协同过滤

2016年4月19日 星期二　　下午10:04

一、简介

协同过滤是一种推荐算法，协同过滤推荐（Collaborative Filtering recommendation）。

假设推荐系统中有user、item这两种角色，我们现有信息是user对部分item的评价（打分）（很稀疏的矩阵），需要预测userA与itemB之间的评价（打分）。

在这样稀疏的观测数据中，利用user与item之间的关联关系，计算不同user之间、不同item之间的相关性，使用"与userA相似user对itemB的评价"，刻画userA对itemB的评价；使用"与itemB相同的item收到userA的评价"，刻画userA对itemB的评价。

二、原理思想

是一种非参数估计方法。设计思想是为了处理user-item矩阵稀疏的问题。

user和item可以看做地位基本相同，都是多组随机变量。

1、如何通过稀疏的观测数据，找到user之间、item之间相关性。

使用两个user之间pearson相关性系数，作为两个user相关性的表示；item同理

2、如何确定与userA相似的user

在所有user中取与userA的pearson系数最大的top N个；item同理

3、如何通过与相似user对itemB评价，刻画userA对itemB的评价

加权平均，权重就是userA和相似user的pearson系数

注意每个user对item的打分要做"归一化"，就是都减去打分的均

值，比如：使得（1，2，3），（4，5，6）是一样的。

三、算法

1、基于user的协同过滤

A 思想

step1：找到相似的user

计算userA与所有user的pearson，取top N

step2：预测对item的打分

使用这N个user对itemB的打分加权平均，得到userA对itemB的打分的预测

注意，要进行"归一化"，每个用户都减去自己评价的所有item的均值

B 公式：

变量定义：

要预测用户u对商品i的评分ru,i

用户u对所有商品的平均得分为$\bar{r_u}$

用户x评分的商品集合为Ix，用户y评分的商品集合为Iy，其并集为Ixy？（交集还是并集？应该是交集吧）

U是用户u的近邻，z是归一化因子

step1：pearson公式

$$sim(x,y)=\frac{\sum_{i\in I_{xy}}(r_{x,i}-\bar{r_x})(r_{y,i}-\bar{r_y})}{\sqrt{\sum_{i\in I_{xy}}(r_{x,i}-\bar{r_x})^2\sum_{i\in I_{xy}}(r_{y,i}-\bar{r_y})^2}}$$

step2：预测打分公式

$$r_{u,i}=\bar{r_u}+z\sum_{u'\in U}sim(u,u')(r_{u',i}-\bar{r_{u'}})$$

$$z=\frac{1}{\sum_{u'\in U}sim(u,u')}$$

2、基于item的协同过滤

A 思想（与基于user同理）

step1：找到相似的item

计算itemB与所有item的pearson（博客写计算余弦，应该不太对吧？？？），选取top M个item

step2：预测userA对itermB的打分

使用userA对这M个item的打分，加权平均，得到userA对itemB的打分的预测

这里不用归一化，因为没法做归一化

B 公式

step1：

$$sim(i,j) = \frac{\sum_x (r_{x,i} - \bar{r_x})(r_{x,j} - \bar{r_x})}{\sqrt{\sum_x (r_{x,i} - \bar{r_x})^2 \sum_x (r_{x,j} - \bar{r_x})^2}}$$

step2：

$$r_{u,i} = \frac{\sum_{i' \in N} sim(i,i') r_{u,i'}}{\sum_{i' \in N} sim(i,i')}$$

3、混合协调过滤

A 思想

a 基于user的协同过滤为基础，计算两个user相似度时，又嵌套了item-based CF的思想

b 计算两个userX、userY相似度时，在userX、userY的并集中，如果包含user没评价的item，先通过item-based CF方法预测item的打分。

四、代码

参考资料1的代码就很清晰

放到开发机上了

[tianzhiliang@cp01-rdqa-dev371.cp01.baidu.com collaborative_filtering]$

pwd
/home/users/tianzhiliang/tools/code_ml_dnn/collaborative_filtering

五、思考问题

为何以用户为基础？

是不是因为item没法归一化

相似的user也没有评价itemB怎么办？

看代码是直接把打分=0这个信息带进去计算了

userA没有预测相似的item怎么办？

看代码是直接把打分=0这个信息带进去计算了

六、参考资料

简单、具体、清晰，有代码
http://www.cnblogs.com/zhangchaoyang/articles/2664366.html
英文wiki https://en.wikipedia.org/wiki/Collaborative_filtering