**Types of recommender systems:**

1. **Content based** – show me more of the same of what I have liked before
2. **Collaborative Filtering** – tell me what’s popular among my neighbors, I also might like it
3. **Singular Value Decomposition (SVD)**
4. **Neural Network Approach**

**Implementation methods:**

1. Memory based – pearson correlation, cosine similarity, Euclidean distance, etc.
2. Model based – regression, clustering, classification
3. **Content based recommender systems**

* Content-based recommendation system tries to recommend items to the users based on their profile

1. **Collaborative filtering recommender systems**

Has two approaches:

1. **User-based:**
   * measure the similarity between target users and other users
2. **Item-based:**
   * measure the similarity between the items that target users rates/ interacts with and other items

**User-based Collaborative Filtering**

* We know that we need to compute the similarity between users in user-based CF. But how do we measure the similarity? There are two options, Pearson Correlation or cosine similarity. Let u\_{i, k} denotes the similarity between user i and user k and v\_{i, j} denotes the rating that user i gives to item j with v\_{i, j} = ? if the user has not rated that item. These two methods can be expressed as the followings:
* Both measures are commonly used. The difference is that Pearson Correlation is invariant to adding a constant to all elements.

**Challenges of collaborative filtering:**

* Data Sparsity – rating for limited number of items
* Cold Start – new user
* Scalability – increase in number of users or items

The process for creating a User Based recommendation system is as follows:

* Select a user with the movies the user has watched
* Based on his rating to movies, find the top X neighbours
* Get the watched movie record of the user for each neighbour.
* Calculate a similarity score using some formula
* Recommend the items with the highest score

**Advantages and Disadvantages of Collaborative Filtering**

**Advantages**

* Takes other user's ratings into consideration
* Doesn't need to study or extract information from the recommended item
* Adapts to the user's interests which might change over time

**Disadvantages**

* Approximation function can be slow
* There might be a low of amount of users to approximate
* Privacy issues when trying to learn the user's preferences

**Item-based Collaborative Filtering**

* Instead of measuring the similarity between users, the item-based CF recommends items based on their similarity with the items that the target user rated. Likewise, the similarity can be computed with Pearson Correlation or Cosine Similarity. The major difference is that, with item-based collaborative filtering, we fill in the blank vertically, as oppose to the horizontal manner that user-based CF does. The following table shows how to do so for the movie Me Before You.

1. **Singular Value Decomposition**

* One way to handle the scalability and sparsity issue created by CF is to leverage a latent factor model to capture the similarity between users and items. Essentially, we want to turn the recommendation problem into an optimization problem. We can view it as how good we are in predicting the rating for items given a user. One common metric is Root Mean Square Error (RMSE). The lower the RMSE, the better the performance. Since we do not know the rating for the unseen items, we will temporarily ignore them. Namely, we are only minimizing RMSE on the known entries in the utility matrix. To achieve minimal RMSE, Singular Value Decomposition (SVD) is adopted as shown in the below formula.

1. **Neural Network approach**

<https://hackernoon.com/introduction-to-recommender-system-part-1-collaborative-filtering-singular-value-decomposition-44c9659c5e75>

<https://towardsdatascience.com/introduction-to-recommender-system-part-2-adoption-of-neural-network-831972c4cbf7>