Recommender Systems

Helping Users Discover What They'll Love Next

Introduction to Recommender Systems

Imagine walking into a giant bookstore with millions of titles. Without guidance, you'd spend hours picking books. A **recommender system** acts like a friendly librarian who, based on your past reads or preferences, suggests the next book you'll enjoy. In online platforms—Netflix, Amazon, Spotify—recommenders personalize content, improving user experience and engagement.

Why it matters:

- **Personalization**: Tailors options to individual tastes.
- **Discovery**: Surfaces relevant items users might not find alone.
- **Business value**: Boosts sales, retention, and satisfaction.

Recommender systems broadly fall into two categories:

1. **Collaborative Filtering**: Uses patterns of user-item interactions (e.g., ratings).
2. **Content-Based Filtering**: Uses item attributes and user profiles.

Short Example: User-Item Rating Matrix (Collaborative Filtering)

Suppose we have 4 users and 5 movies. Ratings range from 1 (dislike) to 5 (love):

	Movie A	Movie B	Movie C	Movie D	Movie E
User 1	5	3	?	1	?
User 2	4	?	2	1	?
User 3	?	2	5	4	3
User 4	1	1	2	?	4

To predict User 1's rating for Movie C:

1. Compute similarity between User 1 and other users (e.g., cosine similarity).
2. Weight neighbors' ratings by similarity.
3. Estimate missing rating.

Key idea: "Users who liked what you liked tend to agree on other movies too."

Discussing the Output

After computing similarities and weighted sums, you might predict:

- **User 1's rating for Movie C** ≈ 4.2 (on a scale of 1–5).
- You can then recommend the top unseen movie with highest predicted rating.

This simple collaborative approach leverages community tastes to fill gaps in a user's profile.

Reflection and Best Practices

Key Takeaways:

- **Collaborative Filtering** finds like-minded users or items.
- **Content-Based Filtering** uses item features (genres, keywords) to match user profiles.
- **Hybrid approaches** combine both for better accuracy.

Common Pitfalls:

- **Cold start**: New users or items have no data—hard to recommend.
- **Sparsity**: Rating matrices are mostly empty—similarity estimates become noisy.
- **Popularity bias**: Over-recommending popular items, ignoring niche interests.

Real-World Applications:

- **E-commerce**: Product recommendations on Amazon.
- **Streaming**: Movie suggestions on Netflix, playlist generation on Spotify.
- **Social networks**: Friend or content recommendations on Facebook and LinkedIn.

This document offers a concise, beginner-friendly overview of recommender systems. Download the PDF for a ready-to-publish guide.