AI in Search & Recommendations

Prof. Tarachand Amgoth

Learning Objectives

- Understand the importance of search and recommendation in digital platforms.
- Explore collaborative and content-based filtering with real-world case studies.
- Learn to implement a basic recommender system using AI techniques.

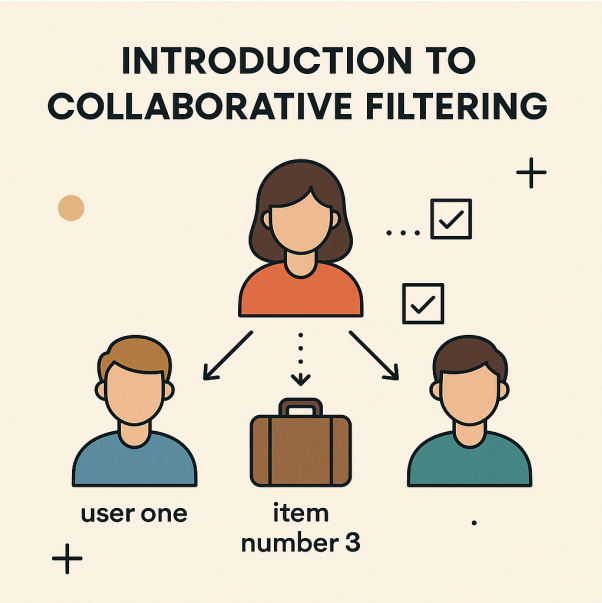

Figure 1: Users connected by shared preferences.

1 Why Do We Need Recommendations?

In the digital world, we are surrounded by choices. Be it movies on Netflix, products on Amazon, or songs on Spotify, the number of available items is overwhelming.

Imagine this: You walk into a massive grocery store with no signs, no staff, and everything randomly scattered. You want pasta, but you don't know where to look. How long before you give up?

Now replace the store with a website. This is where **recommendation systems** come in. They help:

- Users discover relevant content quickly.
- Businesses retain users longer and increase engagement.
- Platforms personalize experiences and generate more revenue.

2 What is a Recommender System?

A **recommender system** is a tool or algorithm that suggests items to users based on some form of data. The most common types are:

- **Collaborative Filtering** – based on user interactions and similarities.
- **Content-Based Filtering** – based on features of items and user preferences.
- **Hybrid Approaches** – combining the above methods for improved results.

3 Collaborative Filtering: The Core Idea

Definition

Collaborative filtering uses the preferences of a group of users to make predictions about what a particular user might like. It assumes that if users have agreed in the past, they will agree in the future.

Intuition Behind It: A Story

Think of it like a Netflix movie night with friends. Suppose you and your best friend have very similar movie tastes. If she loved a movie you haven't seen yet, you'll probably enjoy it too. That's the magic of collaborative filtering — **using similar people to make smart guesses.**

Main Assumptions

- People who agreed in the past will agree in the future.
- Users prefer items that similar users have liked.

4 Types of Collaborative Filtering

4.1 User-Based Collaborative Filtering

This approach recommends items based on what similar users have liked.

Example

User A and User B both liked items 1 and 2. If User B liked item 3, then User A might like item 3 too.

Mathematical Formula:

$$\hat{r}_{u,i} = \frac{\sum_{v \in N(u)} \text{sim}(u, v) \cdot r_{v,i}}{\sum_{v \in N(u)} \text{sim}(u, v)}$$

Where:

- $\hat{r}_{u,i}$ is the predicted rating for user u on item i
- $N(u)$ is the set of nearest neighbors to user u
- $\text{sim}(u, v)$ is the similarity between users u and v
- $r_{v,i}$ is the actual rating of item i by user v

4.2 Item-Based Collaborative Filtering

This approach recommends items similar to what the user already likes.

Example

If a user likes a specific book, the system recommends other books that are similar to it in terms of who liked them.

Mathematical Formula:

$$\hat{r}_{u,i} = \frac{\sum_{j \in N(i)} \text{sim}(i,j) \cdot r_{u,j}}{\sum_{j \in N(i)} \text{sim}(i,j)}$$

Where:

- $N(i)$ is the set of similar items to item i
- $\text{sim}(i,j)$ is the similarity between items i and j
- $r_{u,j}$ is the rating of item j by user u

5 A Classroom Example: User One and Item Three

During the lecture, Prof. Amgoth introduced a basic example:

- **User 1** has shown interest in items 1 and 2.
- **Item 3** is favored by users who also liked items 1 and 2.
- Hence, the system recommends **Item 3** to User 1.

This is a basic but powerful illustration of how collaborative filtering leverages community data.

6 Challenges in Collaborative Filtering

1. Cold Start Problem

- What if a new user joins the system?
- Or a new item is added with no ratings?
- Solution: Use hybrid or content-based methods for new users/items.

2. Data Sparsity

- Most users only interact with a small portion of items.
- The user-item matrix is often very sparse.

3. Scalability

- Large platforms have millions of users and items.
- Computing similarities for all pairs becomes expensive.

7 What Comes Next?

The lecture is expected to cover:

- **Content-Based Filtering** – Using item features to make recommendations.
- **Case Studies** – How companies like Amazon and Spotify use recommendations.
- **Hands-On** – Building a basic recommender system using Python.

Conclusion

Collaborative filtering is a brilliant idea — instead of figuring out what an item is, it focuses on who likes it. It mimics how humans naturally give advice: "Hey, you liked that restaurant? I trust your taste, so I'll try it too."

In a world full of noise and choices, recommender systems are the quiet guide whispering, "Try this — you might like it."