也就比较大；而只有其中一个比较大的时候，乘积就会比较小。然后，皮尔森相关度对所有的乘积结果进行“加和”并且“归一化”。

第二，当有了用户之间的相关度信息后，我们可以设定一些“阈值”来筛选刚才所说的相关用户集合。对于每个目标用户，我们可以设置最多达到前 K 个相似用户（比如 K 等于 100 或者 200），这也是有效构造相似集合的办法。

最后，我们来谈一下刚才所说的加权平均里面的权重问题。一种权重，就是直接使用两个用户的相似度，也就是我们刚计算的皮尔森相关度。当然，这里有一个问题，如果直接使用，我们可能会过分“相信”有一些相关度高但自身数据也不多的用户。什么意思呢？比如有一个用户 M，可能和目标用户 I 很相似，但是 M 自己可能也就偏好过一两件物品，因此我们可能还需要对相关度进行一个“**重新加权**”（Re-Weighting）的过程。具体来说，我们可以把皮尔森相关度乘以一个系数，这个系数是根据自身的偏好数量来定的。

基于相似物品的协同过滤

在协同过滤的历史上，人们很快就意识到在进行构建推荐的过程中，用户和物品的“对称性”。什么意思？也就是说，我们刚才对于用户的讨论其实完全可以变换到物品中。

具体说来，那就是我们不去构造和用户 I 相似的用户，而是去构造所有和物品 J 相似的物品。这些相似物品集合必须要满足两点：第一，和 J 相似；第二，已经被用户 I 触碰了。这里的一个基本的假设类似于，虽然我不知道用户 I 对于《红海行动》的偏好，但我可以去看一看用户过去看的电影里有哪些和《红海行动》是类似的，我们就可以从那些类似的电影中进行加权平均，取得对《红海行动》的预测。

小结

今天，我为你讲了推荐系统的另外一个基本的形式：基于相似度的协同过滤推荐系统。

一起来回顾下要点：第一，我们简要介绍了整个基于相似度协同过滤的内涵以及我们这么做的基本假设；第二，我们详细介绍了如何构造一个基于用户相似度的协同过滤系统；第三，我们简要地提及了如何构造物品相似的协同过滤系统。

最后，给你留一个思考题，协同过滤的一个致命问题是什么？

欢迎你给我留言，和我一起讨论。

上一篇　063 | 简单推荐模型之一：基于流行度的推荐模型

下一篇　065 | 简单推荐模型之三：基于内容信息的推荐模型

精选留言 (2)

写留言

林彦
2018-03-02

3

谢谢洪老师的分享。

按照网上的信息，这里列一下传统的协同过滤常见问题
稀疏性（Sparsity）问题：用户和项目的数量非常大时评分矩阵会极度稀疏，对算法的效率产生消极影响；同时由于这个问题的存在，两个用户的之间的相似度很有可能为零，产...

展开

Peter
2018-02-28

2

冷启动阶段啥用户行为不够多的时候，感觉会有问题

展开