DESIGN DOCUMENT

BUILDING A RECOMMENDER SYSTEM USING COLLABORATIVE FILTERING, MATRIX DECOMPOSITION AND LATENT FACTOR MODELLING

SUBJECT- INFORMATION RETRIEVAL (CS F469)

Group Members-

SUMANASA SOMU	2017A7PS0114H
L SRIHARI	2017A7PS1670H
PRANAV ANAND P	2017AAPS0379H
ASHWIN KUMAR RAJA	2017B4A70599H

About the System

The main aim of the system is to predict unknown ratings of movies. The corpus used is a Dataset of movies with ratings. The dataset is divided into training and testing and results are tabulated.

Data Structures Used

NumPy Matrix – To store the ratings per movie

Distances Used

1. **Euclidean distance-** The euclidean distance between any two points(vectors) (x1,y1) and (x2,y2) is defined as sqrt((x1-x2)^2+(y1-y2)^2)

2. **Cosine distance**- In order to remove discrepancies that could arise due to the variation in the size of the query and the document vectors themselves, we use a cosine distance measure. The higher the cosine coefficient between the two vectors, the lesser is the angle between them and the more similar they are. (Lesser distance)

3. **Jaccard coefficient measure**- It is a number between 0 and 1. It is defined as the number of elements in the intersection of two sets A and B divided by the number of elements in their union. The higher the coefficient, more is the similarity. (Lesser distance)

ARCHITECTURE / WORKING

1. The data which was extracted from https://grouplens.org/datasets/movielens/ is pre-processed by the preprocess.py file which creates and saves the sparse matrix A and test data in the same directory.
2. Collaborative Filtering on the sparse matrix A and predicting the test data is done by running collaborative_filtering.py.

3. Collaborative Filtering with Baseline approach on the sparse matrix A and predicting the test data is done by running collaborative_filtering_baseline.py.
4. Singular Value Decomposition (SVD) approach to predicting matrix A is done by running svd.py. This includes both 100% energy retention model and the 90% energy retention model.
5. CUR decomposition approach with 100% energy retention is done on sparse matrix A by running cur.py.
6. CUR decomposition approach with 90% energy retention is done on sparse matrix A by running cur_90.py.

Recommender System Technique	Root Mean Squared Error (RMSE)	Mean Average Error (MAE)	Time taken for prediction
Collaborative	0.334	0.268	1.3 seconds per value
Collaborative along with Baseline approach	0.318	0.242	2.3 seconds per value
SVD	0.110	0.102	1 min 14 seconds
SVD with 90 per cent energy	0.742	0.638	1 min 2 seconds
CUR	0.982	0.899	43 seconds
CUR with 90 per cent energy	0.998	0.912	32 seconds
Latent factor model			