

Collaborative filtering

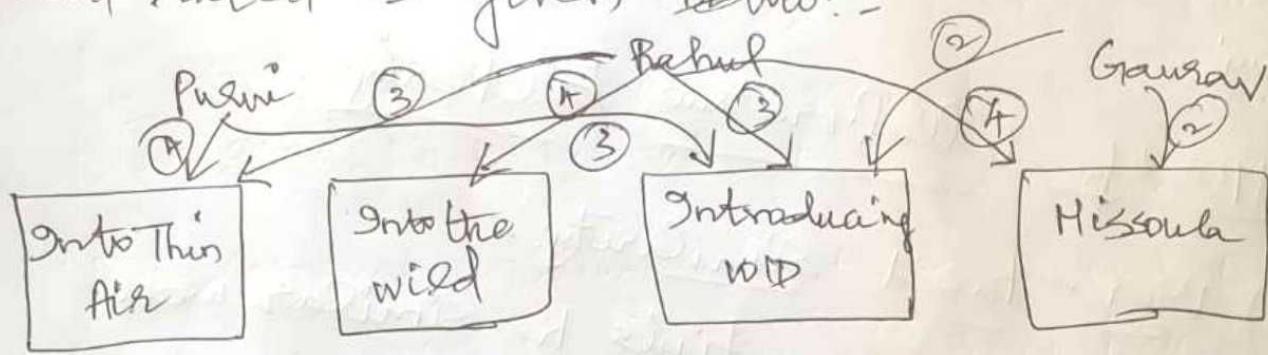
It is based on the notion of similarity (or distance). For ex, if two users A and B have purchased the same products and have rated them similarly on a common rating scale, then A and B can be considered similar in their buying and preference behavior. Hence, if A buys new product and rates high, then that product can be recommended to B. Alternatively, products that A has already bought and rated high can be recommended to B, if not already bought by B.

Finding similarity bet' users:-

Similarity / Distance between users can be computed using the rating the users have given to the common items purchased.

Common measures used are:- Jaccard coefficient, Euclidean distance and Cosine Similarity.

Example:- Assume three users Rakesh, Purnima and Gaurav and the books they have bought and rated as given below:-



The two books "Into Thin Air" and "Missoula" are commonly bought by all three users. They can be represented in the Euclidean Space as in the fig. below -



In this Euclidean Space, we observe that Rakesh's preferences are similar to Purnima than Gaurav's. Hence the other book "Into

"the Wild", which Rachel has bought and rated high, can now be recommended to Priya.

Types of Collaborative Filtering:-

a) User-based Similarity:-

Finds K similar users based on common items they have bought.

b) Item-based Similarity:- Finds K similar items based on common users who have bought those items.

Both algorithms are similar to K-Nearest Neighbors (KNN).

Implementation of User-Based Similarity:-

Data used:- a) MovieLens - using this to find similar users based on common movies the users have watched & how they have rated.

b) "ratings.csv" - has ratings given by a user to a movie. Ratings are on a scale of 1 to 5. The dataset has the following features.

userId, movieID, rating and timestamp
Refer the Colab Notebook for the
implementations.