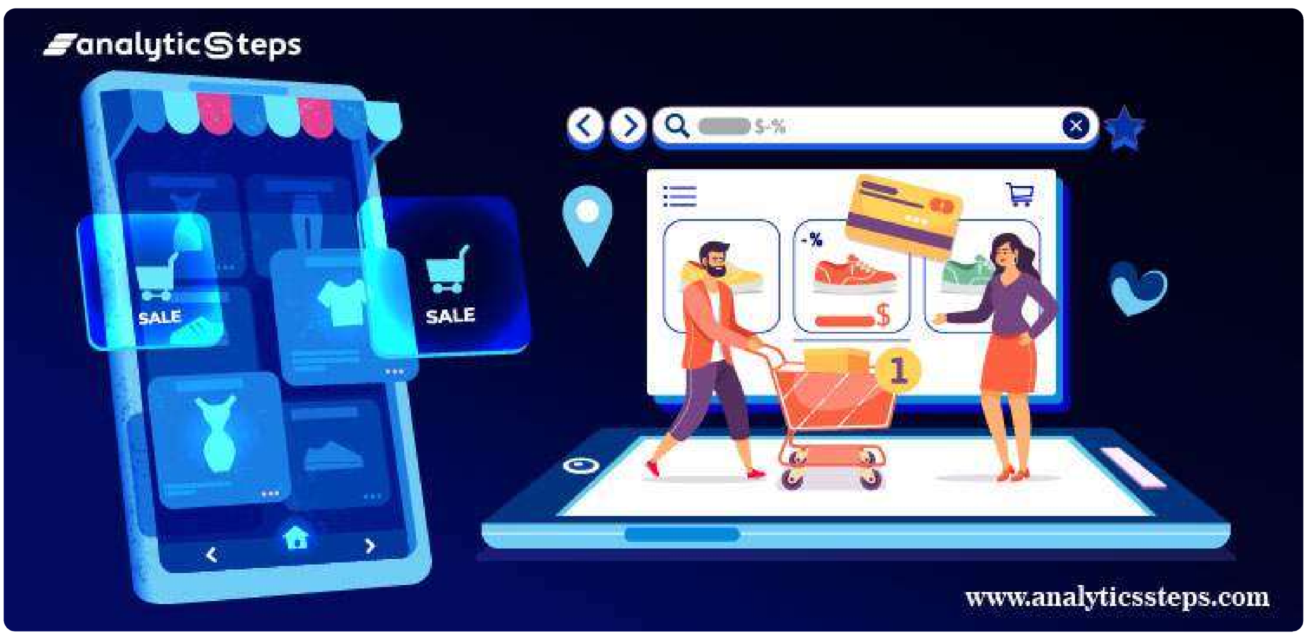
What Are Recommendation Systems in Machine Learning?

Rohit Dwivedi | Apr 16, 2020 | Updated on: Jan 19, 2021



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.

- 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.

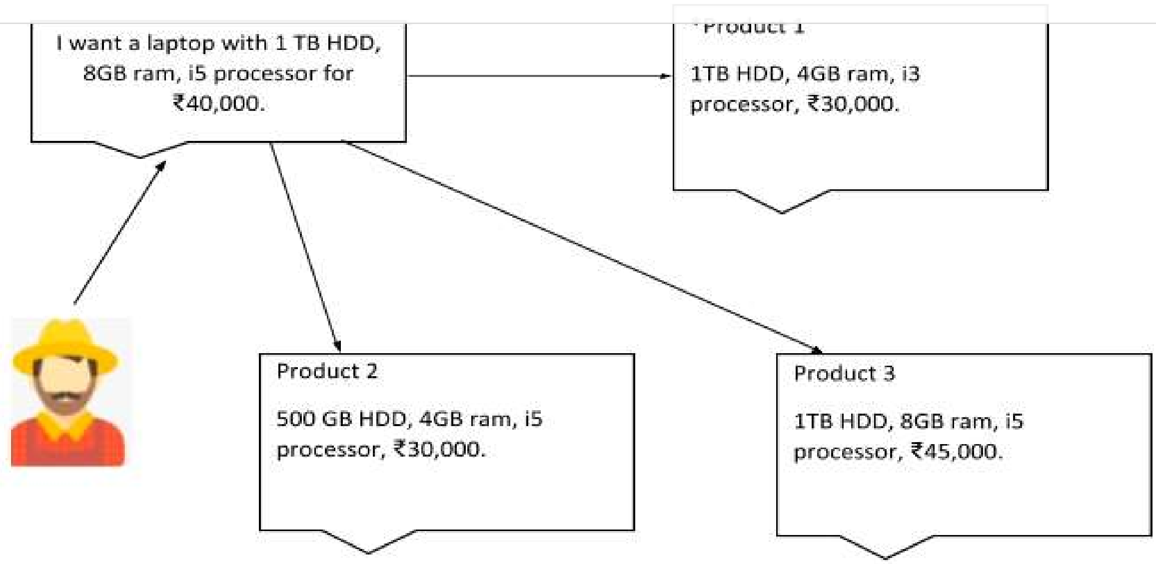
How do User and Item matching is done?

In order to understand how the item is recommended and how the matching is done, let us a look at the images below;

SOCIAL WEBSITES	USER	ITEM
Amazon	Members	Product
Netflix	Members	Movies
Linkedin	Members	Members
Facebook	Members	Jobs

Showing user-item matching for social websites

Perfect matching may not be recommended



Real-life user interaction with a recommendations system

The above pictures show that there won't be any perfect recommendation which is made to a user. In the above image, a user has searched for a laptop with 1TB HDD, 8GB ram, and an i5 processor for 40,000₹. The system has recommended 3 most similar laptops to the user.

Types of Recommendation System

1. Popularity-Based Recommendation System

It is a type of recommendation system which works on the principle of popularity and or anything which is in trend. These systems check about the product or movie which are in trend or are most popular among the users and directly recommend those.

For example, if a product is often purchased by most people then the system will get to know that that product is most popular so for every new user who just signed it, the system will recommend that product to that user also and chances becomes high that the new user will also purchase that.

Merits of popularity based recommendation system

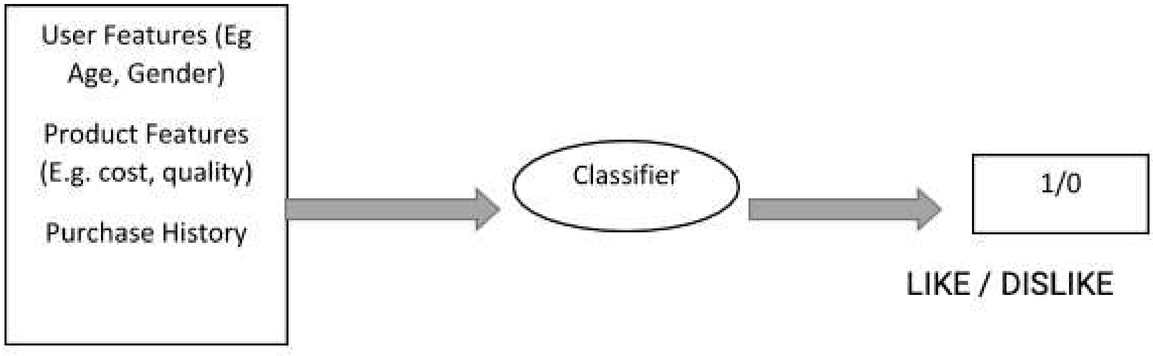
- It does not suffer from cold start problems which means on day 1 of the business also it can recommend products on various different filters.
- There is no need for the user's historical data.

Demerits of popularity based recommendation system

- Not personalized
- The system would recommend the same sort of products/movies which are solely based upon popularity to every other user.

Example

- Google News: News filtered by trending and most popular news.
- YouTube: Trending videos.



Classification model

The output can be either 0 or 1. If the user likes it then 1 and vice-versa.

Recommended blog: [Introduction to XGBoost Algorithm for Classification and Regression]

Limitations of Classification Model

- It is a rigorous task to collect a high volume of information about different users and also products.
- Also, if the collection is done then also it can be difficult to classify.
- Flexibility issue.

3. Content-Based Recommendation System

It is another type of recommendation system which works on the principle of similar content. If a user is watching a movie, then the system will check about other movies of similar content or the same genre of the movie the user is watching. There are various fundamentals attributes that are used to compute the similarity while checking about similar content.

To explain more about how exactly the system works, an example is stated below:

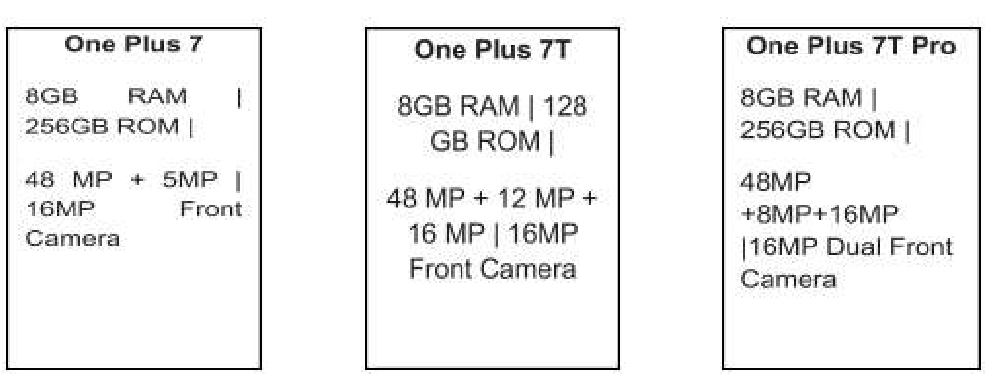


Figure1: Different models of one plus.

Figure 1 image shows the different models of one plus phone. If a person is looking for one plus 7 mobile then, one plus 7T and one plus 7 Pro is recommended to the user.

To check the similarity between the products or mobile phone in this example, the system computes distances between them. One plus 7 and One plus 7T both have 8Gb ram and 48MP primary camera.

If the similarity is to be checked between both the products, Euclidean distance is calculated. Here, distance is calculated based on ram and camera;

$$\sqrt{(8-8)^2 + (48-48)^2} = 0$$

Euclidean distance (7T,7)

$$\sqrt{(8-12)^2 + (48-48)^2} = 4$$

Euclidean distance (7Pro,7)

Euclidean distance between (7T,7) is 0 whereas Euclidean distance between (7pro,7) is 4 which means one plus 7 and one plus 7T have similarities in them whereas one plus 7Pro and 7 are not similar products.

In order to explain the concept through this example, only the basic thing (camera and ram) was taken but there is no restriction. We can compute distance calculation for any of the features of the product. The basic principle remains the same if the distance between both is 0, they are likely to have similar content.

There are different scenarios where we need to check about the similarities, so there are different metrics to be used. For computing the similarity between numeric data, Euclidean distance is used, for textual data, cosine similarity is calculated and for categorical data, Jaccard similarity is computed.

Euclidean Distance: Distance between two points can be calculated by the equation;

$$\text{Inner}(x, y) = \sum_i x_i y i = (x, y)$$

The formula for Euclidean distance

Cosine Similarity: Cosine of the angle between the two vectors of the item, vectors of A and B is calculated for imputing similarity. If the vectors are closer, then small will be the angle and large will be the cosine.

$$\text{Similarity}(X,Y) = \frac{X.Y}{|X| \times |Y|}$$

Cosine Similarity

Jaccard Similarity: Users who have rated item A and B divided by the total number of users who have rated either A or B gives us the similarity. It is used for comparing the similarity.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Jaccard Similarity

Merits

- There is no requirement for much of the user's data.
- We just need item data that enable us to start giving recommendations to users.
- A content-based recommender engine does not depend on the user's data, so even if a new user comes in, we can recommend the user as long as we have the user data to build his profile.
- It does not suffer from a cold start.

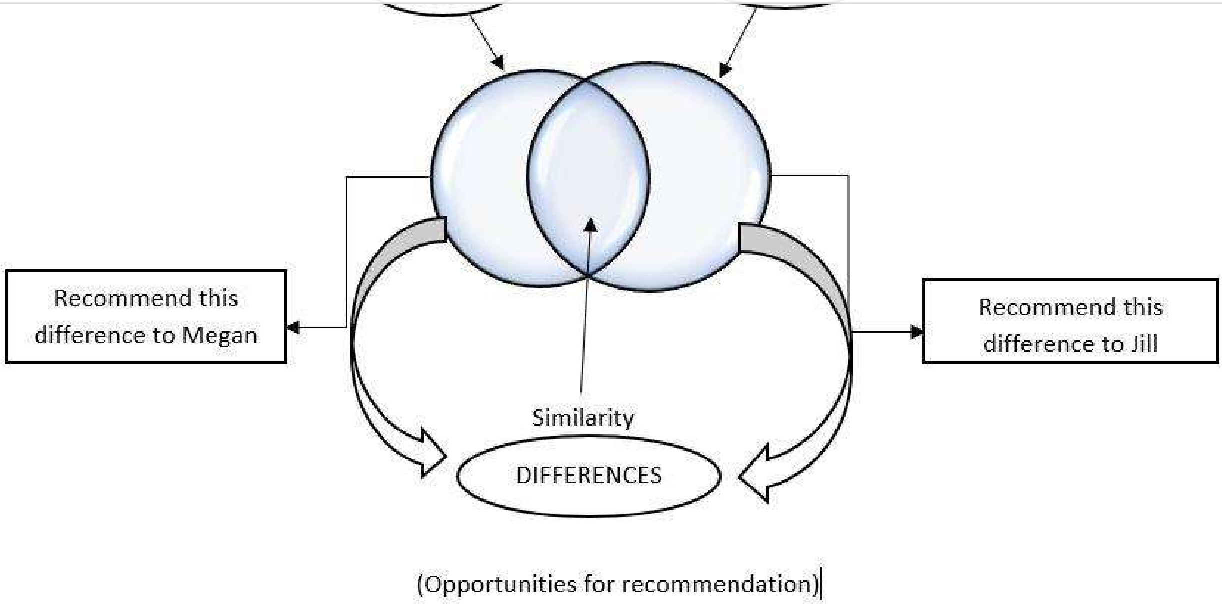
Demerits

- Items data should be in good volume.
- Features should be available to compute the similarity.

3. Collaborative Filtering

It is considered to be one of the very smart recommender systems that work on the similarity between different users and also items that are widely used as an e-commerce website and also online movie websites. It checks about the taste of similar users and does recommendations.

The similarity is not restricted to the taste of the user moreover there can be consideration of similarity between different items also. The system will give more efficient recommendations if we have a large volume of information about users and items.



Concept of collaborative filtering.

Figure 2 shows the two different users and their interests along with the similarity between the taste of both the users. It is found that both Jil and Megan have similar tastes so Jill's interest is recommended to Megan and vice versa.

This is the way collaborative filtering works. Mainly, there are two approaches used in collaborative filtering stated below;

a) User-based nearest-neighbor collaborative filtering

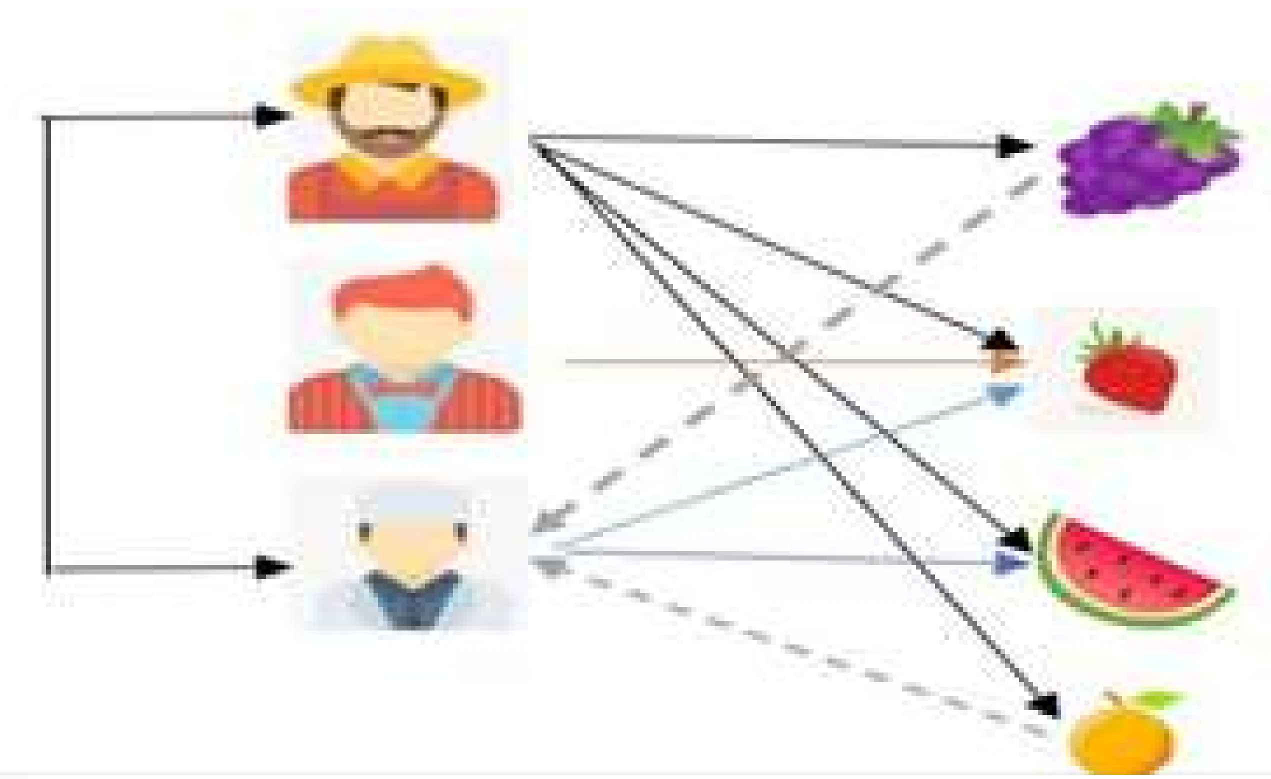


Figure 3 shows user-user collaborative filtering where there are three users A, B and C respectively and their interest in fruit. The system finds out the users who have the same sort of taste of purchasing products and similarity between users is computed b... purchase behavior. User A and User C are similar because they have purchased similar products.

b) Item-based nearest-neighbor collaborative filtering

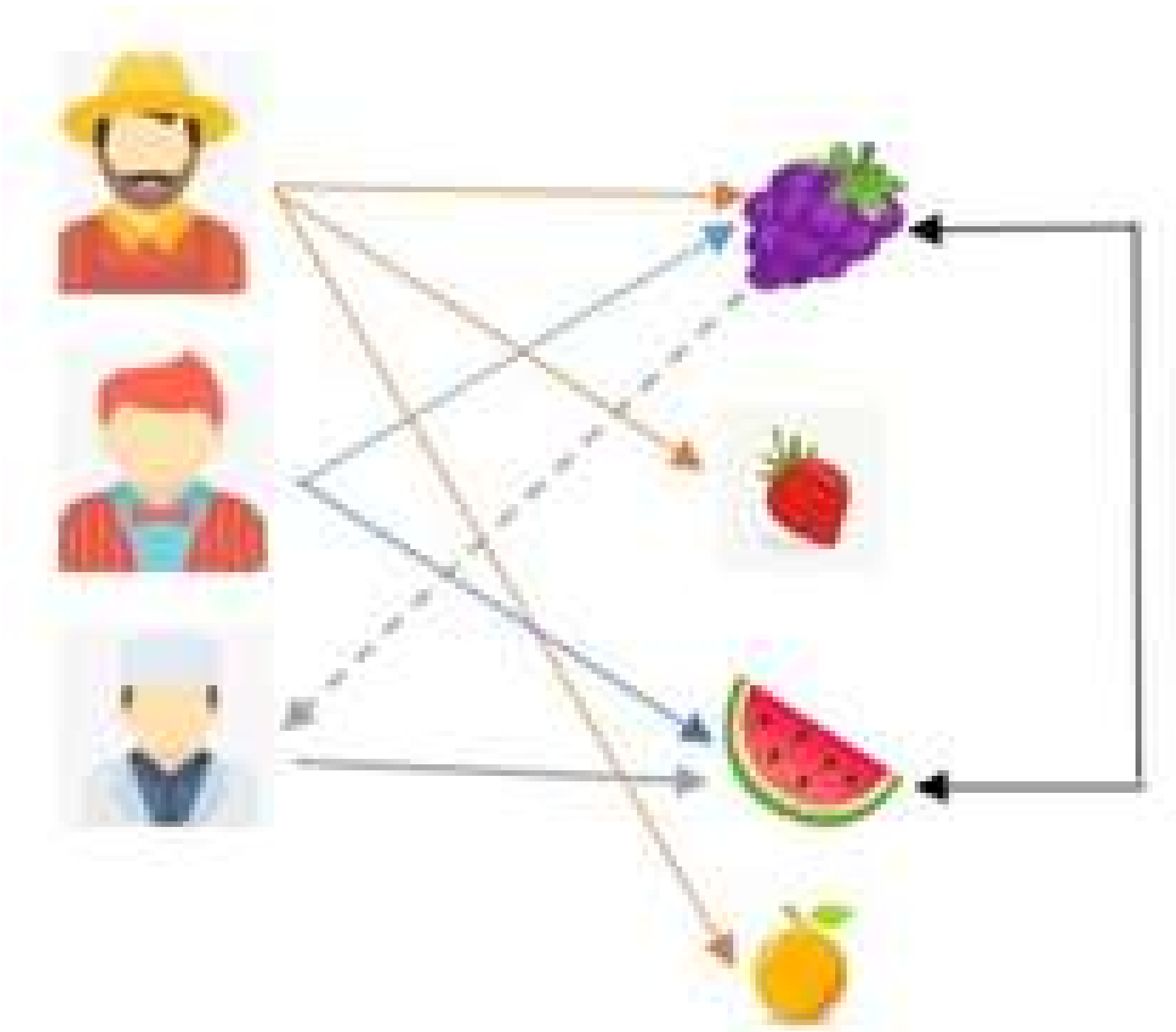


Figure 4: Item-Item Collaborative filtering.

Figure 4 shows user X, Y, and Z respectively. The system checks the items that are similar to the items the user bought. The similarity between different items is computed based on the items and not the users for the prediction. Users X and Y both purchased items A and B so they are found to have similar tastes.

Limitations

- Enough users required to find a match. To overcome such cold start problems, often hybrid approaches are made use of between CF and Content-based matching.
- Even if there are many users and many items that are to be recommended often, problems can arise of user and rating matrix to be sparse and will become challenging to find out about the users who have rated the same item.
- The problem in recommending items to the user due to sparsity problems.

Singular value decomposition also known as the **SVD algorithm** is used as a collaborative filtering method in recommen

SVD is a matrix factorization method that is used to reduce the features in the data by reducing the dimensions from N to K where (K<N).

For the part of the recommendation, the only part which is taken care of is matrix factorization that is done the user-item rating matrix. Matrix-factorization is all about taking 2 matrices whose product is the original matrix. Vectors are used to represent item 'qi' and user 'pu' such that their dot product is the expected rating.

$$\text{expected rating} = \hat{r}_{ui} = q_i^T p_u$$

The Formula for an expected rating

'qi' and 'pu' can be calculated in such a way that the square error difference between the dot product of user and item and the original ratings in the user-item matrix is least.

$$min(p,q) \sum_{(u,i)\varepsilon k} \left(r_{ui} - q_i^T \cdot p_u\right)^2$$

The formula for regularization without regularization factor

Regularization: Avoiding overfitting of the model is an important aspect of any machine learning model because it results in low accuracy of the model. Regularization eliminates the risk of models being overfitted.

For this purpose in regularization, a penalty term is introduced to the above minimization equation. λ is the regularization factor which is multiplied by the square sum of the magnitudes of user and item vectors.

$$minimum(p,q) \sum_{(u,i)\varepsilon k} \left(r_{ui} - q_i^T \cdot p_u\right)^2 + \lambda\left(\|q_i\|^2 + \|p_u\|^2\right)$$

The formula for regularization with regularization factor

To understand and explore the importance of the factor which is introduced above, let's consider a case where a user has rated a very low rating to a movie and has not rated any other movie except that.

The above algorithm will reduce the error by imputing 'qi' a bigger value which will result in all ratings to all the movies be low.

Bias terms: Algorithms make use of features of the data to minimize the error between the actual value and the predicted value. To be specific for each user/user item u & i, we can pull three parameters; 'bu' (Ratings are given by the user u which tell about rating), 'μ' (Ratings of all items) & 'bi' (the rating of item i - μ).

$$\hat{r}_{ui} = q_i^T \cdot p_u + \mu + b_i + b_u$$

Bias Term

The minimized equation is,

$$minimum\,(p, q, bi, bu) = \sum_{(u,i)\epsilon k} \left(r_{ui} - q_i^T \cdot p_u - \mu - b_i - b_u\right)^2 + \lambda\left(\|q_i\|^2 + \|p_u\|^2 + b_i^2 + b_u^2\right)$$

The minimized equation for the above formula

Minimizing with Stochastic Gradient Descent (SGD): SGD is used to reduce the above equation. SGD functions by taking the parameters of the equation which we are trying to reduce to initial values and then iterating it to minimize the incorrect error between the actual value & the predicted value by making the use of a small factor each time to correct.

SGD makes the usage of the learning rate to check about the previous values and the new value after every other iteration.

Recommended Blog: How Does Support Vector Machine Algorithm Works In Machine Learning?

Conclusion

I would conclude the blog by stating that the recommendation system changed the whole scenario by making it easy for the user to choose their desired choices and of interest. It recommends user personalized content. There are various other platforms where these systems are currently used.

In the blog, we have discussed the recommendation system, its types, and the multiple techniques that are used in a recommendation system. But, there can be many advancements in technology that can be foreseen in the future as there lie many challenges in the recommendation system ahead.

Or

Be a part of our *Instagram* community ♥

TRENDING BLOGS

Elasticity of Demand and its Types
READ MORE

5 Factors Influencing Consumer Behavior
READ MORE

What is PESTLE Analysis? Everything you need to know about it
READ MORE

An Overview of Descriptive Analysis
READ MORE

5 Factors Affecting the Price Elasticity of Demand (PED)
READ MORE

Dijkstra's Algorithm: The Shortest Path Algorithm
READ MORE

6 Major Branches of Artificial Intelligence (AI)
READ MORE

What is Managerial Economics? Definition, Types, Nature, Principles, and Scope
READ MORE

Latest Comments

Digvijay.Bisht1992 Apr 17, 2020
That was nice Rohit. Keep going!!
2 Likes 1 Reply

Rohit Dwivedi Apr 17, 2020
Thanks alot. Keep reading! :)
1 Like

Write a comment..

Post Comment Contact Us