CS F469 INFORMATION RETRIEVAL

ASSIGNMENT 3

Design Document

Topic: Recommender System

Group Members

Rohan Maheshwari (2017B4A70965H)

Giridhar Bajpai (2017B4A71451H)

Soumil Agarwal (2017B4A71606H)

Introduction

This project is aimed at implementing and comparing various techniques for building a Recommender System for the given dataset.

Several techniques were used to achieve this

a. Collaborative filtering
b. SVD
c. CUR

For each method used, RMSE, Precision on top K, Spearman Rank Correlation and time taken for computation was calculated and based on that comparison could be made on the methods used.

Techniques

1. **Collaborative Filtering** :

The process of identifying similar users and recommending what similar users like is called collaborative filtering.

Assumptions

- We can predict the rating of item i by user x through a set of other users whose ratings are “similar” to x’s ratings or by a set of other items whose ratings are “similar” to i’s ratings.

Advantages and Disadvantages of Collaborative Filtering

Advantages

- No feature selection needed.Recommendation are based on user-user or item-item similarity.
- Takes care of strict and lenient raters.

Cons

- Sparsity : The user or ratings matrix is sparse so it is hard to find users that have rated the same items
- Popularity bias : It tends to recommend popular items so it cannot recommend items to someone with unique taste .

Collaborative Filtering with baseline:

Assumptions

- We can predict the rating of item i by user x through a set of other users whose ratings are “similar” to x’s ratings or by a set of other items whose ratings are “similar” to i’s ratings.
- We also need to consider the deviation of that item i and user x from over-all mean of the corpus.

Advantages and Disadvantages of Collaborative Filtering with Baseline

Advantages

- No feature selection needed. Recommendations are based on user-user or item-item similarity.
- For any negative or zero predictions we are replacing it with its baseline estimator.

Disadvantages

- Sparsity : The user/ratings matrix is sparse so it is hard to find users that have rated the same items.
- Popularity bias : It tends to recommend popular items so it cannot recommend items to someone with unique taste .

2. **Single Value Decomposition** :

It is a factorization method which is used to decompose a real valued matrix. SVD factorizes a given matrix A into U, Sigma and V^T.

$$A_{[m \times n]} = U_{[m \times r]} \Sigma_{[r \times r]} (V_{[n \times r]})^T$$

- A is the input data matrix of size mxn.
- Matrix U: An mxr column orthonormal matrix where r is the rank of the original matrix.
- Matrix ∑: An rxr diagonal matrix containing the singular values of the original matrix.
- Matrix V: An rxn column orthonormal matrix where r is the rank of the original matrix

The original matrix can then be approximated by calculating the product of these matrices as follows: Approximation of original matrix = U*∑*VT

Dimensionality Reduction: If the rank r of the matrix is significantly smaller than N, we can observe that the combined sizes of the component matrices will be smaller than that of the original matrix, thereby giving a notable reduction in dimensionality, resulting in ease of computational load.

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.

Advantages and Disadvantages of SVD

Advantages

- Discover hidden correlations Correlation in two dimensions in 2 matrix gives us less information
- Remove redundant and noisy features : Features that are highly correlated to some other feature or feature that are giving noisy data can be removed by reducing the dimensions of the sigma matrix.
- Interpretation and visualization : The new space between user and movie can give much more insight than previous space.

Disadvantages

- Lack of Sparsity : Singular Matrices are dense.
- Computational Time : Computational cost is cubic time in the size of data to compute.

3. CUR Decomposition :

CUR matrix decomposition is a low-rank matrix decomposition algorithm that uses a lesser number of columns and rows than the data matrix.This number is represented by the variable k. In our data, we have taken k = 1000.

Rows in C: 1000
Columns in C: 6040
Rows in R: 1000
Column in R: 3952

Formulation:

- Utility matrix of size mxn is decomposed into three matrices C, U, R
- Matrix C: A mxr column orthonormal matrix where r is the rank of the original matrix.
- Matrix U: A rxr diagonal matrix containing the singular values of the original matrix.
- Matrix R: A rxn column orthonormal matrix where r is the rank of the original matrix.

The original matrix can then be approximated by calculating the product of these matrices as follows: Approximation of original matrix = C * U * R^T

Advantages and Disadvantages of CUR

Advantages

- Easy Interpretation : Since the basis vectors are actual columns and rows.
- Sparse Basis : Since the basis vectors are actual columns and rows.

Disadvantages

- Duplicate Columns and rows : Columns of large norms will be sampled many times.

Packages Used

1. Pandas: For easy data manipulation by use of DataFrames
2. Numpy: For matrix multiplication
3. Sklearn.model_selection: for train_test_split function
4. Sklearn.metrics: for finding Mean squared Error
5. Math: Sqrt function
6. Time: to compute time taken by various techniques

Data Structures Used

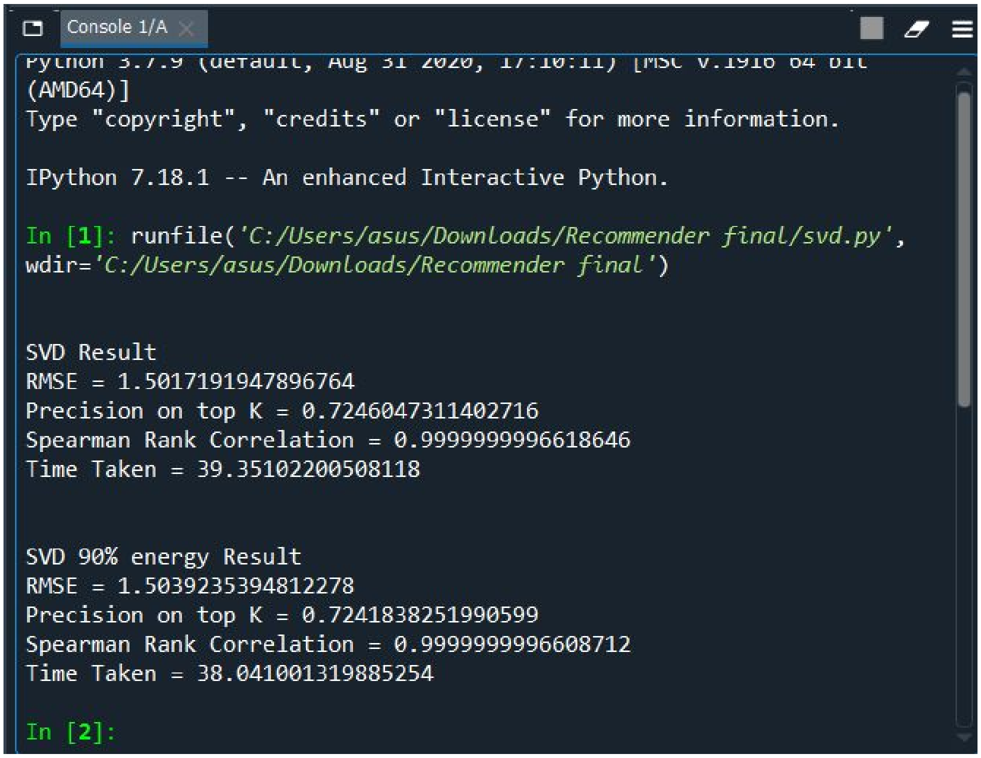
1. pandas.DataFrame: used to store the MovieLens dataset
2. Dictionary: used to store user to movie and movie to user vectors for faster retrieval
3. Numpy NDarray: for storing matrices in SVD and CUR procedures

Results

Recommender System Technique	RMSE	Precision on top K	Spearman Rank Correlation	Time taken for prediction(seconds)
Collaborative	1.3258	0.7058	0.9999	80.0964*
Collaborative along with baseline approach	1.3179	0.6759	0.9999	85.0095*
SVD	1.501791	0.724604	0.999999	39.351
SVD with 90% retained energy	1.503923	0.724183	0.999999	38.041
CUR	0.687982	0.966478	0.999999	117.246304
CUR with 90% retained energy	0.687287	0.966478	0.999999	110.048465

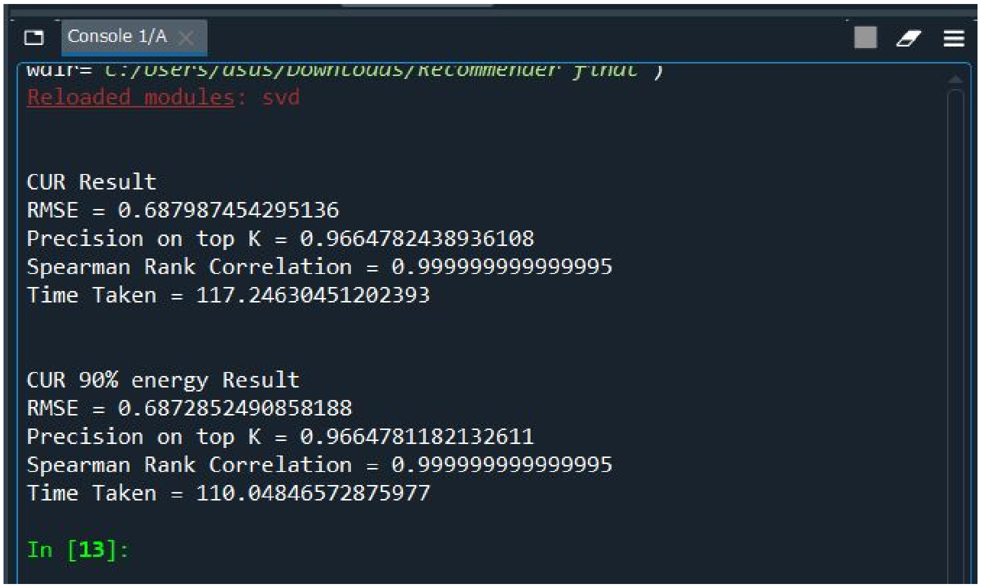
* For demo purposes we have only taken 100 ratings in the test dataset as collaborative filtering is an expensive process. Time Complexity is O(m*m*n) where m is the number of users and n is the number of movies here m = 6000 and m = 4000. To get more accurate results, pls increase the split_value variable.

Output for SVD



```
Python 3.7.9 (default, Aug 31 2020, 17:10:11) [MSC v.1916 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.18.1 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/asus/Downloads/Recommender final/svd.py',
wdir='C:/Users/asus/Downloads/Recommender final')


SVD Result
RMSE = 1.5017191947896764
Precision on top K = 0.7246047311402716
Spearman Rank Correlation = 0.9999999996618646
Time Taken = 39.35102200508118


SVD 90% energy Result
RMSE = 1.5039235394812278
Precision on top K = 0.7241838251990599
Spearman Rank Correlation = 0.9999999996608712
Time Taken = 38.041001319885254

In [2]:
```

Output for CUR



```
wdir='C:/Users/asus/Downloads/Recommender final')
Reloaded modules: svd


CUR Result
RMSE = 0.687987454295136
Precision on top K = 0.9664782438936108
Spearman Rank Correlation = 0.999999999999995
Time Taken = 117.24630451202393


CUR 90% energy Result
RMSE = 0.6872852490858188
Precision on top K = 0.9664781182132611
Spearman Rank Correlation = 0.999999999999995
Time Taken = 110.04846572875977

In [13]:
```