

UserBasedSlopOne 算法实现

陈秋良 李正 卢晓燕

根据 A Creative Personalized Recommendation Algorithm——User-based Slope One Algorithm 这篇论文，使用 Matlab 编程完成。

主要算法分为 4 个步骤：

- 计算 Item 之间的平均评分差值矩阵 D ，并得到同时给 Item i 和 j 打分的用户个数矩阵 W
- 计算用户相似度矩阵 Sim ，根据余弦相似度
- 计算 S 矩阵， Sti 表示给 item i 打分的其余用户和用户 t 的平均相似度
- 计算用户对 item 的打分估计值。

$$P_{ti} = \frac{\sum_{R_{tc} \neq 0} (R_{tc} + D_{ic}) * S_{tc} * W_{ic}}{\sum_{R_{tc} \neq 0} S_{tc} * W_{ic}}$$

当然在实现的时候还需要考虑读取数据集等问题。

最重要的一点是，在 Matlab 中如何优化代码，提高运行效率。

在优化以前，由于整个算法的复杂度是 $O(m*n*n)$ -- m 为用户数， n 为物品数，在 matlab 中运行非常慢，因为直接写代码会有三重循环，基本上要 20 个小时左右才能出结果。

优化的方法是：

1. 利用 matlab 和 c 混合编程，将大量循环操作放到 c 中完成。这里主要利用 mex 函数接口。具体见代码。
2. 尽量用矩阵运算，转置、点乘等代替不必要的循环操作。
3. 计算 Sim 矩阵时候，利用 repmat 函数优化时间。

提交代码说明：

1. MovieLens.mat 是训练数据集，主要是 user-item 的评分矩阵。
2. MatrixData2.mat 是训练的结果，保存了上面的 R, D, W, Sim, S 矩阵。 -- 已删
3. UserBasedSlopeOne.m 是训练的代码，主要实现上面算法的四个步骤。
使用方法： $p = \text{UserBasedSlopeOne}(\text{userID}, \text{itemID}, \text{file})$; 给 user 和 item 的 ID，以及 mat 文件如 'MovieLens.mat'，自动训练并输出评分预测
4. UserBasedSlopeOne2.m 是利用训练好的 matrixData2.mat 的数据进行预测。
使用方法： $p = \text{UserBasedSlopeOne2}(\text{userID}, \text{itemID})$; 给 user 和 item 的 ID，输

出对应的评分预测

5. MAE 和 RMSE 文件，给定预测值和目标值，输出计算误差。
6. test.m 文件，调用 MAE 和 RMSE 计算误差。
7. dataset.mat 测试数据文件
8. GetTheAveSimMat.c、GetTheDiff&WeightMat.c 和 GetTheDiffMat.c 是优化的 c 代码。用来和 matlab 协作。
9. GetTheAveSimMat.mexw64、GetTheDiff&WeightMat.mexw64 和 GetTheDiffMat.mexw64 是 c 代码生成的 mex 函数，供 matlab 调用。

实验和测试：

利用训练好的数据进行预测，结果如下：

实际值	预测值
-----	-----

5.0000	5.0076
--------	--------

5.0000	5.0255
--------	--------

5.0000	5.0429
--------	--------

5.0000	4.9796
--------	--------

5.0000	5.0012
--------	--------

3.0000	3.2311
--------	--------

3.0000	3.1510
--------	--------

3.0000	2.9424
--------	--------

3.0000	3.0729
--------	--------

误差分析：

进行 100 个预测，与实际结果对比，计算误差：

```
mae =  
  
0.7422  
  
rmse =  
  
1.0389
```

结果表明该算法的有效性。

最后，感谢 TA 哥指导！

2015.6.1