Recommendation System

Recommender systems are the systems that are designed to recommend things to the user based on many different factors. These systems predict the most likely product that the users are most likely to purchase and are of interest to. Companies like Netflix, Amazon, etc. use recommender systems to help their users to identify the correct product or movies for them.

The recommender system deals with a large volume of information present by filtering the most important information based on the data provided by a user and other factors that take care of the user's preference and interest. It finds out the match between user and item and imputes the similarities between users and items for recommendation.

Both the users and the services provided have benefited from these kinds of systems. The quality and decision-making process has also improved through these kinds of systems.

Why Recommendation System:

- Benefits users in finding items of their interest.
- Help item providers in delivering their items to the right user.
- Identity products that are most relevant to users.
- Personalized content.
- Help websites to improve user engagement.

What can be recommended:

There are many different things that can be recommended by the system like movies, books, news, articles, jobs, advertisements, etc. Netflix uses a recommender system to recommend movies & web-series to its users. Similarly, YouTube recommends different videos. There are many examples of recommender systems that are widely used today.

Types of Recommendation System:

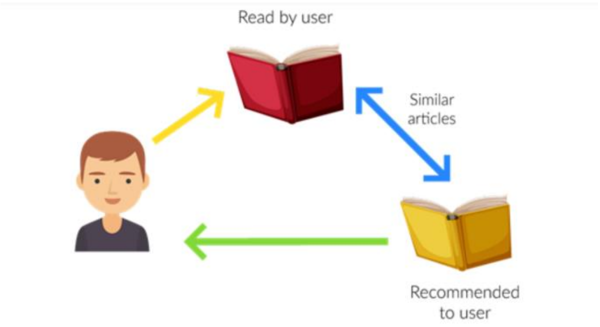
1. **Content Based Filtering:**

In this type of recommendation system, relevant items are shown using the content of the previously searched items by the users. Here content refers to the attribute/tag of the product that the user like. In this type of system, products are tagged using certain keywords, then the system tries to understand what the user wants and it looks in its database and finally tries to recommend different products that the user wants.

Content-based filtering is a type of recommendation system that recommends items to users based on their previous actions and preferences of particular user. It works by analyzing the features and attributes of items that the user has previously interacted with or liked, and then recommends items that are similar in content to those items.

For example, in a movie recommendation system, a content-based filtering approach would recommend movies to a user based on the attributes of the movies they have previously watched and rated highly, such as the genre, actors, director, plot summary, etc. The system would then recommend other movies with similar attributes.

Content-based filtering systems are effective in recommending items to users with specific and narrow interests. They do not require large amounts of data to work and can provide recommendations even for new or unpopular items. However, they may fail to recommend items outside the user's previous interests, resulting in a lack of diversity in recommendations.



Let us take an example of the movie recommendation system, where every movie is associated with its genres/ categories which in the above case is referred to as tag/attributes. Now let assume user A comes and initially system don't have any data about user A. so initially, the system tries to recommend the popular movies to the users or the system tries to get some information of the user by getting a form filled by the user. After some time, users might have given a rating to some of the movies like it gives a good rating to movies based on the action genre and a bad rating to the movies based on the anime genre. So here system recommends action movies to the users. But here you can't say that the user dislikes animation movies because maybe the user dislikes that movie due to some other reason like acting or story but actually likes animation movies and needs more data in this case.

Advantages:

- Model doesn't need data of other users since recommendations are specific to a single user.
- It makes it easier to scale to a large number of users.
- The model can Capture the specific Interests of the user and can recommend items that very few other users are interested in.

Disadvantages:

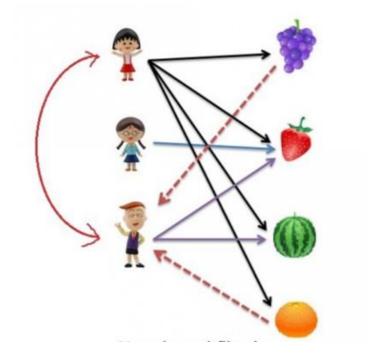
- Feature representation of items is hand-engineered to some extent, this tech requires a lot of domain knowledge.
- The model can only make recommendations based on the existing interest of a user. In other words, the model has limited ability to expand on the user's existing interests.

2. Collaborative Based Filtering:

Recommending the new items to users based on the interest and preference of other similar users is basically collaborative-based filtering. For e.g.: - When we shop on Amazon it recommends new products saying *"Customer who brought this also brought"* as shown below (like, you want to purchase a toy, then engine is recommended some toys because most of users are also bought those toys from the shop/ website & give good ratings).

There are 2 types of collaborative filtering: -

- **User Based Collaborative Filtering:**



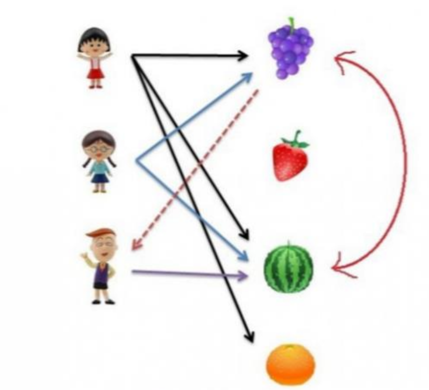
User-based filtering

Rating of the item is done using the rating of neighbouring users. In simple words, it is based on the notion of users' similarity.

Let see an example. On the left side, you can see a picture where 3 children named A, B, C, and 4 fruits i.e., grapes, strawberry, watermelon, and orange respectively.

Based on the image let assume A purchased all 4 fruits, B purchased only strawberry and C purchased strawberry as well as watermelon. Here A & C are similar kinds of users because of this C will be recommended Grapes and Orange as shown in dotted line.

- **Item Based Collaborative Filtering:**



Item-based filtering

The rating of the item is predicted using the user's own rating on neighbouring items. In simple words, it is based on the notion of item similarity.

Let us see with an example as talked above about users and items. Here the only difference is that we see similar items, not similar users like if you see grapes and watermelon, you will realize that watermelon is purchased by all of them but grapes are purchased by Children A & B. Hence Children C is being recommended grapes.

Now after understanding both of them, you may be wondering which to use when. Here is the solution if No. of items is greater than No. of users go with user-based collaborative filtering as it will reduce the computation power and If No. of users is greater than No. of items go with item-based collaborative filtering. For Example, Amazon has lakhs of items to sell but has billions of customers. Hence Amazon uses item-based collaborative filtering because of less no. of products as compared to its customers.

Advantages:

- It works well even if the data is small.
- This model helps the users to discover a new interest in a given item but the model might still recommend it because similar users are interested in that item.
- No need for Domain Knowledge.

Disadvantages:

- It cannot handle new items because the model doesn't get trained on the newly added items in the database. This problem is known as Cold Start Problem.
- Side Feature Doesn't have much importance. Here Side features can be actor name or releasing year in the context of movie recommendation.

3. Popularity Based Filtering:

A popularity-based recommendation system is a type of recommendation system that suggests items to users based on their overall popularity or popularity among the user community. It does not consider personalized preferences or individual user behaviour but instead focuses on the general popularity or trends.

The popularity-based recommendation system works by recommending items that are popular among all users or a specific target user group. It does not take into account user-specific characteristics or preferences. The recommendations are based on aggregate data, such as the number of views, likes, purchases, or ratings an item has received.

Here's a high-level overview of how a popularity-based recommendation system works:

- **Data Collection:** Gather information about items and their associated popularity metrics, such as views, likes, purchases, ratings, etc.
- **Popularity Calculation:** Calculate the popularity score for each item based on the collected metrics. For example, the popularity score can be the total number of views or the average rating.
- **Top-N Recommendations:** Generate a list of the most popular items by sorting them based on their popularity score. The top N items are recommended to users.
- **Recommendation Delivery:** Present the recommended items to users through a user interface, such as website, app, or email.

Advantages of popularity-based recommendation systems include their simplicity and ability to provide recommendations even when limited or no user-specific data is available. They are easy to implement and can be effective in certain scenarios, such as when introducing new users to popular items or when dealing with sparse data.

However, popularity-based recommendation systems have limitations. They do not consider individual user preferences and may recommend items that are already well-known or widely popular. Personalization and diversity in recommendations may be lacking since the system focuses on overall popularity. Therefore, popularity-based recommendation systems are not suitable for providing personalized recommendations tailored to individual user tastes and preferences.

It's important to note that popularity-based recommendation systems are just one approach among several others, such as collaborative filtering, content-based filtering, and hybrid models, each with their own advantages and limitations. The choice of recommendation system depends on the specific requirements, available data, and desired level of personalization.

4. Hybrid Recommendation System:

A hybrid recommendation system is a combination of two or more different recommendation techniques or algorithms to provide more accurate and diverse recommendations to users. It aims to leverage the strengths of different approaches while mitigating their individual limitations. By combining multiple recommendation methods, a hybrid system can potentially offer improved performance and overcome the limitations of single-method systems.

Here's an example to illustrate a hybrid recommendation system:

Let's consider a movie recommendation system that combines collaborative filtering and content-based filtering techniques:

- **Collaborative Filtering:** Collaborative filtering analyses the past behaviour of users and finds similarities among users or items. It recommends items based on the preferences of similar users or items. For example, if User A and User B have similar movie preferences and User A has watched and liked Movie X, then the system can recommend Movie X to User B.

- **Content-Based Filtering:** Content-based filtering focuses on the attributes or features of items. It recommends items that are similar in terms of their content or characteristics. For example, if a user has liked action movies in the past, the system can recommend other action movies with similar attributes, such as genre, director, or actors.

Hybrid Approach:

To create a hybrid recommendation system, we can combine the two approaches:

- **Collaborative Filtering Component:** The system collects user ratings or preferences for movies and identifies similar users based on their preferences. It generates recommendations based on the ratings of similar users.

- **Content-Based Filtering Component:** The system analyses the content features of movies, such as genre, director, and actors. It identifies movies with similar content attributes to the ones the user has liked in the past and generates recommendations based on these similarities.

- **Integration:** The recommendations generated by both components are combined using a fusion or weighting mechanism. This could involve assigning weights to each recommendation based on their respective confidence levels or using a voting system to determine the final recommendation.

The hybrid recommendation system benefits from the collaborative filtering's ability to capture user preferences and the content-based filtering's capability to capture item characteristics. It can provide recommendations that are both personalized based on user behaviour and aligned with the specific attributes of items.

Hybrid recommendation systems are widely used in various domains, including e-commerce, music streaming platforms, and movie recommendation engines. By combining different techniques, they can improve the accuracy, coverage, and diversity of recommendations, leading to enhanced user satisfaction and engagement.

Performance Matrix used in Recommendation System:

There are several performance metrics used in recommendation systems to evaluate their effectiveness. Here are some commonly used metrics:

- **Recall:** Recall is the ratio of the number of relevant items recommended by the system to the total number of relevant items. In other words, it measures the percentage of relevant items that were recommended by the system.
- **Mean Average Recall (MAR):**
- **Precision:** Precision is the ratio of the number of relevant items recommended by the system to the total number of items recommended by the system. It measures the percentage of recommended items that are actually relevant.
- **Mean Average Precision (MAP):**
- **False Positive Rate (FPR):**
- **The Area Under the ROC curve (AUC):**
- **F1 score:** The F1 score is a harmonic mean of recall and precision. It is used to balance the trade-off between recall and precision.
- **Mean absolute error (MAE):** MAE measures the average difference between the predicted and actual ratings of the items. It is used to evaluate the accuracy of the system's predictions.
- **Root mean squared error (RMSE):** RMSE is similar to MAE, but it measures the square root of the average squared differences between predicted and actual ratings.
- **Mean reciprocal rank (MRR):** MRR is the average of the reciprocal ranks of relevant items. It is used to evaluate the quality of the top-ranked items recommended by the system.
- **Normalized discounted cumulative gain (NDCG):** NDCG measures the quality of the recommended items based on their position in the list of recommendations. It takes into account the relevance of the items and their position in the list.

The choice of performance metric depends on the type of recommendation system and the specific requirements of the application.

Book Recommendation system is Content-Based or Collaborative-Based recommendation system?

Book recommendation systems can be both content-based or collaborative filtering, depending on how they generate recommendations.

Content-based book recommendation systems typically recommend books based on the user's past preferences and characteristics of the books, such as genre, author, and publication year. This approach involves building a user profile based on their past interactions with books, and then recommending similar books that match the user's interests.

Collaborative filtering book recommendation systems, on the other hand, generate recommendations based on the past behaviour of other users with similar interests. This approach involves analyzing the user's past interactions with books, as well as the interactions of other users with similar tastes, and then recommending books that those users have liked.

So, it is possible to use both content-based and collaborative filtering approaches in building book recommendation systems, depending on the specific needs and goals of the system.