ASSIGNMENT 2

CS5304 - LARGE SCALE RECOMMENDATION SYSTEMS

1. Task

In this assignment, you'll be building a recommendation system. You'll be given data that comprises Users, Items, Ratings and Timestamps. For this assignment, we'll not be using the timestamps. We'll focus on the User-Item Utility matrix and build ALS based recommendation systems. You'll evaluate these systems in two ways: Objective evaluations via RMSE (root mean square error) metrics on Validation and Test sets and Subjective evaluations by giving it a sample user-item set of ratings and generating recommendations from these systems.

2. Data Set Details

We will be working with up to two different data sets. The first one is the MovieLens data set which includes ratings by a large number of users on a collection of movies. We'll be working with both the **10M** data set which includes 10 million ratings applied to 10,000 movies by 72,000 users and the much larger **MovieLens-latest** data set which includes 22 million ratings applied to 33,000 movies by 240,000 users. In addition, we'll also be working with the Million Song Data set and build a music recommendation system. We will use both these data sets as the Movie Lens data set contains explicit ratings whereas the Million Song Data set contains implicit ratings. Further, MSD is significantly more sparse than MovieLens.

- MovieLens 10M data set
- MovieLens Latest - Chose the larger of the two
- Million Song Data set

3. Task 1

Taking ratings data from the MovieLens (either 10M or the full 22M) data set, build an ALS model with a small number of latent factors, between 10-50 factors. We strongly recommend that you first try your code on the 10M data set and then build up to the larger set from there. Split the data set by time into 60-20-20 train-validate-test partitions. That is, the first 60% of the data sorted by time is the training set. The next 20% is for validation and the remaining 20% is for test. You'll use the training set to learn your ALS model and use the validation set to choose the regularization and number of latent factors. Once you have finished choosing your model using the validation set, you'll test it on the test set and report that error as your final error metric. Make sure you try different regularization

parameters ($\lambda \in (0.01, 0.1, 1.0, 10.0)$ and several latent factor dimensions and select the model that gives you the best RMSE on the validation set.

Finally, write a simple recommendation engine that will take the ALS model and a ratings file that contains a few ratings from one user and then comes back with a recommendation of movies for that user.

4. Task 2

So far, we used the ratings as they were. We didn't try to remove bias, or convert the numerical ratings into binary. Also, we didn't attempt any dimensionality reduction in either the **User** or **Item** space. For the second task, you'll attempt both of these. There are several ways of removing bias and build recommendation systems on the deviations. Remove bias from your data set by calculating the global μ, user-specific bias $\hat{b_x}$, and item-specific bias $\hat{b_i}$. Further, you can attempt to reduce dimensionality of the utility matrix to something manageable by doing KMeans clustering on both the user space and item space. Reduce both the user space and item space by 50%. Now, repeat the ALS model fitting from above. Take care to ensure that you do not do your KMeans clustering on either the validation or test partitions. Similarly, for learning the bias values. Any dimensionality reduction, bias learning etc should be done only on the training set and these learnt parameters should be applied to the validation and test sets.

5. Task 3

Fit an ALS model on the Million Song Data set and build a music recommendation engine. At the end, you should have a program that will take an ALS model, and a ratings file and return back a set of recommendations.

6. Assignment Submission

In your submission, we'll need the following

- Your code that documents the experiments you performed including data reading, splitting, cross validation and testing
- Your experimental results documenting your validation of the 3 models (plain vanilla, bias removed model, and Million Song Data set model)
- Your program that will take an ALS model, and ratings files and generate recommendations
- The Test set RMSE for all your models

7. Grading Rubric

- Tier 1 (70 points) for successfully completing Task 1
- Tier 2 (+20 points) for successfully completing Task 1 and 2
- Tier 3 (+10 points) for completing all 3 tasks