Recommendation Systems

Recommendation systems are classified into two broad groups

1. Content based systems
   1. Examines properties of items
2. Collaborative filtering systems
   1. Similarity measures between items

**Utility Matrix**

This is a matrix which is users on the rows and items on the columns. The value for each user-item pair denotes the degree of preference. The idea of a recommendation system is to predict the blank entries in this matrix.

**The Long Tail**

The distinction between the recommendation systems of the physical world and the online world is called the long tail phenomenon. It graphs popularity on the vertical axis and the items sold by the institution on the x. The long tail is in the online system which provides a variety of items, whereas the physical system shows a long vertical in the beginning.

**Content Based Recommendations**

In a content based system, a profile is built for each item, which represents the important characteristics of that item.

Some features of documents are not available directly and need to be calculated. For example, for text based systems, TF.IDF scores can be calculated to represent the documents.

To measure the similarity of two documents, we can use the Jaccard distance or the cosine distance as well.

To predict the rating of all items, a classifier can be built for each unique user. For this, a decision tree can be used as the simplest form, based on the user’s attributes. A variant of this would be to create an ensemble of many decision trees to make more accurate calculation by voting.

**Collaborative Filtering**

Recommendations for a user U is made by looking at users most similar to U, based on distance measures in their utility matrix.

Jaccard similarity = Intersection size / Union size

Jaccard distance = 1 - Jaccard similarity

Cosine distance can also be taken as a distance measure.

Ratings can be normalised or rounded to provide ease of distance measurement, by exaggerating how far apart or how near two users lie in terms of distance.

**The Duality of Similarity**

Finding similar users can be used to find similar items with the same utility matrix. This symmetry can be broken as

1. Use information about users to recommend items. However, there is no symmetry, as if we find pairs of similar items, we need to take an additional step to recommend items to users.
2. There is a difference in behavior of users and items, as it pertains to similarity.

One way of predicting the value of a utility matrix is to find n users most similar to U and take the average of their ratings for I, counting only those users who have rated I. Average the difference for those users and add the average to the average rating that U gives to all items, normalising the estimate.

Alternatively, we can find m similar items as well and take their average rating.

The tradeoff with working with user similarity or item similarity is

1. If we find similar users, we need to do this process once for each users. If we use items, we have to compute similarities for almost all items before we can estimate for row U.
2. Item-item similarity is more reliable as it is easier to bucket items than users.

We can cluster items to reduce the tradeoff by explicitly grouping similar items, and the revised utility matrix can use this clustered set of items rather than the whole itemset.

To predict

1. Find the clusters to which U and I belong, say C and D
2. If the entry in the cluster utility matrix for C and D is not blank, use this value as the estimate for the U-I entry in the original matrix
3. If the entry for C-D is blank, estimate entry by considering clusters similar to C and D.