# 推荐系统 [1] [2]

leolinuxer

July 14, 2020

## Contents



# 1 概述

2 种简单的推荐方法：

- 非定制的推荐系统：简单来说就是，什么最热卖，什么关注的人多，就推荐你什么；
- 定制化的推荐系统：针对用户、内容等进行定制化推荐

# 2 定制化推荐系统的推荐方法

定制化的推荐系统里面常用的方法，一般常用的有两大类。

- 协同过滤（collaborative filtering）；
- 基于内容的推荐 (content-based recommendation)：基于内容的推荐大致是，我看了一篇关于足球的报道，之后又向我推荐了足球的相关报告。里面用的技术就是基于内容的推荐。

# 3 协同过滤

**协同过滤这个算法，目的就是找相似**。其中：找相似，可以是找相似的人，也可以找相似的东西。主要分为三类：

- user-based；基于用户 (user-based) 的协同过滤主要考虑的是用户和用户之间的相似度，只要找出相似用户喜欢的物品，并预测目标用户对对应物品的评分，就可以找到评分最高的若干个物品推荐给用户。
- item-based；基于项目 (item-based) 的协同过滤和基于用户的协同过滤类似，只不过这时我们转向找到物品和物品之间的相似度，只有找到了目标用户对某些物品的评分，那么我们就可以对相似度高的类似物品进行预测，将评分最高的若干个相似物品推荐给用户。
- model based；

基于用户的协同过滤和基于项目的协同过滤的比较：基于用户的协同过滤需要在线找用户和用户之间的相似度关系，计算复杂度肯定会比基于基于项目的协同过滤高。但是可以帮助用户找到新类别的有惊喜的物品。而基于项目的协同过滤，由于考虑的物品的相似性一段时间不会改变，因此可以很容易的离线计算，准确度一般也可以接受，但是推荐的多样性来说，就很难带给用户惊喜了。一般对于小型的推荐系统来说，基于项目的协同过滤肯定是主流。但是如果是大型的推荐系统来说，则可以考虑基于用户的协同过滤，当然更加可以考虑我们的第三种类型，基于模型的协同过滤。

## 3.1 相似的度量方式——相似度

余弦相似度：

$$\cos(u_i, u_k) = \frac{\sum_{j=1}^{m} v_{ij} v_{kj}}{\sqrt{\sum_{j=1}^{m} v_{ij}^2 \sum_{j=1}^{m} v_{kj}^2}}$$

## 3.2 基于模型的协同过滤

基于模型的协同过滤作为目前最主流的协同过滤类型，其相关算法可以写一本书了，当然我们这里主要是对其思想做有一个归类概括。我们的问题是这样的 m 个物品，m 个用户的数据，只有部分用户和部分数据之间是有评分数据的，其它部分评分是空白，此时我们要用已有的部分稀疏数据来预测那些空白的物品和数据之间的评分关系，找到最高评分的物品推荐给用户。

对于这个问题，用机器学习的思想来建模解决，主流的方法可以分为：用关联算法，聚类算法，分类算法，回归算法，矩阵分解，神经网络, 图模型以及隐语义模型来解决。下面我们分别加以介绍。

### 3.2.1 用关联算法做协同过滤

一般我们可以找出用户购买的所有物品数据里频繁出现的项集活序列，来做频繁集挖掘，找到满足支持度阈值的关联物品的频繁 N 项集或者序列。如果用户购买了频繁 N 项集或者序列里的部分物品，那么我们可以将频繁项集或序列里的其他物品按一定的评分准则推荐给用户，这个评分准则可以包括支持度，置信度和提升度等。常用的关联推荐算法有 Apriori，FP Tree 和 PrefixSpan。

### 3.2.2 用聚类算法做协同过滤

用聚类算法做协同过滤就和前面的基于用户或者项目的协同过滤有些类似了。我们可以按照用户或者按照物品基于一定的距离度量来进行聚类。如果基于用户聚类，则可以将用户按照一定距离度量方式分成不同的目标人群，将同样目标人群评分高的物品推荐给目标用户。基于物品聚类的话，则是将用户评分高物品的相似同类物品推荐给用户。常用的聚类推荐算法有 K-Means, BIRCH, DBSCAN 和谱聚类等。

### 3.2.3 用分类算法做协同过滤

如果我们根据用户评分的高低，将分数分成几段的话，则这个问题变成分类问题。比如最直接的，设置一份评分阈值，评分高于阈值的就是推荐，评分低于阈值就是不推荐，我们将问题变成了一个二分类问题。虽然分类问题的算法多如牛毛，但是目前使用最广泛的是逻辑回归。为啥是逻辑回归而不是看起来更加高大上的比如支持向量机呢？因为逻辑回归的解释性比较强，每个物品是否推荐我们都有一个明确的概率放在这，同时可以对数据的特征做工程化，得到调优的目的。目前逻辑回归做协同过滤在 BAT 等大厂已经非常成熟了。常见的分类推荐算法有逻辑回归和朴素贝叶斯，两者的特点是解释性很强。

### 3.2.4 用回归算法做协同过滤

用回归算法做协同过滤比分类算法看起来更加的自然。我们的评分可以是一个连续的值而不是离散的值，通过回归模型我们可以得到目标用户对某商品的预测打分。常用的回归推荐算法有 Ridge 回归，回归树和支持向量回归。

### 3.2.5 用矩阵分解做协同过滤

用矩阵分解做协同过滤是目前使用也很广泛的一种方法。由于传统的奇异值分解 SVD 要求矩阵不能有缺失数据，必须是稠密的，而我们的用户物品评分矩阵是一个很典型的稀疏矩阵，直接使用传统的 SVD 到协同过滤是比较复杂的。

目前主流的矩阵分解推荐算法主要是 SVD 的一些变种，比如 FunkSVD，BiasSVD 和 SVD++。这些算法和传统 SVD 的最大区别是不再要求将矩阵分解为 $UV^T$ 的形式，而变是两个低秩矩阵 $P^TQ$ 的乘积形式。

#### 3.2.6 用神经网络做协同过滤

用神经网络乃至深度学习做协同过滤应该是以后的一个趋势。目前比较主流的用两层神经网络来做推荐算法的是限制玻尔兹曼机 (RBM)。在目前的 Netflix 算法比赛中，RBM 算法的表现很牛。当然如果用深层的神经网络来做协同过滤应该会更好，大厂商用深度学习的方法来做协同过滤应该是将来的一个趋势。

#### 3.2.7 用图模型做协同过滤

用图模型做协同过滤，则将用户之间的相似度放到了一个图模型里面去考虑，常用的算法是 SimRank 系列算法和马尔科夫模型算法。对于 SimRank 系列算法，它的基本思想是被相似对象引用的两个对象也具有相似性。算法思想有点类似于大名鼎鼎的 PageRank。而马尔科夫模型算法当然是基于马尔科夫链了，它的基本思想是基于传导性来找出普通距离度量算法难以找出的相似性。

#### 3.2.8 用隐语义模型做协同过滤

隐语义模型主要是基于 NLP 的，涉及到对用户行为的语义分析来做评分推荐，主要方法有隐性语义分析 LSA 和隐含狄利克雷分布 LDA。

## References

[1] [Online]. Available: https://zhuanlan.zhihu.com/p/69888124

[2] [Online]. Available: https://zhuanlan.zhihu.com/p/25069367