PARSHWANATH CHARITABLE TRUST'S
A.P. SHAH INSTITUTE OF TECHNOLOGY
Department of Computer Science and Engineering
Data Science
CSE DATA SCIENCE

Semester : VIII Subject : RS Academic Year: 2024 - 20 25

(Ch 1)

⇒ Introduction to ~~Cont~~ Recommendation System.

* Recommendation System

Examples :-

① Youtube

② Ecommerce Website ~~Amazon~~

As per User Interest → eg⁴ given.

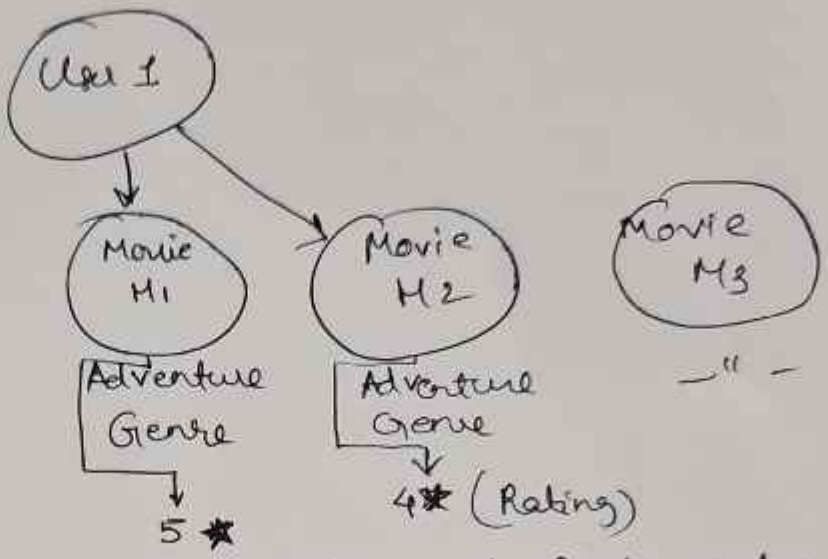
⇒ Gym → Order Best protein powder from a website Visit Again on Website – All offers, best prices, all product's types related to purchased protein powder. (Recommendations given employed)

⇒ In History of the website, all products purchased are seen. As per them, the recommendations are given further related to it (Amazon website)

⇒ To carry out Recommendation System

2 types → Content Based
→ Collaborative filtering

(1) Content Based.



⇒ Both with same genre with good ~~deal~~ rating.

⇒ This is called Content Based Similarity or Item Based Similarity.

⇒ Here, Movie is the Item with Adventure genre.

M_1, M_2 ⇒ 2 Items.

M_3 ⇒ Director's, Actor's, Genre are "same" of M_1, M_2 & M_3.

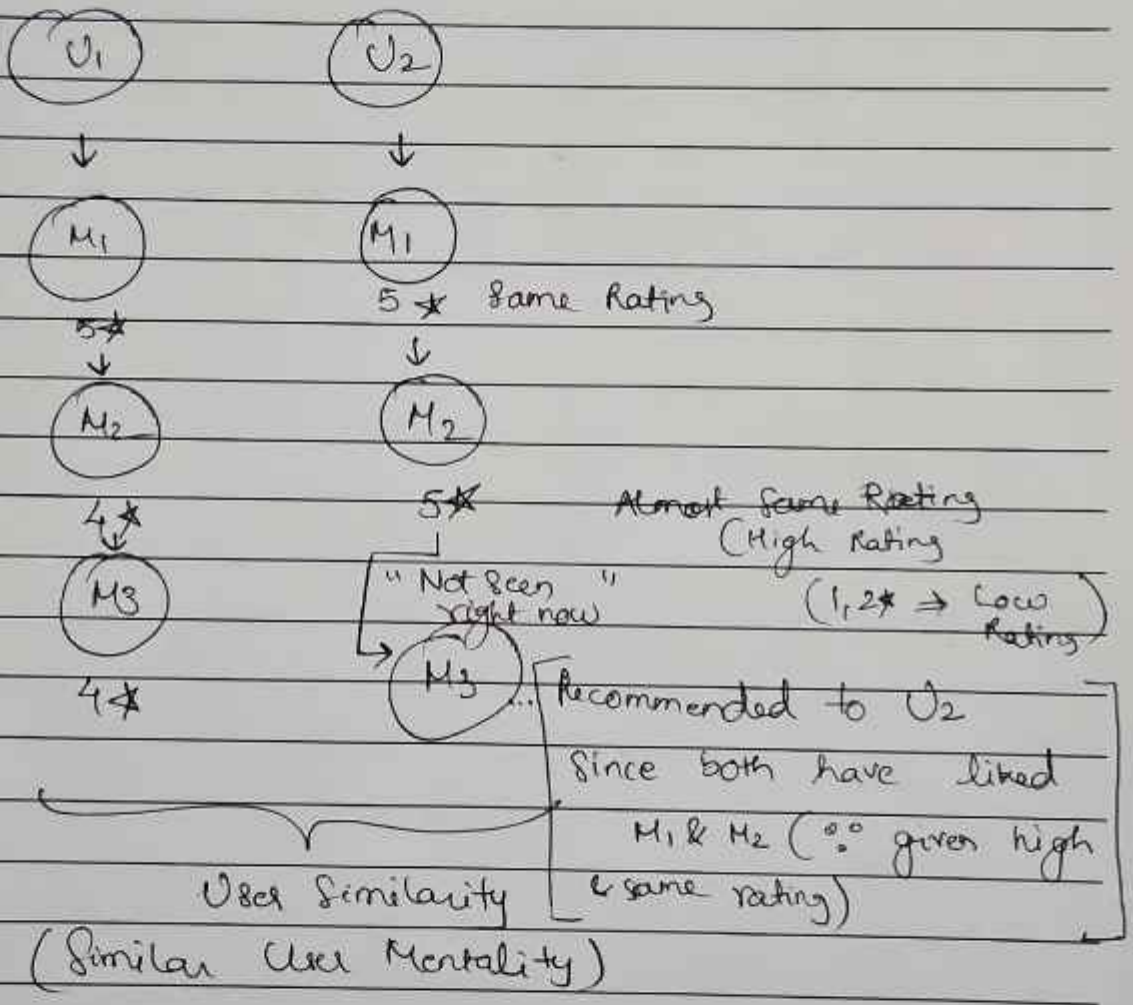
⇒ Therefore, we will recommend the new movie M_3 to user 1, based on Content Based Similarity / Item Similarity.

PARSHWANATH CHARITABLE TRUST'S
A.P. SHAH INSTITUTE OF TECHNOLOGY
Department of Computer Science and Engineering
Data Science

CSE DATA SCIENCE

Semester : _____ Subject : _____ Academic Year: 20 - 20

(2) Collaborative Filtering.



⇒ User 2 gives almost similar rating to Movie M3 after watching it.

⇒ [Content Based → Item Based Similarity
Collaborative Filtering → User Based Choices (User Mentality)]

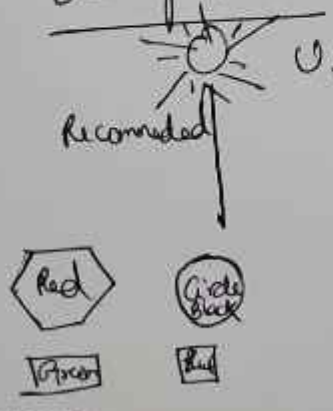
* Content Based Filtering (In Detail)
For Recommendation System

⇒ Definition

Recommend items to customer similar to previous items rates by x i.e. uses item features to recommend new items that are similar to what user has liked in the past.

Scrolling on a website → Liked a product on it.
→ Perform certain actions on site ⇒ Thus website recommends ~~sg~~ to user based on actions identified.

Example :-



Instagram
→ clicked on a Link/waiton a product/video
→ Implicit Signal
→ Likes/Purchased
→ Explicit S/g

build User Profile
Red Circle Triangle

Recommends
Match

Item Profile
Features liked by user

Semester : _____ Subject : _____ Academic Year: 20 - 20

* Implicit Signals
  - ↳ Click on a Link
  - ↳ Wait on a video eg; cat while scrolling to watch it.
  - ↳ Zoom on which place
  - ↳ etc.

→ * Instagram notes it with above implicit signals

* Explicit Signals
  - ↳ Liked a reel / video / picture
  - ↳ Purchased a product directly

Above both type of signals are used to recommend user's with desired items.

Pros :-

(1) No community required.

Cons :-

(2) Need of Content Description
Computer / Application / Website ⇒ needs product description

(2) New user face cold start.

Complete New on a website / application, no feeds on a website about likes / dislikes then the user might not get the actual recommendations as per his/ her likes / dislikes.

Semester : VIII Subject : RS Academic Year: 2024 - 2025

Collaborative Filtering

Definition :-

Recommend items to customer 'x' similar to previous item rated highly by 'y' i.e. uses item features to recommend new items that are similar to what the user has liked in the past.

eg.,
Netflix app → Actions performed (Like, Scroll, Wait) → notes likes → Identifies which users like similar movies with you → then recommends similar movies from group of people with similar likings. → because the user will like them also.

eg.,
NETFLIX

[Can be group of people also]

5★ Hangover

Hangover 5★

5★ Project x

Project x 5★

The Stoneman Murders 5★

Recommends

Pros:

1. Continuous learning required (Continuous formation of group of people required)

Cons:

1. Rating feedback is required
2. New items and users face a cold start.

It is used to identify relationships between pieces pieces of data.

It is frequently used in recommender systems to identify similarities between user data and items.

Eg.,

It means that if Users A and B both like Product A, and User B also likes Product B, then Product could be recommended to User A by the system.

Methodology :-

→ model keeps track of what product users like and their characteristics to see what similar users like & then make recommendations.

Semester : VIII Subject : RS Academic Year: 2024-2025

→ Product features should be given numerical values

Whenever possible, makes the model give more accurate decisions, features should be given numerical values instead of string kind of data. Once features are identified & assigned numerical values, data collection begins here.

⇒ 2 ways the model can identify whether or not a user enjoyed the product.
User can assigned numerical values or system can ask what what user likes about a particular product.
For eg., when a user uses/purchased a product, we can ask the user to give ratings to the product (1*, 2*, etc.)

⇒ Once user interests have been established, recommendations can be made.

Numerical Example :-

Let's look at an example of a model that only considers one feature to make recommendations

U \ P	Product 1	Product 2	Product 3	Product 4
User 1		1	1	✓
User 2	1			
User 3		✓	1	(1) ✓
User 4		✓	✓	(1)

Semester : VIII Subject : RS Academic Year: 20__ - 20__

Recommendation System (3 Buckets)

- COLLABORATIVE FILTERING
- CONTENT BASED
- HYBRID (Mix of Both)

Memory Based
- Item Based
- User Based

Model Based

(1) Memory Based

Here,

⇒ We see the historical behaviour of users and items, & we come to a conclusion that since this has happened in history, it will be repeated again.

⇒ " History repeated by user in future "

⇒ We try to fit a model, & predict the behaviour

of user, what will be user's affinity towards the product.

CF — Item
CF — User

base of CF :- User Item Interaction Matrix.

	Movies on Imdb →				
	I_1	I_2	I_3	I_4 ...	I_n
U_1	8	7	X	X	
U_2	X				
U_3	X	8	6		
U_4					

⇒ As a user, we do not rate every movie.
⇒ Matrix may therefore null values (blank values)
⇒ When there is such a matrix, it is called as "Sparse matrix".

Problem with CF approach :-
(1) We ~~have~~ may not have rating for every movie by every user (∴ it generates sparse matrix in presence of lot of users & lot of movies on Imdb.)
(2) This Rating is called Implicit Rating
Other Matrices like amazon or Flipkart, we see
(i) Time spent on product page

Semester : VIII Subject : RS Academic Year: 20__ - 20__

(ii) No. of times clicked on the particular product.

(iii) No. of times visited that particular product.

Thus, we capture user's affinity towards a product, which is our objective. And we fill the numbers in the matrix.

for eg.

	I_1	I_2
U_1	7	8 ✓
U_2	9	4
U_3	8	6 ✓
U_4	—	7

Steps
(1) Find similar users

$\frac{7+8}{2}$ = Supposed rating for I_1 movie by User(U_4) is (8)

⇒ What if we have 10 movies;
Then, Similar user likings, are considered as per say genre of movie, and these movies are recommended to the person searching them.

⇒ If there are 1 million users, there will be some similarity within all users (U)

⇒ One way to measure this similarity is called "Euclidean Distance".

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$\sqrt{(9-7)^2 + (4-8)^2}$$ ⇒ Similarity between 2 different users U_1 & U_2

⇒ Thus, we make recommendations based on 10 similar users using similarity measure & average %.

⇒ What if we want to give higher weightage recommendation?

U_4 is more similar to U_3 & U_1 based on magnitude.

What if we want to give more weightage to User 'U_3' since its magnitude is higher than 'U_1'.

$$\sum_{i=1}^{n} R_i / n \Rightarrow \left\{ \left(\frac{\text{Rating of user } i}{n} \right) \right\}$$ Average

$$\sum_{i=1}^{n} R_i \times W_i \Big/ \sum_{i=1}^{n} W_i \quad \cdots \left\{ \left(\text{Weighted Average} \right) \right\}$$

⇒ More similar user is given weight W_i & thus weighted average is calculated

Semester : VIII Subject : RS Academic Year: 2024-2025

⇒ Thus, more similar users are recommended the same movie based on similar ratings

⇒ If not aggressive raters,

	I_1	I_2	I_3	
U_1	1-4/3	2-5/3	1-4/3	2+1+2/3 = 4/3
U_2	9-25/3	8-25/3	8-25/3	25/3
U_3				

Out of 10

⇒ Here, we normalize the rating before computing the distance.

⇒ This is how user based CF is implemented in back.

⇒ * User Based CF :-

(1) Find similar users

(2) Compute the rating (taking

⇒ Then {Best rating of movie} ⇒ {recommend to user if he/she has not watched it}

⇒ * Item Based CF

(1) Find similar items rated by User(U)

(2) Out of 10 movies, which are the similar movies & how user(U) has rated them.

~~(ii) If 'U' have to make a~~

(ii) If a recommendation has to be made by system of a horror movie, the recommendation depends on how the user has rated the other horror movies.

⇒ These 2 are techniques under memory based.
⇒ Other way,
CF ⟹ Model Based CF.

⇒ We try to implement a ML model to predict the rating, & then problem of sparsity in sparse matrix is dealt with in model based CF.

* Note :-
⇒ If we have right data to model, then CF is easier to implement.