

Model

我們主要參考了[1], [2]這兩篇paper裡的方法，利用orthogonal tri-factorization來同時對source domain的user和item分群，然後製作"codebook"並將其transfer到target domain。

流程大致如下：

使用libmf分解source，相乘之後做orthogonal tri-factorization[1]，並按照[2]的方法製作codebook B。之後實作[2]的Algorithm 2，即Codebook Transfer，得到新的target。

在過程中遇到的其中一個問題是source domain的rating matrix是sparse的，但在sparse的矩陣上做tri-factorization並不容易，因此我們先將矩陣用libmf分解再重新乘回來，得到一個dense的矩陣。

另外，我們發現在 $U^T 1 1^T V$ 中有許多的entry都是0，若是按照paper中的方法計算codebook B的話，會出現許多NaN值，因此我們在計算B時將分子分母都同加1，做一個簡單的smoothing。

在我們的model中，有一個需要調整的參數 k ，代表做matrix factorization時的dim，以及最後codebook的dimension($k \times k$)。實驗發現調大 k 可以讓performance更好，但會讓計算時間提昇。

Validation

我們對train.txt使用5-fold cross validation來evaluate結果。

Experiment results

task / performance	k	baseline (libmf)	Cross Validation
1	250	0.2115	0.1840
2	500	0.2128	0.1978
3	100	1.4416	1.3409

Discussion

由於[2]的Codebook Transfer方法是一般MF的transfer learning版本，所以baseline應該只需與libmf的結果比較就好。由上表可以看出進步顯著，表示有真的transfer一些source domain的資訊進來。

Contribution

B02902030 曹燁文: write report, write NMTF.py, think method, run task1, task2
B02705008 江昱熹: write report, write CBT.py, think method, run task3

Reference

- [1] Li, Bin, Qiang Yang, and Xiangyang Xue. "Can Movies and Books Collaborate? Cross-Domain Collaborative Filtering for Sparsity Reduction." IJCAI. Vol. 9. 2009
- [2] Ding, Chris, et al. "Orthogonal nonnegative matrix t-factorizations for clustering." Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006.