بسم الله الرحمن الرحیم

Information Technology Fundamentals

Mohammad Hossein Manshaei

manshaei@gmail.com

Web Systems: Collaborative Filtering Recommendation : Memory-Based Module 5: Part 5

Module 5. Main Objectives

1. Review Web System Architecture
2. Explain E-Commerce Business Models
3. **Review Recommender Systems**
4. Describe Blockchain Systems, Cryptocurrency, and Smart Contracts

Recommendation Techniques

Main Reference

Kim Falk. ***Practical Recommender Systems,*** Manning Publication Co., 2019

SOFTWARE DEVELOPMENT/MACHINE LEARNING

Practical Recommender Systems

Kim Falk

Free eBook
See first page

Online recommender systems help users find movies, jobs, restaurants—even romance! There's an art in combining statistics, demographics, and query terms to achieve results that will delight them. Learn to build a recommender system the right way: it can make or break your application!

Practical Recommender Systems explains how recommender systems work and shows how to create and apply them for your site. After covering the basics, you'll see how to collect user data and produce personalized recommendations. You'll learn how to use the most popular recommendation algorithms and see examples of them in action on sites like Amazon and Netflix. Finally, the book covers scaling problems and other issues you'll encounter as your site grows.

What's Inside

- How to collect and understand user behavior
- Collaborative and content-based filtering
- Machine learning algorithms
- Real-world examples in Python

Readers need intermediate programming and database skills.

Kim Falk is an experienced data scientist who works daily with machine learning and recommender systems.

To download their free eBook in PDF, ePub, and Kindle formats, owners of this book should visit manning.com/books/practical-recommender-systems

"Covers the technical background and demonstrates implementations in clear and concise Python code."
—Andrew Collier, Exegetic

"Have you wondered how Amazon and Netflix learn your tastes in products and movies, and provide relevant recommendations? This book explains how it's done!"
—Amit Lamba, Tech Overture

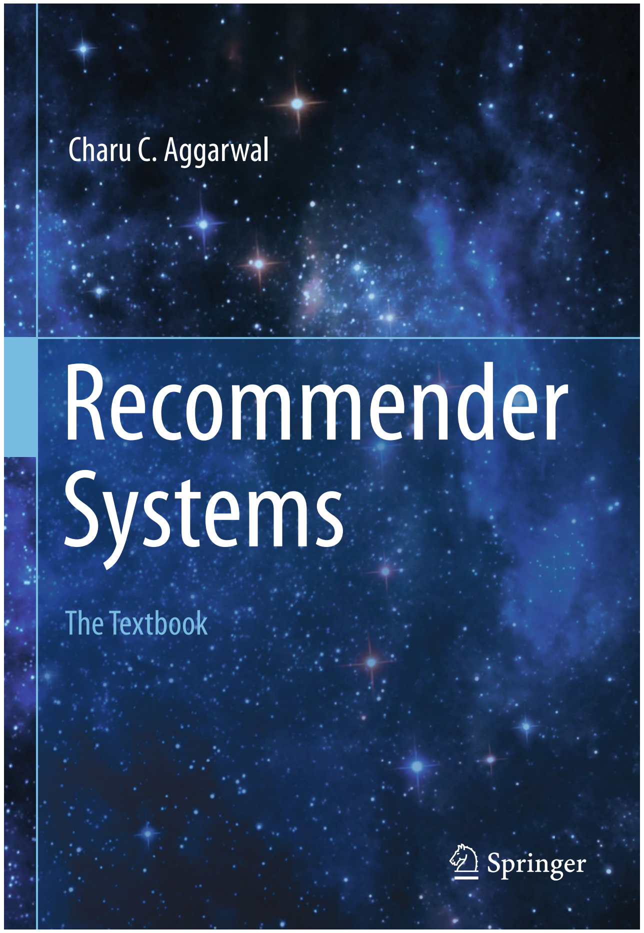
"Everything about recommender systems, from entry-level to advanced concepts."
—Jaromir D.B. Němec, DBN

"A great and practical deep dive into recommender systems!"
—Peter Hampton
Ulster University

Main Reference

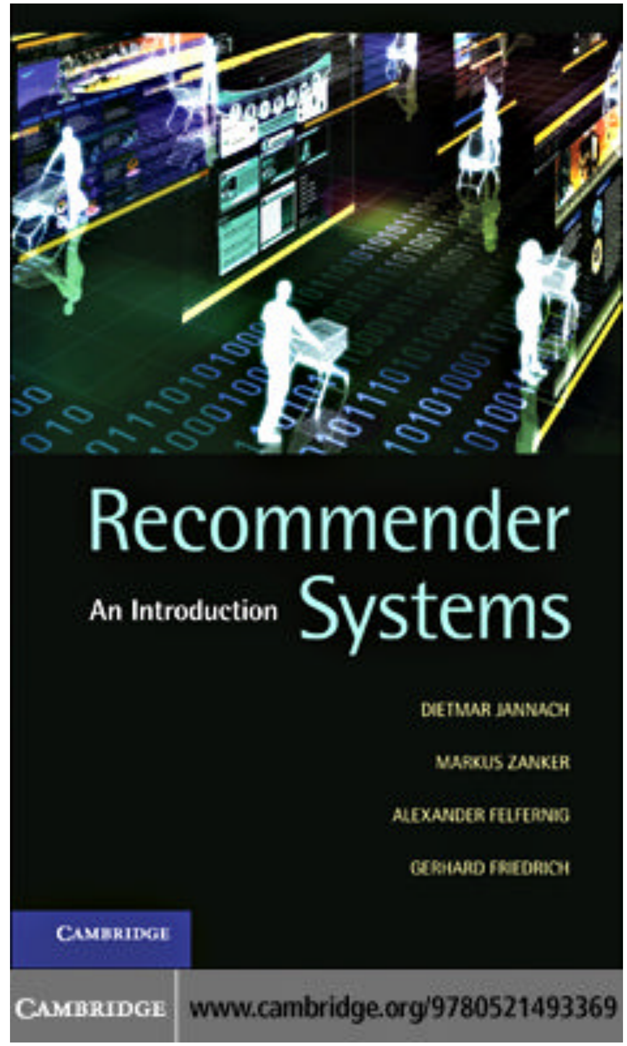
Charu C. Aggarwal. ***Recommender Systems, The Textbook.*** Springer, 2016

Chapter 1: An Introduction to Recommender Systems



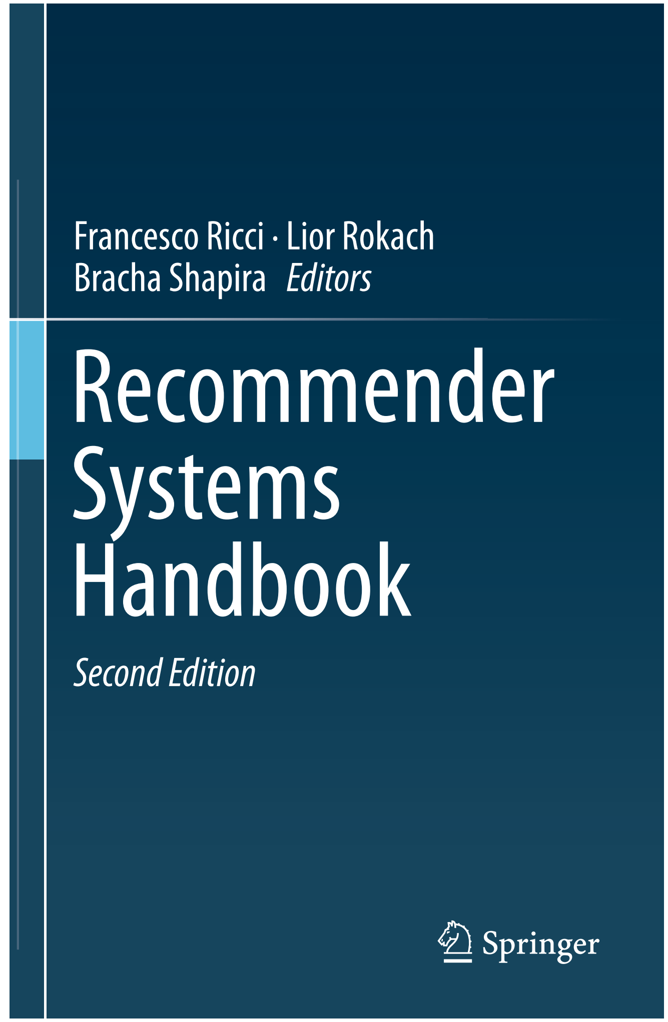
Main Reference

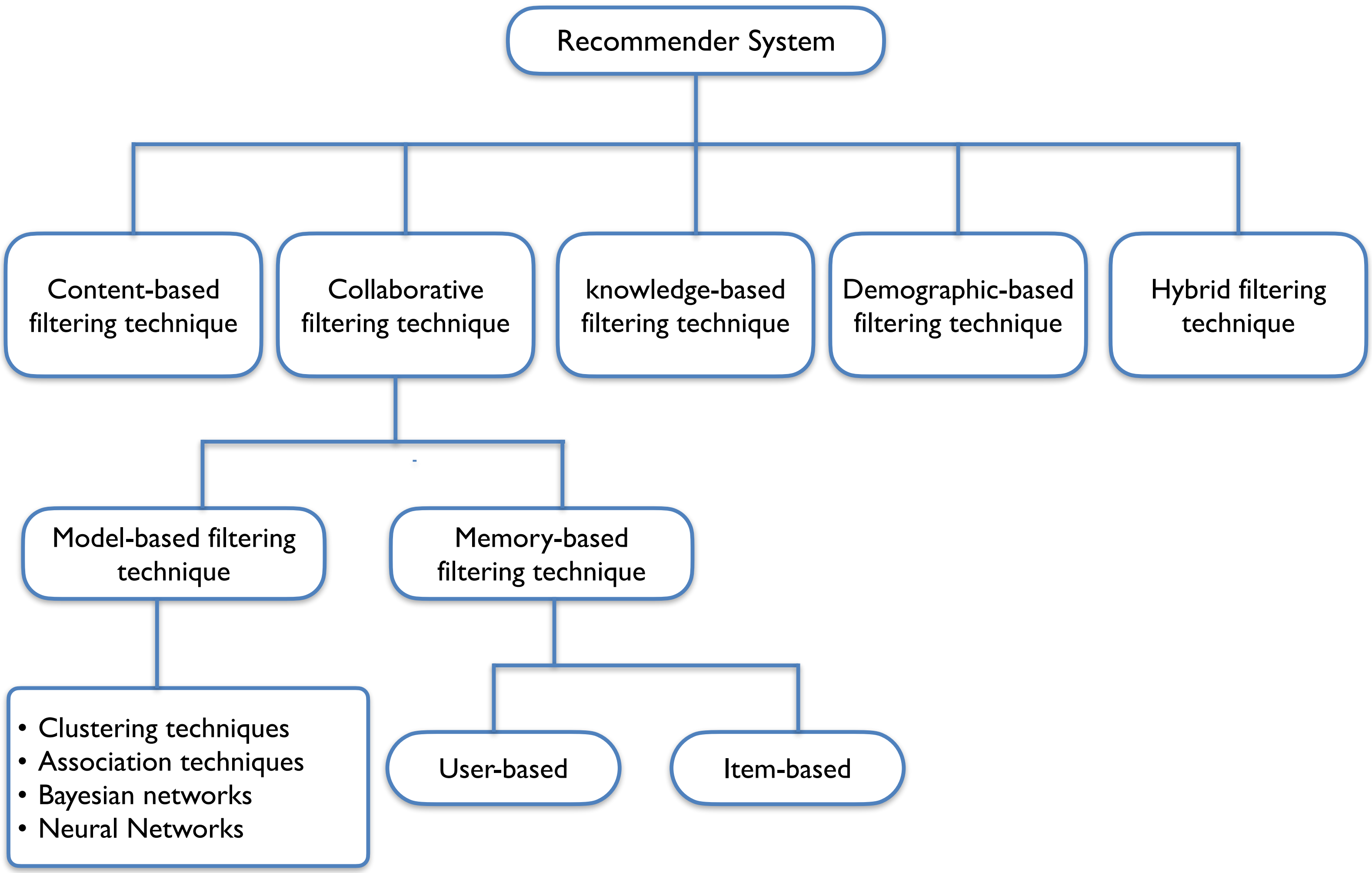
D. Jannach, M. Zanker, A. Felfering, G. Friedrich. ***Recommender Systems: An Introduction.*** Cambridge University Press, 2011



Main Reference

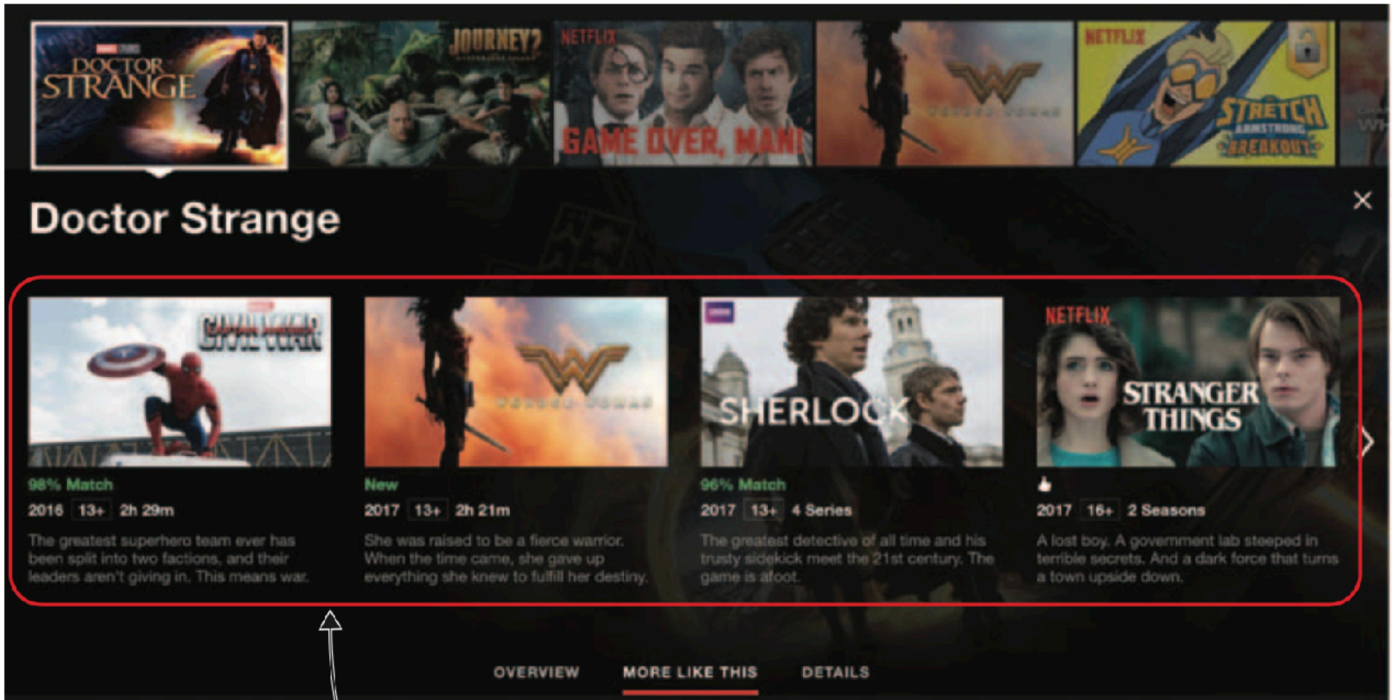
F. Ricci,L. Rokach, B. Shapira. ***Recommender Systems Handbook.*** Springer, 2015




Recommender System
Content-based filtering technique
Collaborative filtering technique
knowledge-based filtering technique
Demographic-based filtering technique
Hybrid filtering technique
Model-based filtering technique
Memory-based filtering technique
• Clustering techniques
• Association techniques
• Bayesian networks
• Neural Networks
User-based
Item-based

Contents

- Collaborative Filtering Recommendation
  - ✓ Memory-Based
    - User-Based and Item-Based
  - ✓ Model-Based
    - Matrix Factorization/latent factor models and Association Rules
- Content-Based Recommendation
- Knowledge Based Recommendation
- Demographic Recommendation
- Hybrid Recommendation Systems
- RS Evaluation
  - ✓ Online and Offline Evaluations
  - ✓ Design Issues and RS Properties

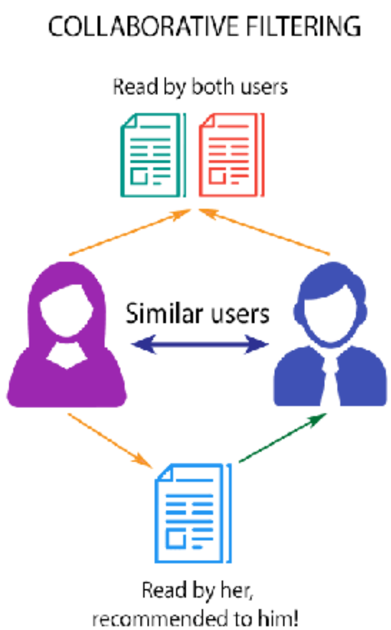


In Netflix you can get *More Like This* recommendations, which is probably found by calculating similarity.

"More Like This" personalized recommendations on Netflix based on the TV series The Flash

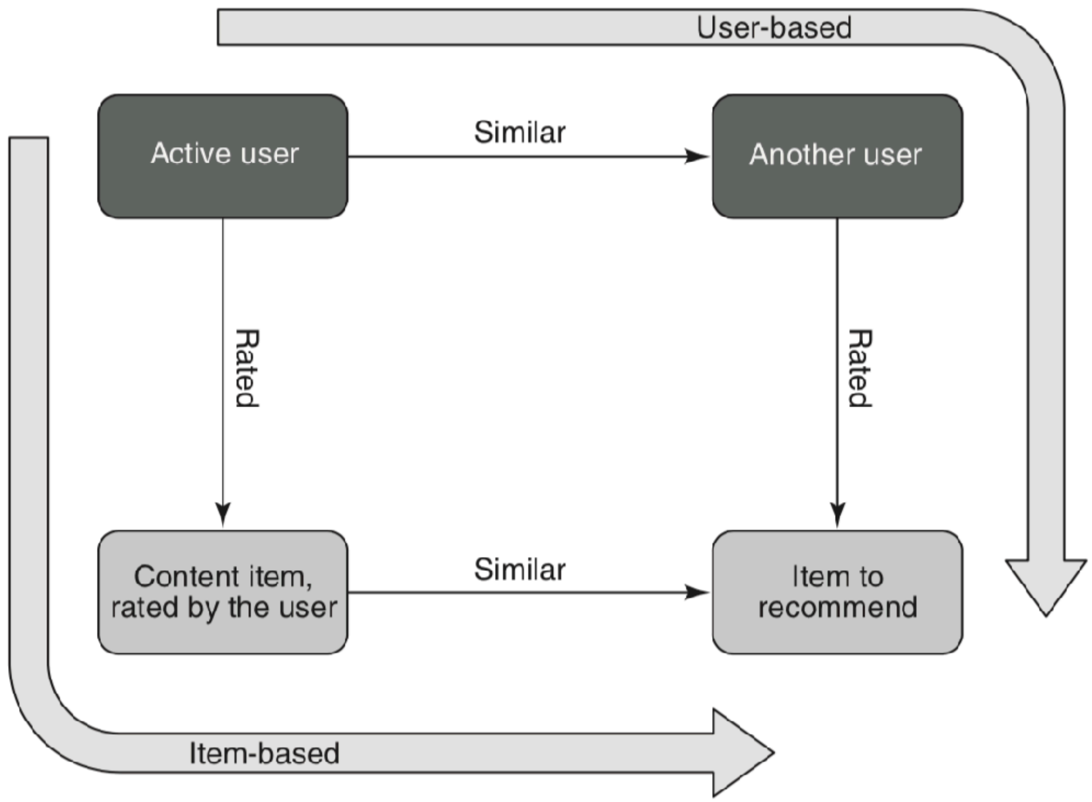
Collaborative Filtering

Works by building a database **(user-item matrix)** of preferences for items by users. It then matches users with relevant interest and preferences by **calculating similarities** between their profiles to make recommendations.



Introduction to Similarity Functions

- Given two items, i_1 and i_2, the similarity between them is given by $Sim(i_1, i_2)$.
  - This function's return values will increase the more similar the items are
  - We can say that the similarity between the same item is $Sim(i1, i1) = 1$
  - Two items that have nothing in common will be $Sim(i1, \text{nothing in common with } i1) = 0$.
- Generally you can say that the relationship between similarity and distance is the following:
  - When distance gets larger, the similarity goes toward zero.
  - When distance goes toward zero, the similarity goes toward one.



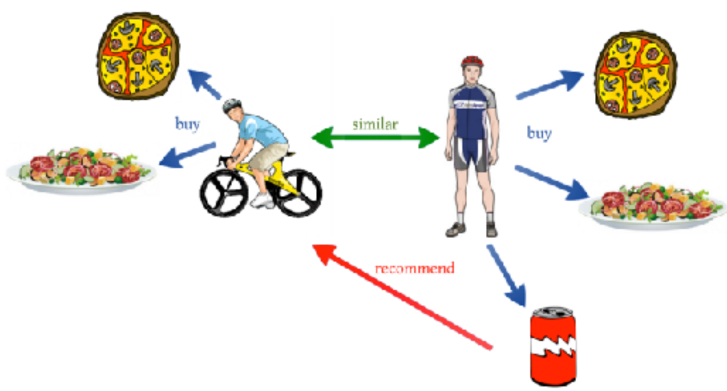
The two ways of performing "neighborhood-based filtering".
One method uses similar users, while the other uses items similar to items the active user liked.

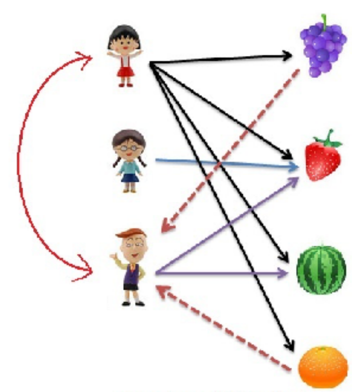
Contents

- ➢ Collaborative Filtering Recommendation
  - ✓ Memory-Based:
    - User-Based and Item-Based
  - ✓ Model-Based
    - Matrix Factorization/latent factor models and Association Rules
- ➢ Content-Based Recommendation
- ➢ Knowledge Based Recommendation
- ➢ Demographic Recommendation
- ➢ Hybrid Recommendation Systems
- ➢ RS Evaluation
  - ✓ Online and Offline Evaluations
  - ✓ Design Issues and RS Properties

User-User Collaborative Filtering

1. Given a ratings database and the *ID* of the current (active) user as an input, identify other users (sometimes referred to as peer users or nearest neighbors) that had similar preferences to those of the active user in the past.
2. For every product p that the active user has not yet seen, a prediction is computed based on the ratings for p made by the peer users.





User-based filtering

Ratings Database for Collaborative Recommendation

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

- This table shows a database of ratings of the current user, Alice, and some other users.

Pearson Correlation Coefficient

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

- The similarity of users a and b (Pearson correlation coefficient or covariance divided by standard deviation):

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_{p \in P}(r_{a,p} - \overline{r_a})^2}\sqrt{\sum_{p \in P}(r_{b,p} - \overline{r_b})^2}}$$

Comparing Alice with Other Users

- The similarity of Alice to User1:

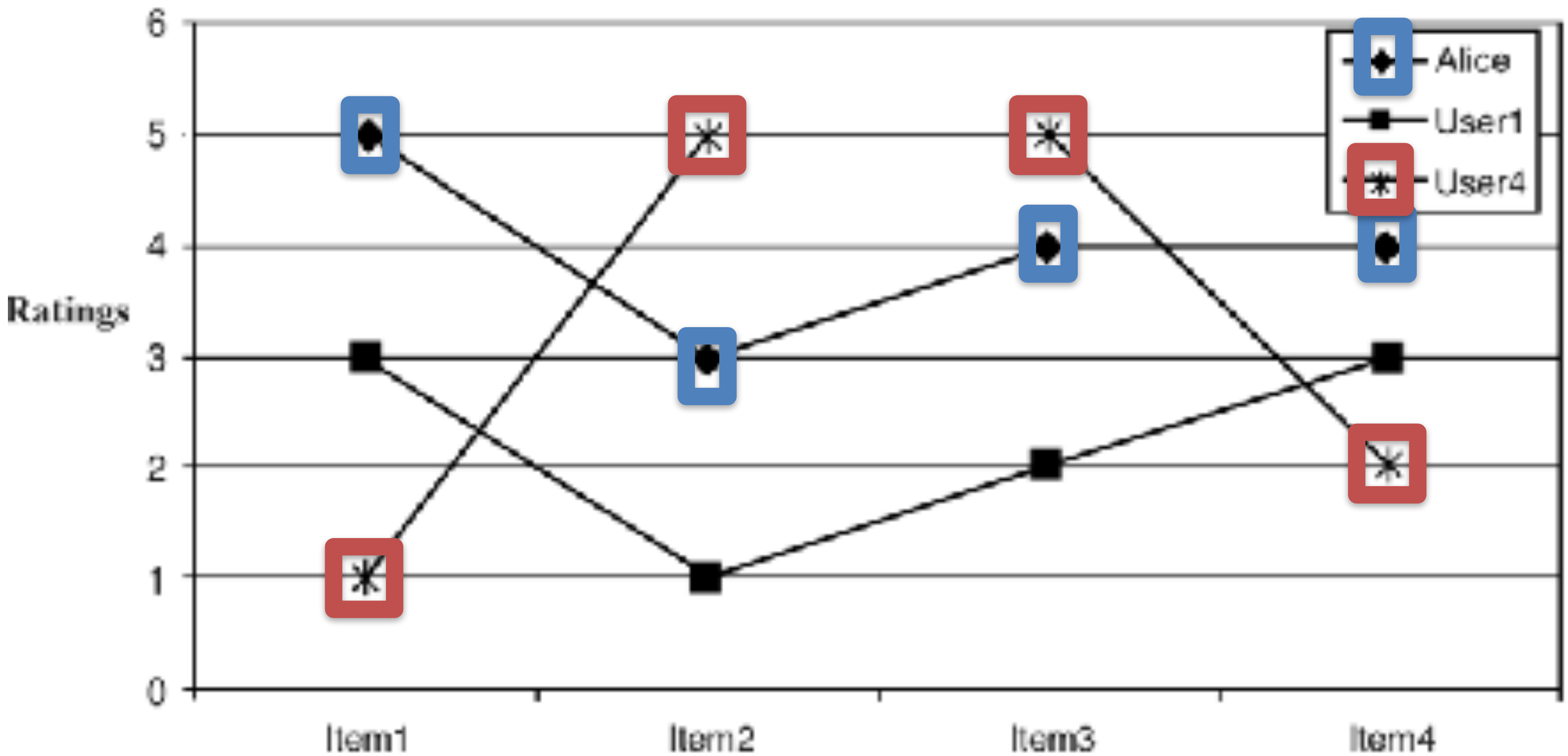
$$\overline{r_{Alice}} = \overline{r_a} = 4$$

$$\bar{r}_{User1} = r_b = 2.4$$

$$\frac{(5-\overline{r_a})*(3-\overline{r_b})+(3-\overline{r_a})*(1-\overline{r_b})+\cdots+(4-\overline{r_a})*(3-\overline{r_b}))}{\sqrt{(5-\overline{r_a})^2+(3-\overline{r_a})^2+\cdots}\sqrt{(3-\overline{r_b})^2+(1-\overline{r_b})^2+\cdots}} = 0.85$$

- The similarity of Alice to User2: 0.70
- The similarity of Alice to User3: 0
- The similarity of Alice to User4: -0.79

Comparing Alice with User 4 (Sim = -0.79)

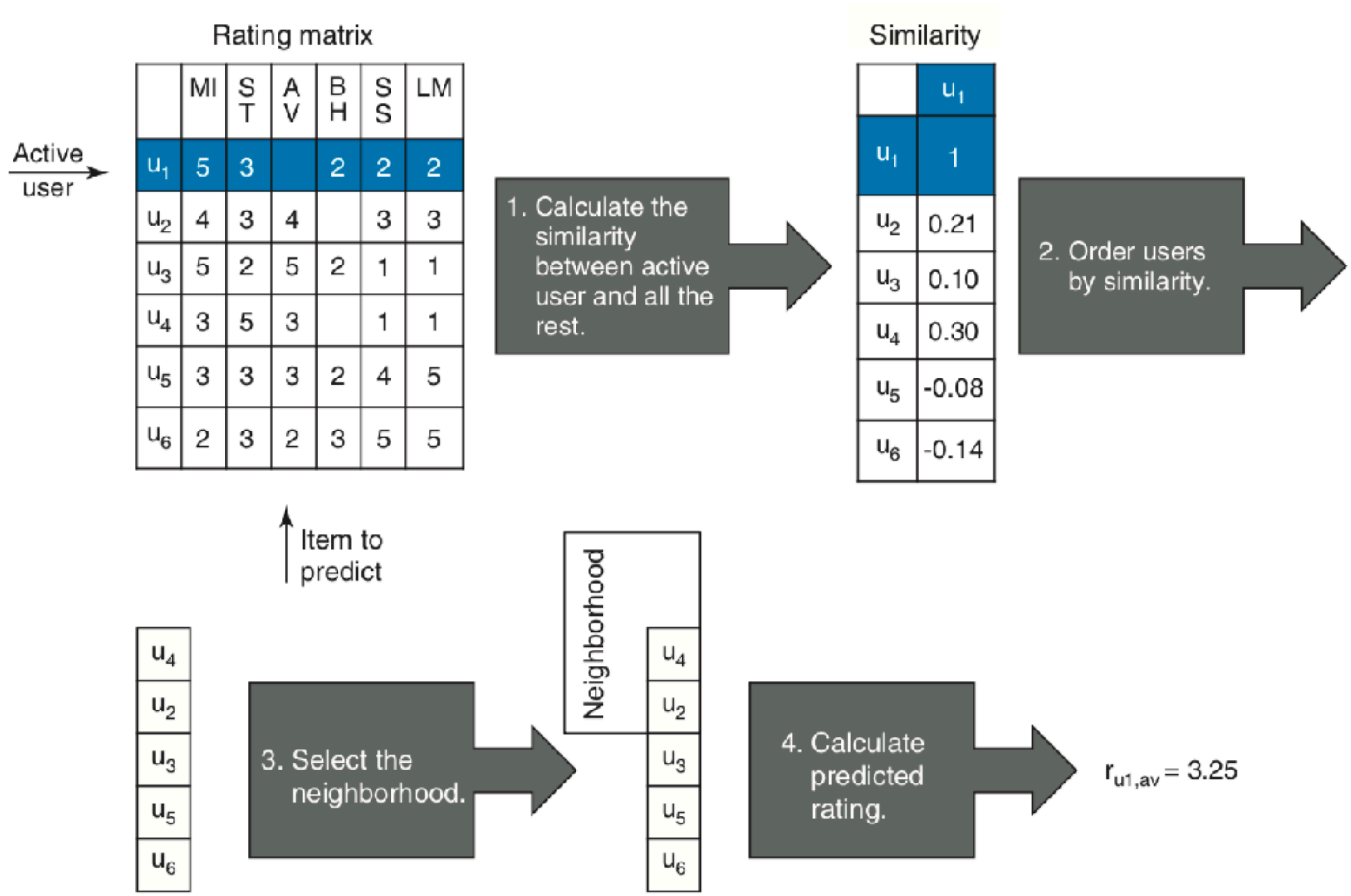


Predict Alice's Rating for Item 5

- We select User 1 and 2 (similarity of 0.85 and 0.7)

$$pred(a,p) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)}$$

$$4 + 1/(0.85 + 0.7) * (0.85 * (3 - 2.4) + 0.70 * (5 - 3.8)) = 4.87$$

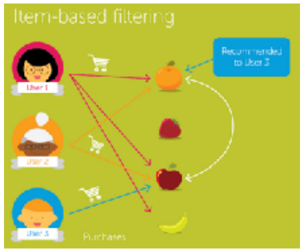


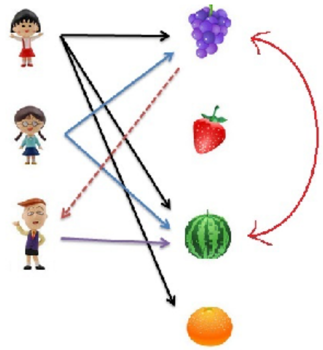
The user-based filtering pipeline

Item-Item Collaborative Filtering

Compute predictions using the similarity between items and not the similarity between users.

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1





Item-based filtering

The Cosine Similarity Measure

$$sim(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\, \vec{a} \,| * |\, \vec{b} \,|}$$

$$sim(I5, I1) = \frac{3*3+5*4+4*3+1*1}{\sqrt{3^2+5^2+4^2+1^2} * \sqrt{3^2+4^2+3^2+1^2}} = 0.99$$

- The adjusted cosine similarity value for Item5 and Item1:

$$sim(a, b) = \frac{\sum_{u \in U}(r_{u,a} - \overline{r_u})(r_{u,b} - \overline{r_u})}{\sqrt{\sum_{u \in U}(r_{u,a} - \overline{r_u})^2}\sqrt{\sum_{u \in U}(r_{u,b} - \overline{r_u})^2}}$$

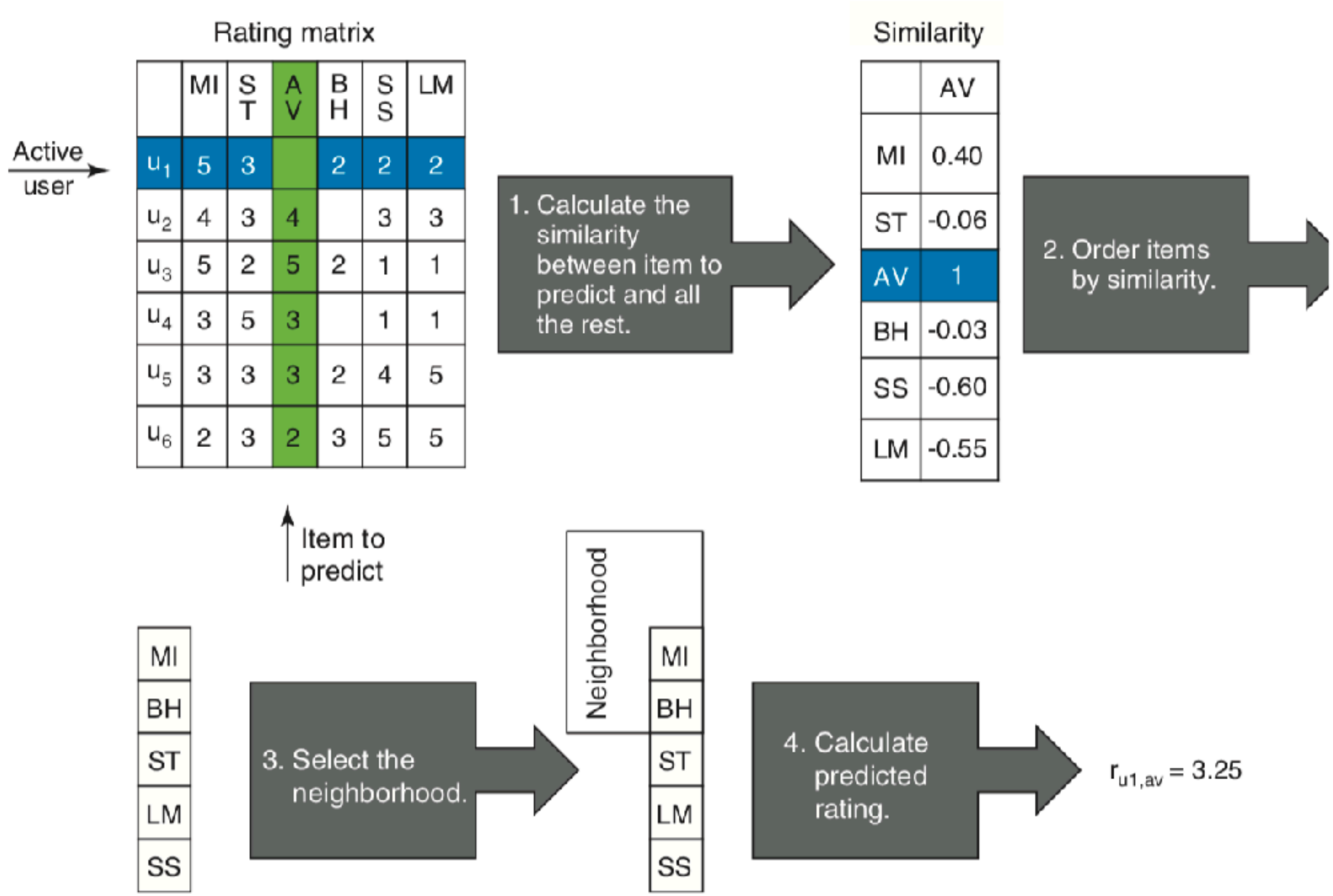
$$\frac{0.6*0.6+0.2*1.2+(-0.2)*0.80+(-1.8)*(-1.8)}{\sqrt{(0.6^2+0.2^2+(-0.2)^2+(-1.8)^2} * \sqrt{0.6^2+1.2^2+0.8^2+(-1.8)^2}} = 0.80$$

	Item1	Item2	Item3	Item4	Item5
Alice	1.00	−1.00	0.00	0.00	?
User1	0.60	−1.40	−0.40	0.60	0.60
User2	0.20	−0.80	0.20	−0.80	1.20
User3	−0.20	−0.20	−2.20	2.80	0.80
User4	−1.80	2.20	2.20	−0.80	−1.80

Mean-adjusted ratings database

Item-to-item collaborative filtering is the technique used by Amazon.com to recommend books or CDs to their customers.a

$$pred(u, p) = \frac{\sum_{i \in ratedItems(u)} sim(i, p) * r_{u,i}}{\sum_{i \in ratedItems(a)} sim(i, p)}$$



The item-based filtering pipeline