Neighborhood-based Collaborative Filtering

Introduction

- The big drawbacks of Content-based is building the model which does not depend on users and only depends on items in one user, there are 2 drawbacks:
  - The approach cannot utilize the users' vector in user-item matrix which can show the correlation among the users
  - We have to build the profile for each items which is sometimes hard to build or cannot build
- Neighborhood-based approach can solve the 2 problems:
  - The approach utilizes the user-item matrix to predict the missing ratings
  - We don't need to build the profile for each item
- Neighborhood-based Collaborative Filtering: the ratings of user U on item I is based on the ratings of users, which is almost similar to user U, on item I
  - User A and B likes series "Canh Sat Hinh Su" due to ratings of 5 on this series. Ratings of user A on series "Nguoi Phan Xu" is 5, so it's more likely to let rating of user B on series "Nguoi Phan Xu" be 5
- Based on main idea of NBCF, we need to solve 2 problems
  - How to know the users which is similar to user B?
  - After getting these users, how to predict the missing rating based on these users?
- 2 types of NBCF:
  - User-User CF: Predict missing ratings of user B on item I based on other users (which users are similar to user B)
  - Item-Item CF: Predict missing ratings of user B on item I based on other items (which items are similar to item I)

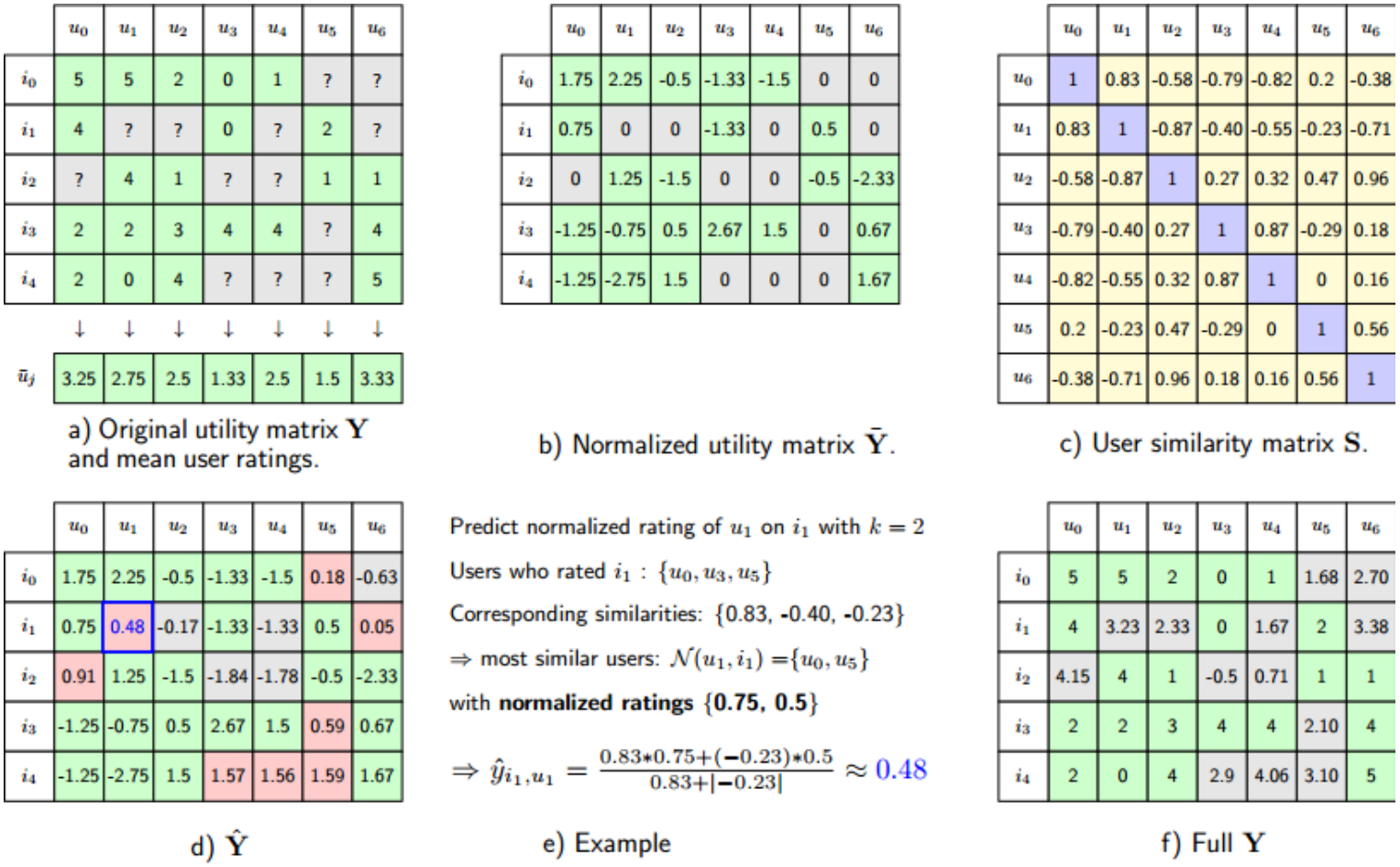
User-User Collaborative Filtering

	u_0	u_1	u_2	u_3	u_4	u_5	u_6
i_0	5	5	2	0	1	?	?
i_1	3	?	?	0	?	?	?
i_2	?	4	1	?	?	1	2
i_3	2	2	3	4	4	?	4
i_4	2	0	4	?	?	?	5

Hình 18.1: Ví dụ về utility matrix dựa trên số sao một *user* đánh giá một *item*. Một cách trực quan, *hành vi* của u_0 giống với u_1 hơn là u_2, u_3, u_4, u_5, u_6. Từ đó có thể dự đoán rằng u_0 sẽ quan tâm tới i_2 vì u_1 cũng quan tâm tới *item* này.

- Build the similarity matrix among users
  - Which values should missing ratings "?" be replaced?
    - One option is **2.5**, but this options equates the mean and generous users. The generous users may rate 3 for items they don't like, but the mean users may rate 3 for items they really like, so this options may be not so good
    - Another option is mean of ratings in one user, this option may solve problem of mean and generous users
    - Another option is **0** and normalize the utility (user-item) matrix by minus user vector to mean of ratings of user, the subtraction may solve the difference between the mean and generous users
  - We use the cosine similarity of determine: $sim(u_1, u_2) = \cos(u_1, u_2) = \frac{u_1^T u_2}{\|u_1\|_2 \|u_2\|_2}$
  - The cosine similarity will be in [-1, 1], 1 means 2 users is completely similar, 0 means 2 users are irrelevant
  - The cosine similarity matrix is the symmetric matrix because cosine function is even function, $\cos(A) = \cos(-A)$ so $\cos(u_1, u_2) = \cos(u_2, u_1)$
- Sparse matrix is the good type to store the utility matrix:
  - Dimension of utility matrix is too large especially when amount of users and items increases
  - The matrix includes considerable amount of missing ratings or **0** ratings, so sparse matrix will save the memory
- Look at figure **c** we will see that
  - u_0 is more similar to u_1 and u_5 than the others
  - u_2 is more similar to u_3, u_4, u_5 and u_6 than the others

→ It's more important to cluster the user and item based on its similarity matrix to identify the types of customers or items



Hình 18.2: Ví dụ mô tả User-user Collaborative Filtering. a) Utility Matrix ban đầu. b) Utility Matrix đã được chuẩn hoá. c) User similarity matrix. d) Dự đoán các (normalized) *ratings* còn thiếu. e) Ví dụ về cách dự đoán normalized rating của u_1 cho i_1. f) Dự đoán các (denormalized) *ratings* còn thiếu.

Item-Item Collaborative Filtering

- The user-user CF has some disadvantages
  - The user similarity matrix is too large when amount of users is much more than amount of items, so storing this matrix is almost impossible
  - The utility matrix is usually sparse, so when amount of users is much more than items and there are users which don't rate any items, calculating the users sim matrix doesn't mean anything and when users who doesn't rate any items rate one items, we need to re-determine the means of users' ratings and users sim matrix which is very time-consuming
- Item-Item CF can solve these problems
  - When amount of users is much more than amount of items, the items sim matrix is much smaller so the storing this matrix is more feasible
  - Calculation of items sim matrix is more reliable than users sim matrix when amount of users is much more than amount of items

→ Item-item CF can be calculated based on User-user CF by transpose the utility matrix or treat items as users and treat users as items

- Similar to above part, we can cluster the items based on items similarity matrix
  - i_0 is more similar to i_1 and i_2 than others
  - i_3 is more similar to i_4 than the others
- The normalized utility matrix of items is quite similar to the users utility matrix except the last columns

	u_0	u_1	u_2	u_3	u_4	u_5	u_6		
i_0	5	5	2	0	1	?	?	→	2.6
i_1	4	?	?	0	?	2	?	→	2
i_2	?	4	1	?	?	1	1	→	1.75
i_3	2	2	3	4	4	?	4	→	3.17
i_4	2	0	4	?	?	?	5	→	2.75

a) Original utility matrix $\mathbf{Y}$ and mean item ratings.

	u_0	u_1	u_2	u_3	u_4	u_5	u_6
i_0	2.4	2.4	-.6	-2.6	-1.6	0	0
i_1	2	0	0	-2	0	0	0
i_2	0	2.25	-0.75	0	0	-0.75	-0.75
i_3	-1.17	-1.17	-0.17	0.83	0.83	0	0.83
i_4	-0.75	-2.75	1.25	0	0	0	2.25

b) Normalized utility matrix $\bar{\mathbf{Y}}$.

	i_0	i_1	i_2	i_3	i_4
i_0	1	0.77	0.49	-0.89	-0.52
i_1	0.77	1	0	-0.64	-0.14
i_2	0.49	0	1	-0.55	-0.88
i_3	-0.89	-0.64	-0.55	1	0.68
i_4	-0.52	-0.14	-0.88	0.68	1

c) Item similarity matrix $\mathbf{S}$.

	u_0	u_1	u_2	u_3	u_4	u_5	u_6
i_0	2.4	2.4	-.6	-2.6	-1.6	-0.29	-1.52
i_1	2	2.4	-0.6	-2	-1.25	0	-2.25
i_2	2.4	2.25	-0.75	-2.6	-1.20	-0.75	-0.75
i_3	-1.17	-1.17	-0.17	0.83	0.83	0.34	0.83
i_4	-0.75	-2.75	1.25	1.03	1.16	0.65	2.25

d) Normalized utility matrix $\bar{\mathbf{Y}}$.

Hình 18.3: Ví dụ mô tả item-item CF. a) Ma trận utility ban đầu. b) Ma trận utility đã được chuẩn hoá. c) User similarity matrix. d) Dự đoán các (normalized) *rating* còn thiếu.