

DESIGN DOCUMENT FOR RECOMMENDER SYSTEM (CS-535)

Recommender System

Recommendation System: Recommender systems have changed the way people find products, information, and even other people. They study patterns of behavior to know what someone will prefer from among a collection of things he has never experienced. From an algorithmic point of view, recommender systems need to understand the content as well as the user. Applications include business, media, and social recommendation scenarios.

Code Flow

Matrix is initialized with all the values set to 0. The input data provided by the professor, i.e. “train_all_txt.txt” is placed over the matrix. Pearson’s similarity algorithm is used to find the similar user in the list of 943 users. I have used HashMap to store user as key and similar users as value. Hence, depending on the calculation of Pearson’s correlation algorithm, the HashMap is updated.

Weighted Sum calculation is also implemented to find out improved accuracy between two similar users. To calculate the recommendation of the 0 value, the distance is calculated from the weighted sum and is updated in the new matrix. HashMap value is retrieved and the count is incremented for number of users. The rating given by these users is added and then is divided by total count in the list. For structured and maintaining equality throughout the project, when the answers is calculated, 1 is assumed to be the for recommendation less than 1 and in the same way 5 is taken as the answer if the recommended user rating is greater than 5.

Later, the value is then updated in the matrix.

Purpose of the following functions:

Main()-To read the input file->train_all_txt.txt

buildMatrix()-To compute a matrix (943x1682) with value 0 in it.

Pearson()-To calculate the similarity co-relation value, build clusters and produce end results.

weightedSum-To get the similarity level accurate.

References:

- Collaborative Filtering Recommender Systems By Michael D. Ekstrand, John T. Riedl and Joseph A. Konstan
- https://en.wikipedia.org/wiki/Collaborative_filtering
- https://en.wikipedia.org/wiki/Recommender_system
- https://en.wikipedia.org/wiki/Weight_function
- <http://grouplens.org/datasets/movielens/100k/>
- http://www.cs.carleton.edu/cs_comps/0607/recommend/recommender/modelbased.html