Neighborhood based Collaborative Filtering

Jiho Kang
2022.04.01

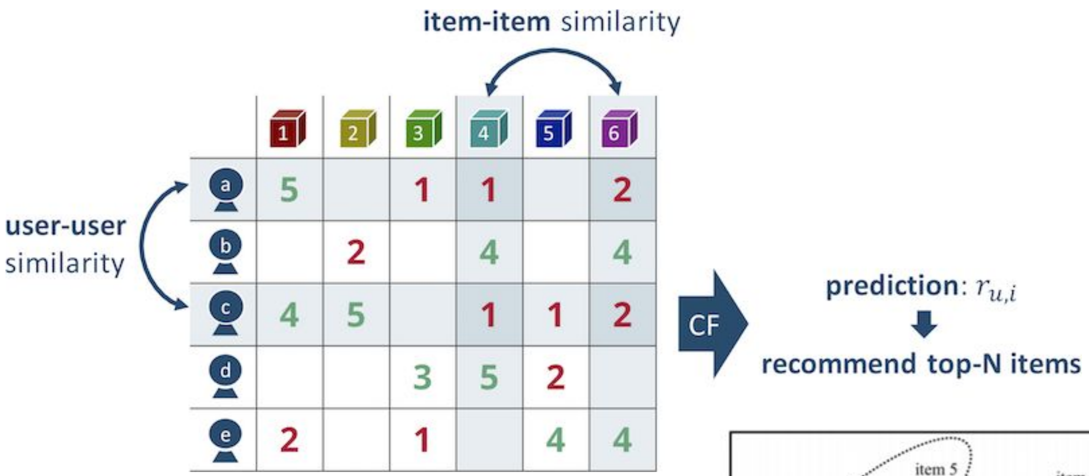
1. Neighborhood based method

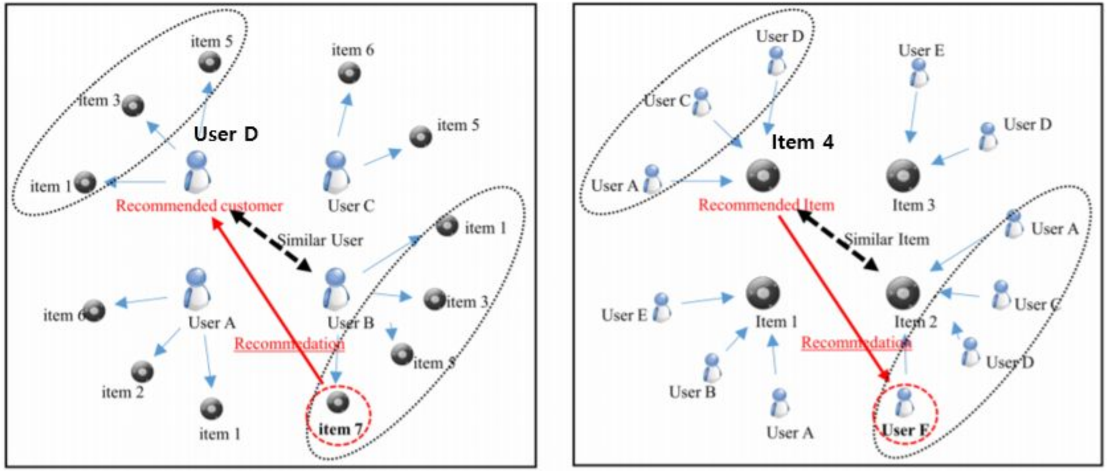
- **What is Neighborhood based Collaborative Filtering?**
  - Uses the rating data by the user on the item to make predictions for not yet scored item.
  - Matrix(user*item) with score and empty values.
  - **By filling in blank values**, high prediction ratings are recommended.
  - Early algorithm developed for collaborative filtering as a **memory-based** algorithm.
- **Algorithm**
  - User-based CF

    Find a user similar to the user's purchase pattern (rating) and create a recommendation list.
  - Item-based CF

    Find similar products between scores given by a particular user and create a recommendation list.

1. Neighborhood based method





(A) (B)

2. Algorithm - user-based

cosine, Pearson

	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	mean	Pearson(i,3)
User 1	7	6	7	4	5	4	5.5	0.894
User 2	6	7		4	3	4	4.8	0.939
User 3	**A**	3	3	1	1	**B**	2	1.0
User 4	1	2	2	3	3	4	2.5	-1.0
User 5	1		1	2	3	3	2	-0.817

$$P_{u,i} = \frac{\sum_{\text{all similar items, N}} (s_{i,N} * R_{u,N})}{\sum_{\text{all similar items, N}} (|s_{i,N}|)}$$

S(i,N) = similarity of user
R(u,N) = score of (user u, item N)

2. Algorithm - user-based

cosine, Pearson

	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	mean	Pearson(i,3)
User 1	7	6	7	4	5	4	5.5	0.894
User 2	6	7		4	3	4	4.8	0.939
User 3	**A 6.49**	3	3	1	1	**B 4**	2	1.0
User 4	1	2	2	3	3	4	2.5	-1.0
User 5	1		1	2	3	3	2	-0.817

$$P_{u,i} = \frac{\sum_{\text{all similar items, N}} (s_{i,N} * R_{u,N})}{\sum_{\text{all similar items, N}} (|s_{i,N}|)}$$

$$A = \frac{7\text{x}0.894 + 6\text{x}0.939}{0.894 + 0.939} = 6.49$$

S(i,N) = similarity of user
R(u,N) = score of (user u, item N)

$$B = \frac{4\text{x}0.894 + 4\text{x}0.939}{0.894 + 0.939} =4$$

2. Algorithm - user-based

cosine, Pearson

	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6	mean	Pearson(i,3)
User 1	7	6	7	4	5	4	5.5	0.894
User 2	6	7		4	3	4	4.8	0.939
User 3	**A 3.35**	3	3	1	1	**B 0.86**	2	1.0
User 4	1	2	2	3	3	4	2.5	-1.0
User 5	1		1	2	3	3	2	-0.817

$$A = \frac{7 \text{x} 0.894 + 6 \text{x} 0.939}{0.894 + 0.939} = 6.49$$

$$A = 2 + \frac{1.5 \text{x} 0.894 + 1.2 \text{x} 0.939}{0.894 + 0.939} = 3.35$$

$$B = \frac{4 \text{x} 0.894 + 4 \text{x} 0.939}{0.894 + 0.939} = 4$$

$$B = 2 + \frac{-1.5 \text{x} 0.894 + -0.8 \text{x} 0.939}{0.894 + 0.939} = 0.86$$

2. Algorithm - item-based

	Item 1	Item 2	Item 3	Item 4	Item 5	Item 6
User 1	7	6	7	4	5	4
User 2	6	7		4	3	4
User 3	**A 3**	3	3	1	1	**B 1**
User 4	1	2	2	3	3	4
User 5	1		1	2	3	3
Cosine(1,j)	1	0.735	0.912	-0.848	-0.813	-0.990
Cosine(6,j)	-0.990	-0.622	-0.912	0.829	0.730	1

$$\hat{r}_{31} = \frac{3 * 0.735 + 3 * 0.912}{0.735 + 0.912} = 3$$

$$\hat{r}_{36} = \frac{1 * 0.829 + 1 * 0.730}{0.829 + 0.730} = 1$$

3. Pros & Cons

- **Pros**
  - Because of a simple and intuitive approach, its easy to implement and debug.
  - It is easy to explain the reason for recommending a specific ITEM, and the interpretability of ITEM-based methods stands out.
  - Relatively stable even with new items and users added to the recommendation list.
- **Cons**
  - User-based methods require a lot of time, speed, and memory.
  - Limited range of items due to scarcity.
    - It is recommended only for items that are selected a lot.
    - If none of A's neighbors evaluates Harry Potter, A cannot be provided a rating forecast for Harry Potter.

References

- T아카데미 | 추천시스템 분석 입문하기
- https://www.cs.carleton.edu/cs_comps/0607/recommend/recommender/itembased.html
- https://pearlluck.tistory.com/667
- https://hackersstudy.tistory.com/126