Design Document
Recommender Systems

Made By-
Mudit Chaturvedi (2018A7PS0248H),
Sristi Sharma (2018A7PS0299H),
Tanay Gupta (2018AAPS0343H)

1. **Dataset**
   https://grouplens.org/datasets/movielens/
   100K movie ratings.100,000 ratings from 1000 users on 1700 movies.

   Note: The generous raters and strict raters are handled appropriately using Pearson Correlation Coefficient.

2. **Collaborative Filtering -**
   Collaborative filtering filters information by using the interactions and data collected by the system from other users.
   This is done on the base that people who agreed on the same thing are likely to agree in the future . In the neighborhood-based approach, a number of users are selected based on their similarity to the active user. Inference for the active user is made by calculating a weighted average of the ratings of the similar users.
   Collaborative-filtering systems focus on the relationship between users and items. The similarity of items is determined by the similarity of the ratings of those items by the users who have rated both items.

   - User based : measures the similarity between target users and other users.
   - Item-based : measures the similarity between items that users rate or prefer with other items.

3. **Collaborative Filtering Using Baseline Approach -**
   This method is identical to the previous one except for the fact that we include the baseline model this time. We manage to correct the negative predictions by averaging the distance from the baseline predictor. It also helps extend collaborative filtering to large user bases. Since we subtract the baseline estimate from the ratings matrix, we get a normalized matrix to which we just need to apply the similarities.

$$r_{xi} = b_{xi} + \frac{\sum_{j \in N(i;x)} s_{ij} \cdot (r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s_{ij}}$$

baseline estimate for r_{xi}

$$b_{xi} = \mu + b_x + b_i$$

- μ = overall mean movie rating
- b_x = rating deviation of user x = (*avg. rating of user x*) – μ
- b_i = rating deviation of movie i

Similarity Measure : **Pearson Correlation**

$$r = \frac{\sum(X-\overline{X})(Y-\overline{Y})}{\sqrt{\sum(X-\overline{X})^2}\sqrt{(Y-\overline{Y})^2}}$$

Where, $\overline{X}$ = mean of X variable

$\overline{Y}$ = mean of Y variable

r = Pearson Correlation

4. Singular Value Decomposition -

For any MxN matrix, SVD can be calculated by decomposing it into 3 constituent matrices :
Let the rank of the original matrix be R.

- Matrix U - A column orthonormal matrix, having dimensions MxR.
- Matrix ∑: A diagonal matrix containing the singular values of the original matrix. Dimensions of this matrix are RxR.
- Matrix V : A column orthonormal matrix, having dimensions RxN.

The product of these matrices will serve as an approximation for the original matrix.
Approximation of original matrix = $U*\sum*V^T$.

5. Singular Value Decomposition with 90% retained energy -

90% energy rule: Retain enough singular values to make up 90% of the energy. That is, the sum of squares of the retained singular values should be at least 90% of the sum of squares of all the singular values.

All the other computation remains same as SVD with 100%, except that the dimensions of the matrices are reduced to keep at least 90% retention.

If this reduction is significant then the computation time is drastically reduced.

6. **CUR -**

The Utility Matrix is divided into 3 constituent matrices C,U and R.

- Matrix C - Formed by picking ‘t’ random columns of the original matrix. The size that we have used is Mx300.
- Matrix U - This matrix is formed by taking the pseudoinverse of the SVD of intersections between C and R. The size of this matrix is ‘t’x‘t’, where ‘t’ is the number of rows/columns to be picked randomly. In our case it is 300x300.
- Matrix R - Formed by picking ‘t’ random rows of the original matrix. The size that we have used is 300xN.

7. **CUR with 90% retained energy -**

90% energy rule: Retain enough singular values to make up 90% of the energy. That is, the sum of squares of the retained singular values should be at least 90% of the sum of squares of all the singular values.

All the other computation remains same as SVD with 100%, except that the dimensions of the matrices are reduced to keep at least 90% retention.

If this reduction is significant then the computation time is drastically reduced.

8. **Results -**

Recommender System Technique	**Root Mean Square Error (RMSE)**	**Precision on top K**	**Spearman Rank Correlation**	**Time taken for prediction**
Collaborative	0.076826	0.76	0.999704	30.9885 s
Collaborative along with Baseline	0.080212	0.76	0.999677	31.4110 s
SVD	0.006612	0.86	0.999999	5.2046 s
SVD with 90% retained energy	0.006601	0.86	0.999998	4.3908 s
CUR	0.0103767	0.9775553	0.9996479	6.9832 s
CUR with 90% retained energy	0.0114145	0.9775553	0.9995831	6.7307 s

Explanation of results :

As can be seen from the above table, CUR has the highest Precision on top K, while Collaborative Filtering has the lowest. Time taken for prediction of ratings and results is least for SVD while that for Collaborative Filtering is maximum.
Thus we can conclude that for the given dataset, CUR gives us the best results while Collaborative Filtering will be most off the mark.