Neighborhood-Based Recommendation Methods

Masoud Mansoury
AMLab, University of Amsterdam
Discovery Lab, Elsevier

Why Recommender Systems?

- **The increasing importance of the Web as a medium for electronic and business transactions**
  - Abundance of data available
    - Users can easily provide feedback with a simple click of a mouse, e.g., five-star rating system
    - Simple act of a user buying or browsing an item
  - Information overload
    - Many choices available

- **Most of todays Internet Businesses deeply root their success in the ability to provide users with strongly personalized experiences.**

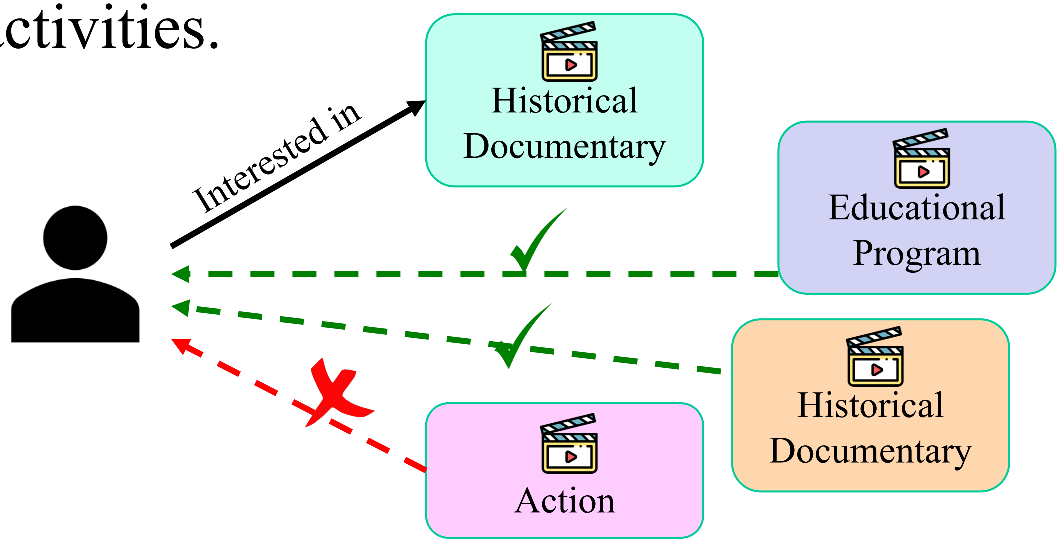
Why Recommender Systems?

- **Recommender Systems are a particular type of personalized Web-based application that provide to users personalized recommendations about the content they may be interested in.**
  - To help people discover new contents
  - To discover which things go together
  - To personalize user experiences in response to user feedback
  - To recommend incredible products that are relevant to our interests
  - To identify things that we like
  - …

To model people’s preferences, opinions, and behavior

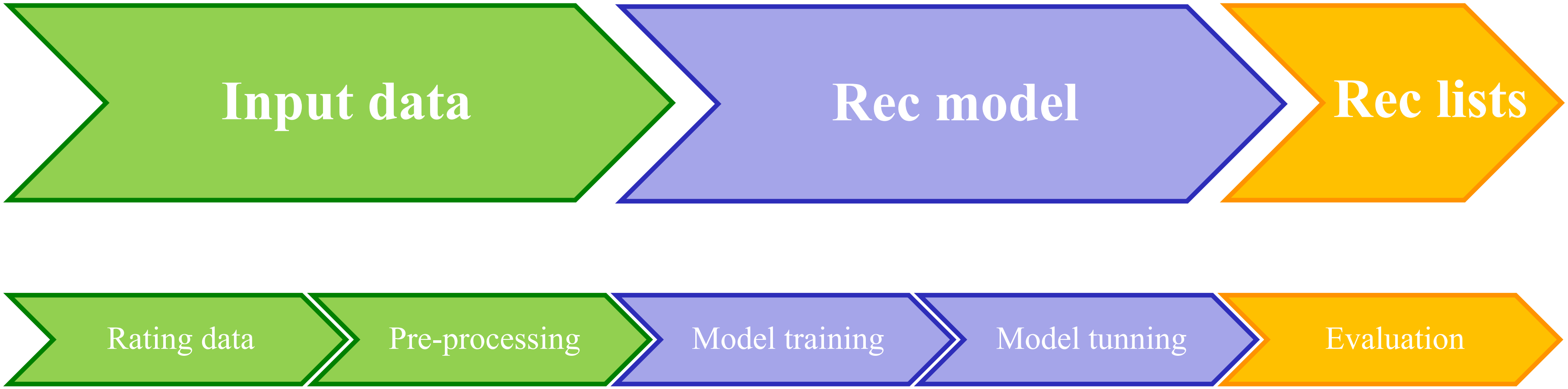
Recommendation Algorithms

- **What is the basic principle that underlies the working of recommendation algorithms?**
  - Significant dependencies exist between user- and item-centric activities.



  - Various categories of items may show significant correlations, which can be leveraged.

Recommendation Pipeline



Recommendation Approaches

- **Collaborative Filtering Models**
  - Memory-based mothods (*neighborhood-based algorithms*)
    - User-based collaborative filtering
    - Item-based collaborative filtering
  - Model-based methods

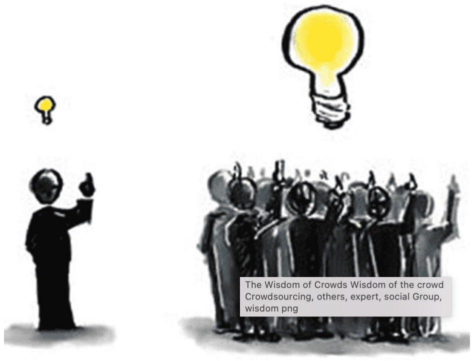
Today class

- **Content-based Recommender Systems**
- **Knowledge-based Recommender Systems**
- **Demographic Recommender Systems**
- **Hybrid and Ensemble-based Recommender Systems**

Collaborative Filtering (CF)

- **The most prominent approach to generate recommendations**
  - Used by large, commenrcial e-commerce sites
  - Well-understood, various algorithms and variations exist
  - Applicable in many domains (book, movie, …)

- **Approach**
  - Use the "wisdom of the crowd" to recommend items



- **Basic assumption and ideas**
  - Users give ratings to catalog items (implicitly or explicitly)
  - Customers who had similar tastes in the past, will have similar tastes in the future

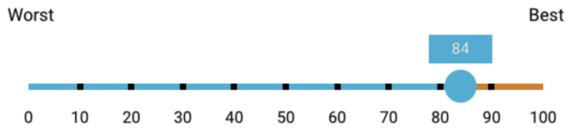
Pure CF Approach

- **Input**
  - Only a matrix of given user-item ratings

- **Output types**
  - A (numerical) prediction indicating to what degree the current user will like or dislike a certain item
  - A top-N list of recommended items

Types of Rating data

- **Continuous ratings**

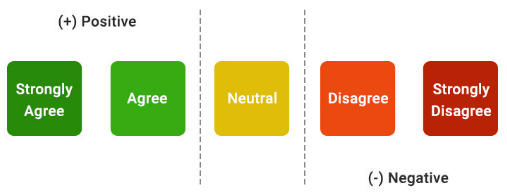


  - e.g., Jester joke recommendattion
  - Continuous rating scale, corresponding to level of like or dislike.
  - Drawback: it creates a burden on the user of having to think of a real value from an infinite number of possibilities.

- **Interval-based ratings**

  - Drawn from a 5-point or 7-point scale.
  - An important assumption is that the numerical values explicitly define the distances between the ratings, and the rating values are typically equidistant.

- **Ordinal ratings**



  - Ordered categorical values
  - e.g., responses such as “Strongly Disagree,” “Disagree,” “Neutral,” “Agree,” and “Strongly Agree.”.

Types of Rating data

- **Binary ratings**

  - Only two options are present, corresponding to positive or negative responses.
  - e.g., the Pandora Internet radio station provides users with the ability to either like or dislike a particular music track.
  - Binary ratings are an example of the case where forced choice is imposed on the user.
  - In cases where the user is neutral, she will often not specify a rating at all.

- **Unary ratings** *(implicit feedback)*

  - Allow the user to specify a positive preference for an item, but there is no mechanism to specify a negative preference.
  - e.g., “like” button on Facebook.
  - More often, derived from customer actions, e.g., the act of buying an item.
  - If the customer has not bought the item, then it does not necessarily indicate a dislike for the item.

Sample Interval-based Ratings

UserID	MovieID	Rating
u1	*i1*	7
u1	*i2*	6
u1	*i3*	7
u1	*i4*	4
u1	*i5*	5
u1	*i6*	4
u2	*i1*	6
u2	*i2*	7
u2	*i4*	4
u2	*i5*	3
u2	*i6*	4
u3	*i2*	3
u3	*i3*	3
u3	*i4*	1
u3	*i5*	1
u4	*i1*	1
u4	*i2*	2
u4	*i3*	2
u4	*i4*	3
u4	*i5*	3
u4	*i6*	4
u5	*i1*	1
u5	*i3*	1
u5	*i4*	2
u5	*i5*	3
u5	*i6*	3

MovieID	Title
i1	*The Shawshank Redemption*
i2	*The Godfather*
i3	*The Dark Knight*
i4	*Schindler's List*
i5	*Pulp Fiction*
i6	*Forest Gump*

Pre-processing

UserID	MovieID	Rating
u1	*i1*	7
u1	*i2*	6
u1	*i3*	7
u1	*i4*	4
u1	*i5*	5
u1	*i6*	4
u2	*i1*	6
u2	*i2*	7
u2	*i4*	4
u2	*i5*	3
u2	*i6*	4
u3	*i2*	3
u3	*i3*	3
u3	*i4*	1
u3	*i5*	1
u4	*i1*	1
u4	*i2*	2
u4	*i3*	2
u4	*i4*	3
u4	*i5*	3
u4	*i6*	4
u5	*i1*	1
u5	*i3*	1
u5	*i4*	2
u5	*i5*	3
u5	*i6*	3

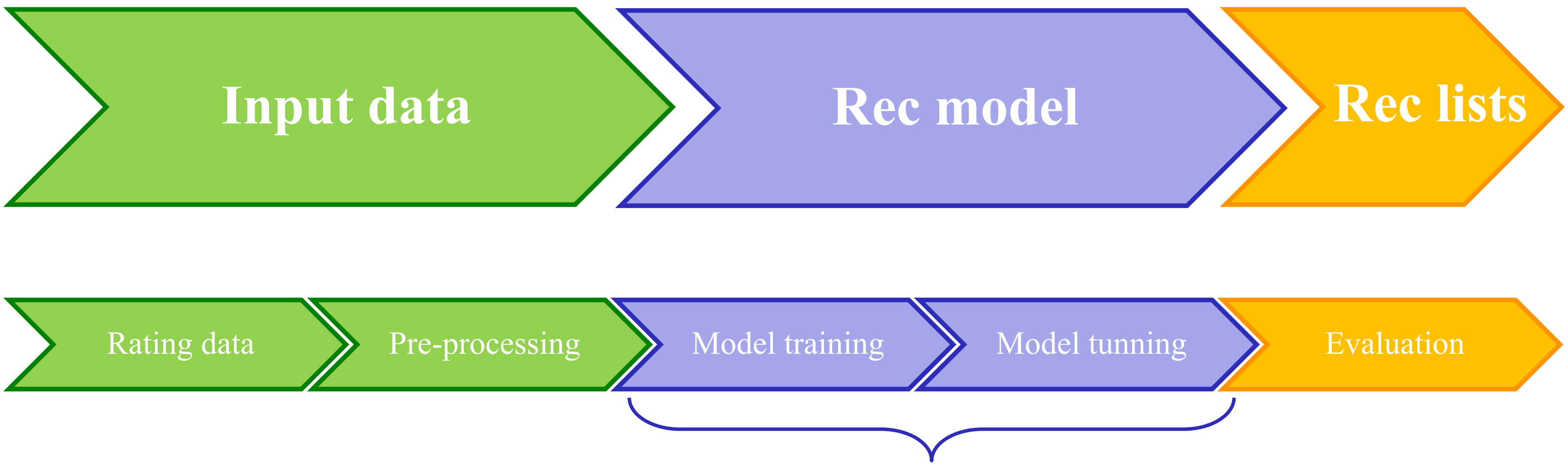
Rating Matrix

Movies

Users

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Recommendation Pipeline



Given user-item rating matrix/data,
how to build a recommendation model to predict what rating a target user might give to a target item?

Recommendation Approaches

- **Collaborative Filtering Models**
  - Memory-based mothods (*neighborhood-based algorithms*)
    - User-based collaborative filtering
    - Item-based collaborative filtering
  - Model-based methods

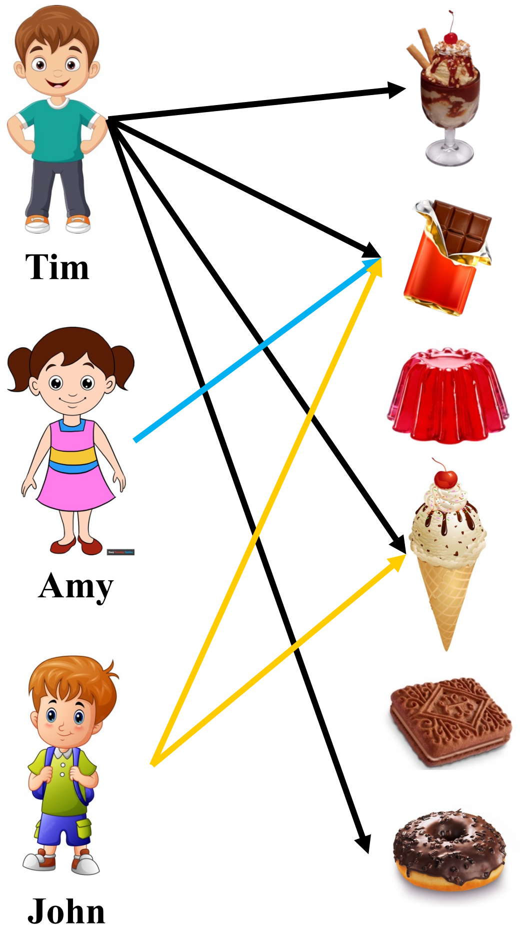
  Today class

- **Content-based Recommender Systems**
- **Knowledge-based Recommender Systems**
- **Utility-based Recommender Systems**
- **Demographic Recommender Systems**
- **Hybrid and Ensemble-based Recommender Systems**

Neighborhood-based CF

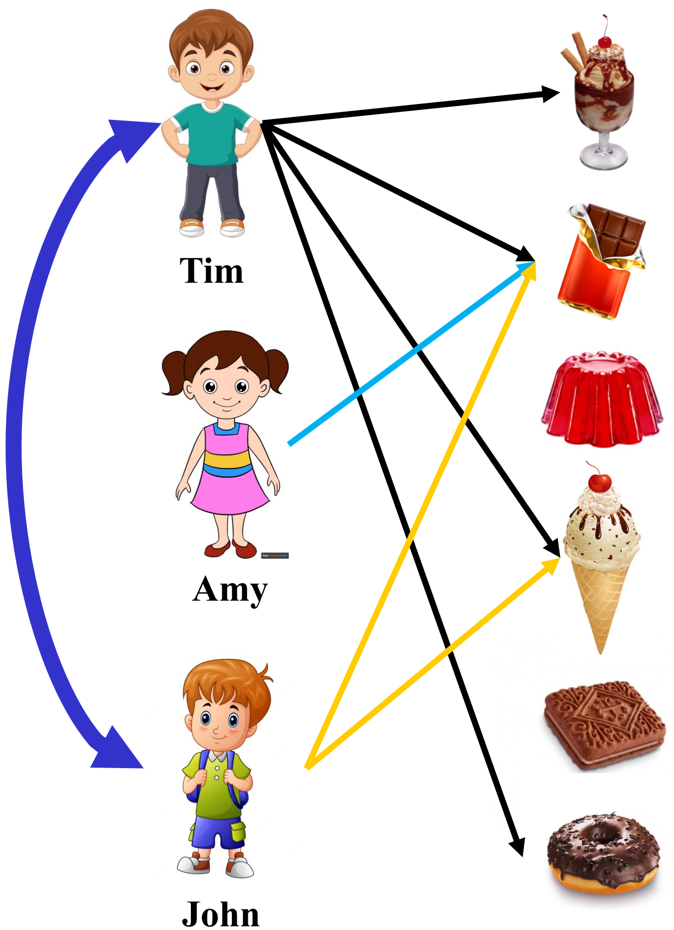
- **User-based Collaborative Filtering**
  - Ratings provided by like-minded users of a target user are used to make the recommendations.
  - The basic idea is to determine users, who are similar to the target user A, and recommend ratings for the unobserved ratings of A by computing weighted averages of the ratings of this peer group.
  - e.g., if Alice and Bob have rated movies in a similar way in the past, then Alice's observed ratings on the movie Terminator can be used to predict Bob's unobserved ratings on this movie.
- **Item-based Collaborative Filtering**
  - To make the rating predictions for target item B by user A, the first step is to determine a set S of items that are most similar to target item B.
  - The ratings in item set S, which are specified by A, are used to predict whether the user A will like item B.
  - e.g., Bob's ratings on similar science fiction movies like Alien and Predator can be used to predict his rating on Terminator.

User-Based VS. Item-Based CF



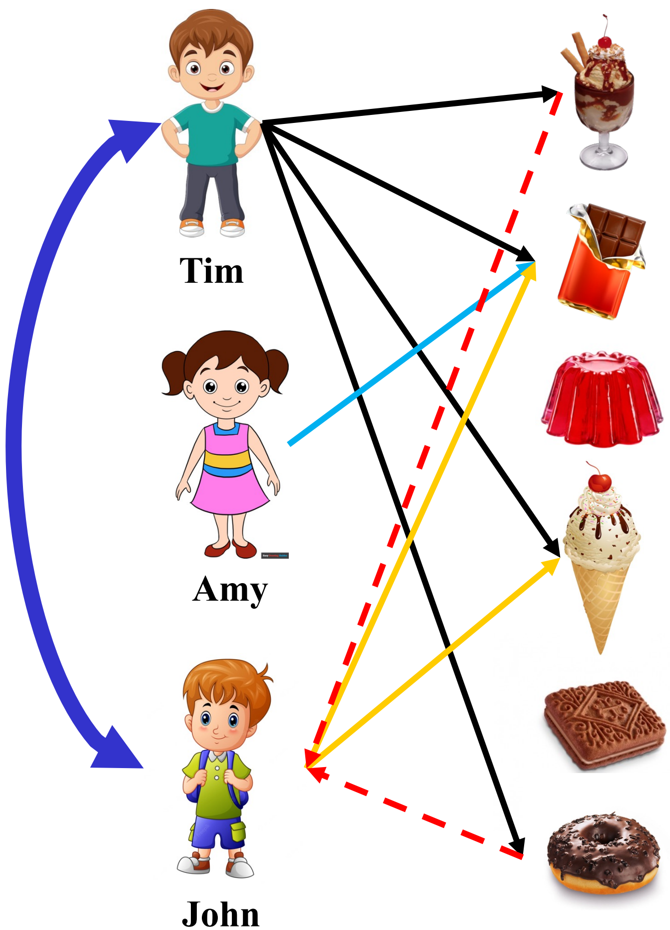
User-based Collaborative Filtering

User-Based VS. Item-Based CF



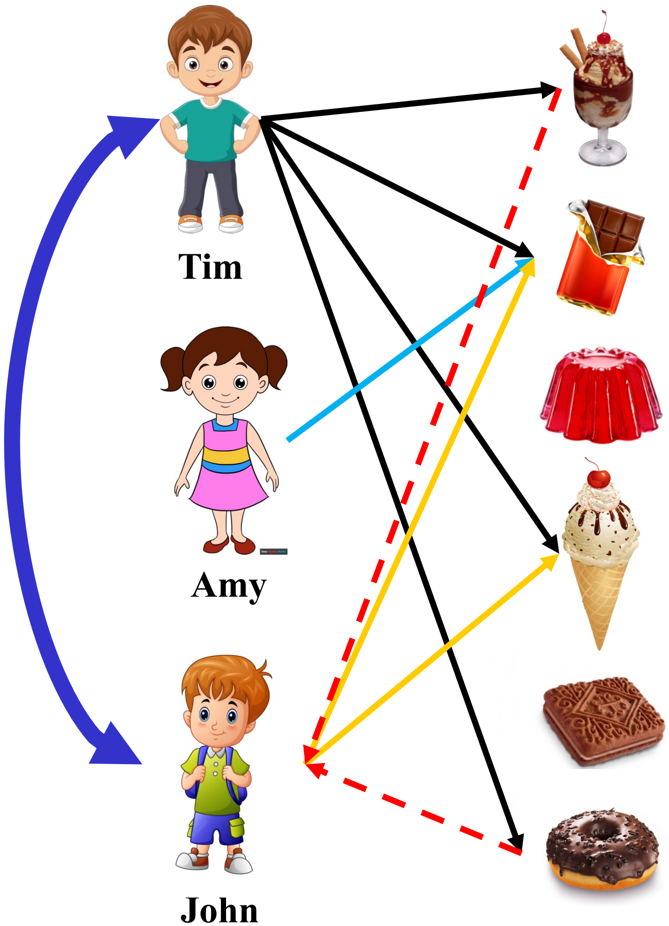
User-based Collaborative Filtering

User-Based VS. Item-Based CF

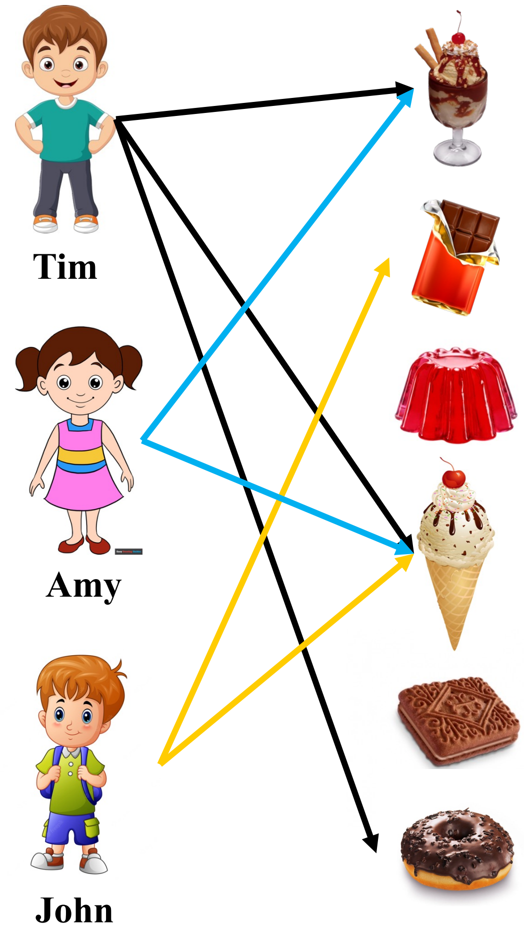


User-based Collaborative Filtering

User-Based VS. Item-Based CF

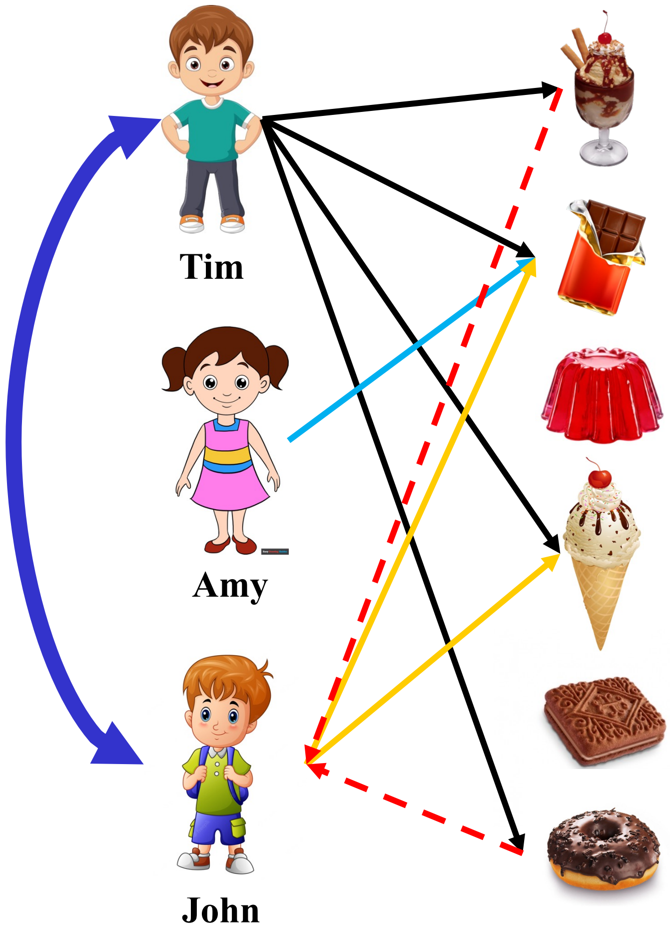


User-based Collaborative Filtering

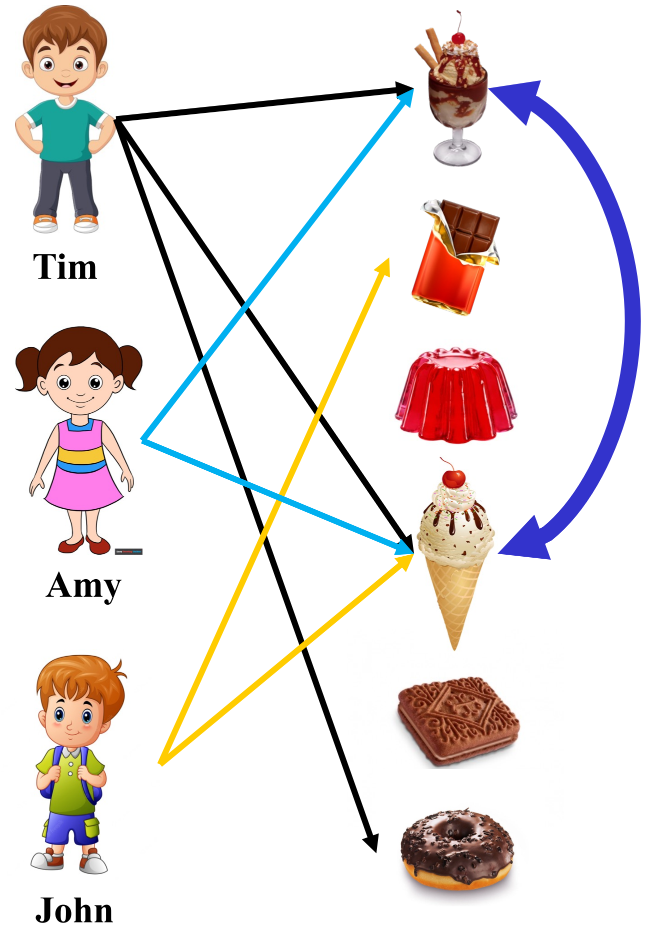


Item-based Collaborative Filtering

User-Based VS. Item-Based CF

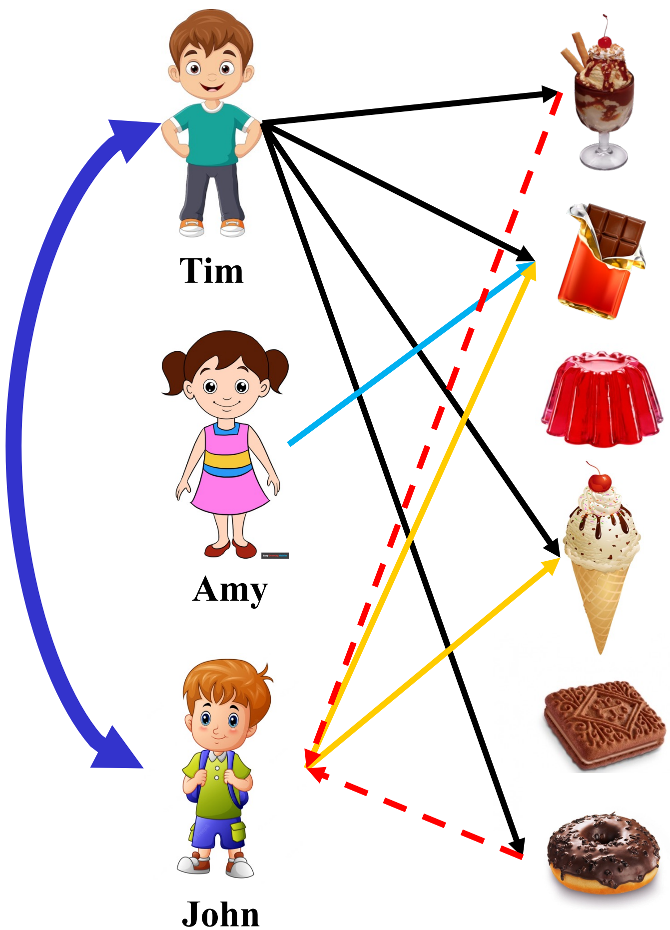


User-based Collaborative Filtering

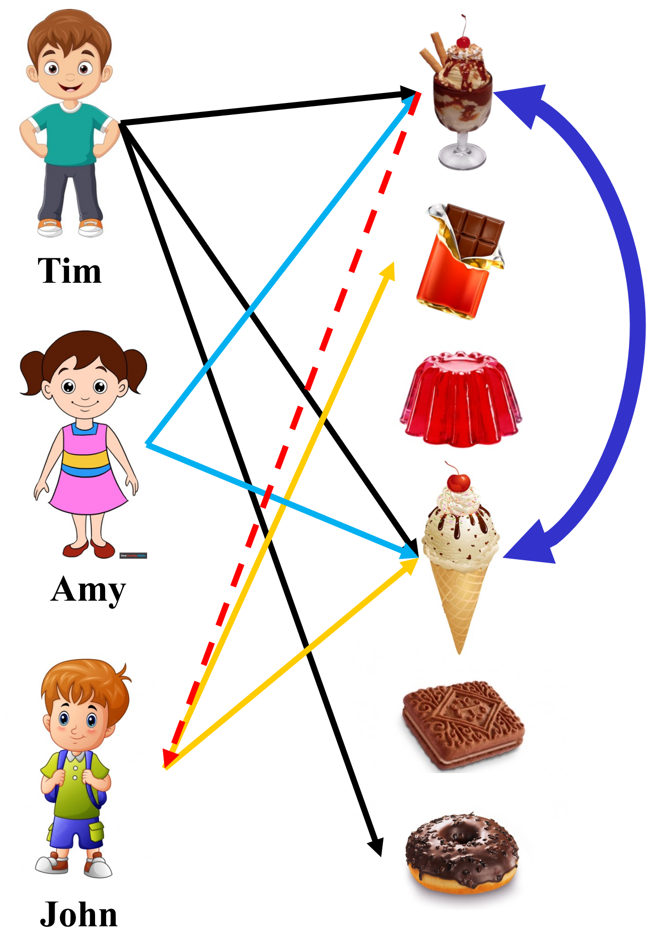


Item-based Collaborative Filtering

User-Based VS. Item-Based CF



User-based Collaborative Filtering



Item-based Collaborative Filtering

User-based Collaborative Filtering

- **K-Nearest Neighbor (KNN) approach in Machine Learning**

- **The basic technique**
  - Given a "target user" (Alice) and an item *i* not yet seen by Alice
    - Find a set of users (neighbors) who liked the same items as Alice in the past **and** who have rated item *i*
    - Use, e.g., the average of their ratings to predict, if Alice will like item *i*
    - Do this for all items Alice has not seen and recommend the best-rated.

- **Basic assumption and idea**
  - If users had similar tastes in the past they will have similar tastes in the future
  - User preferences remain stable and consistent over time

User-based Collaborative Filtering

- **Some first questions**
  - How do we measure **similarity** among users?
  - How many **neighbors** sould we consider?
  - How do we generate a **prediction** from the neighbors' ratings?

Notations

- r_{uk}: **rating given by user** u **to item** k.
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Notations

- $\boldsymbol{r_{uk}}$: **rating given by user $\boldsymbol{u}$ to item $\boldsymbol{k}$.**
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$
- $\boldsymbol{I_u}$: **set of items rated by user $\boldsymbol{u}$.**
  - $I_{u3} = \{i2, i3, i4, i5\}$
  - $I_{u5} = \{i1, i3, i4, i5, i6\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Notations

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

- **r_{uk}: rating given by user u to item k.**
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$
- **I_u: set of items rated by user u.**
  - $I_{u3} = \{i2, i3, i4, i5\}$
  - $I_{u5} = \{i1, i3, i4, i5, i6\}$
- **$I_u \cap I_v$: set of items rated by both users u and v.**
  - $I_{u3} \cap I_{u5} = \{i3, i4, i5\}$

Notations

- r_{uk}: rating given by user u to item k.
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$
- I_u: set of items rated by user u.
  - $I_{u3} = \{i2, i3, i4, i5\}$
  - $I_{u5} = \{i1, i3, i4, i5, i6\}$
- $I_u \cap I_v$: set of items rated by both users u and v.
  - $I_{u3} \cap I_{u5} = \{i3, i4, i5\}$
- μ_u: mean rating of ratings provided by user u.
  - $\mu_u = \frac{\sum_{k \in I_u} r_{uk}}{|I_u|}$
  - $\mu_{u2} = \frac{\sum_{k \in I_{u2}} r_{u2k}}{|I_{u2}|} = \frac{r_{u2i1}+r_{u2i2}+r_{u2i4}+r_{u2i5}+r_{u2i6}}{5} = \frac{6+7+4+3+4}{5} = 4.8$
  - $\mu_{u4} = \frac{\sum_{k \in I_{u4}} r_{u4k}}{|I_{u4}|} = \frac{r_{u4i1}+r_{u4i2}+r_{u4i3}+r_{u4i4}+r_{u4i5}+r_{u4i6}}{6} = \frac{1+2+2+3+3+4}{6} = 2.5$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Step 1: Similarity computation

- **We compute the similarity values between the *target user* and all the other users.**

- **Pearson coefficient is a well-known similarity metric which is used to compute how much two users are similar.**

- **The similarity value between users $\boldsymbol{u}$ and $\boldsymbol{v}$:**

$$\boldsymbol{Sim(u,v) = Pearson(u,v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u).(r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} . \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}}$$

Step 1: Similarity computation

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u).(r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} \, . \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}$$

What is $Pearson(u1, u3)$?

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Step 1: Similarity computation

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u).(r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} \cdot \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}$$

What is $Pearson(u1, u3)$?

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

	μ
u1	5.5
u3	2

Step 1: Similarity computation

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{k\in I_u \cap I_v}(r_{uk}-\mu_u).(r_{vk}-\mu_v)}{\sqrt{\sum_{k\in I_u \cap I_v}(r_{uk}-\mu_u)^2}\cdot\sqrt{\sum_{k\in I_u \cap I_v}(r_{vk}-\mu_v)^2}}$$

What is $Pearson(u1, u3)$?

$I_{u1} \cap I_{u3} = \{i2, i3, i4, i5\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

	μ
u1	5.5
u3	2

Step 1: Similarity computation

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u).(r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} . \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}$$

What is $Pearson(u1, u3)$?

$I_{u1} \cap I_{u3} = \{i2, i3, i4, i5\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

	μ
u1	**5.5**
u3	**2**

$r_{ui} - \mu_u$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	-	0.5	1.5	-1.5	-0.5	-
u3	?	1	1	-1	-1	?

Step 1: Similarity computation

$$\boldsymbol{Sim(u,v) = Pearson(u,v) = \frac{\sum_{k\in I_u \cap I_v}(r_{uk}-\mu_u).(r_{vk}-\mu_v)}{\sqrt{\sum_{k\in I_u \cap I_v}(r_{uk}-\mu_u)^2}.\sqrt{\sum_{k\in I_u \cap I_v}(r_{vk}-\mu_v)^2}}}$$

What is $\boldsymbol{Pearson(u1,u3)}$?

$I_{u1} \cap I_{u3} = \{i2, i3, i4, i5\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

$\boldsymbol{\mu}$

u1	**5.5**
u3	**2**

$r_{ui} - \mu_u$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	-	0.5	1.5	-1.5	-0.5	-
u3	?	1	1	-1	-1	?

$$\boldsymbol{Pearson(u1,u3)} = \frac{(r_{u1i2}-\mu_{u1})\times(r_{u3i2}-\mu_{u3}) + (r_{u1i3}-\mu_{u1})\times(r_{u3i3}-\mu_{u3}) + (r_{u1i4}-\mu_{u1})\times(r_{u3i4}-\mu_{u3}) + (r_{u1i5}-\mu_{u1})\times(r_{u3i5}-\mu_{u3})}{\sqrt{(r_{u1i2}-\mu_{u1})^2 + (r_{u1i3}-\mu_{u1})^2 + (r_{u1i4}-\mu_{u1})^2 + (r_{u1i5}-\mu_{u1})^2}\times\sqrt{(r_{u3i2}-\mu_{u3})^2 + (r_{u3i3}-\mu_{u3})^2 + (r_{u3i4}-\mu_{u3})^2 + (r_{u3i5}-\mu_{u3})^2}}$$

Step 1: Similarity computation

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u).(r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} . \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}$$

What is $Pearson(u1, u3)$?

$I_{u1} \cap I_{u3} = \{i2, i3, i4, i5\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

μ

u1	5.5
u3	2

$r_{ui} - \mu_u$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	-	0.5	1.5	-1.5	-0.5	-
u3	?	1	1	-1	-1	?

$$Pearson(u1,u3) = \frac{(r_{u1i2} - \mu_{u1}) \times (r_{u3i2} - \mu_{u3}) + (r_{u1i3} - \mu_{u1}) \times (r_{u3i3} - \mu_{u3}) + (r_{u1i4} - \mu_{u1}) \times (r_{u3i4} - \mu_{u3}) + (r_{u1i5} - \mu_{u1}) \times (r_{u3i5} - \mu_{u3})}{\sqrt{(r_{u1i2} - \mu_{u1})^2 + (r_{u1i3} - \mu_{u1})^2 + (r_{u1i4} - \mu_{u1})^2 + (r_{u1i5} - \mu_{u1})^2} \times \sqrt{(r_{u3i2} - \mu_{u3})^2 + (r_{u3i3} - \mu_{u3})^2 + (r_{u3i4} - \mu_{u3})^2 + (r_{u3i5} - \mu_{u3})^2}}$$

$$Pearson(u1,u3) = \frac{(0.5 \times 1) + (1.5 \times 1) + (-1.5 \times -1) + (-0.5 \times -1)}{\sqrt{0.5^2 + 1.5^2 + (-1.5)^2 + (-0.5)^2} \times \sqrt{1^2 + 1^2 + (-1)^2 + (-1)^2}} = 0.944$$

Step 1: Similarity computation

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Simlarity values between $u3$ and all other users

$\boldsymbol{Pearson(u3, v)}$
0.944
0.939
1.0
-1.0
-0.817

Step 2: Neighbor Selection

- **Which neighbors to be considered for rating prediction?**
- **One way is to consider the set of *k* users with the highest Pearson coefficient with the target user.**
- **The value for *k* is often experimentally determined.**
- **The closest *k* neighbor users to the target user are separately found for each predicted item, such that each of these *k* users have specified ratings for that item.**
- **The weighted average of these ratings can be returned as the predicted rating for that item.**

Step 3: Rating Prediction

- **Predicting the rating that a target user u might give to a target item j:**

The set of k closest users to target user u, who have specified rating for item j.

The similarity degree of user v and target user u

Rating by user v on target item j

Average rating provided by user v

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u, v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u, v)|}$$

Step 3: Rating Prediction

- **Predicting the rating that a target user u might give to a target item j:**

The set of k closest users to target user u, who have specified rating for item j.

Rating by user v on target item j

Average rating provided by user v

$$\hat{r}_{uj} = \boldsymbol{\mu_u} + \frac{\sum_{v \in P_u(j)} \overbrace{Sim(u,v)}^{\text{The similarity degree of user } v \text{ and target user } u} \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

- **Average rating of user u on different items**

Step 3: Rating Prediction

- **Predicting the rating that a target user u might give to a target item j:**

The set of k closest users to target user u, who have specified rating for item j.

The similarity degree of user v and target user u

Rating by user v on target item j

Average rating provided by user v

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

- **the weighted average of the mean-centered rating of item j in top-k neighbors of target user u.**

Step 3: Rating Prediction

- **Predicting the rating that a target user u might give to a target item j:**

The set of k closest users to target user u, who have specified rating for item j.

The similarity degree of user v and target user u

Rating by user v on target item j

Average rating provided by user v

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (\boldsymbol{r_{vj} - \mu_v})}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

- **The reason is that different users may provide ratings on different scales.**
  - One user might rate all items highly, whereas another user might rate all items negatively.
  - User A tends to rate high: [3,4,4,4,5,5,5,5,5], the average would be 4.44
  - User B tends to rate low: [1,1,1,2,2,2,2,3,3], the average would be 1.89

Example

- **What is the predicted rating for user $u3$ on item $i1$?**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.944
0.939
1.0
-1.0
-0.817

Example

- What is the predicted rating for user $u3$ on item $i1$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.944
0.939
1.0
-1.0
-0.817

- **Users $u1$ and $u2$ are the most similar users to $u3$**
  - $P_{u3}(i1) = \{u1, u2\}$

Example

- **What is the predicted rating for user $u3$ on item $i1$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.944
0.939
1.0
-1.0
-0.817

- **Users $u1$ and $u2$ are the most similar users to $u3$**
  - $P_{u3}(i1) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i1} = 2 + \frac{0.944}{}$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

Example

- **What is the predicted rating for user $u3$ on item $i1$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.894
0.939
1.0
-1.0
-0.817

- **Users $u1$ and $u2$ are the most similar users to $u3$**
  - $P_{u3}(i1) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i1} = 2 + \frac{0.944 \times (7 - 5.5)}{}$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

Example

- What is the predicted rating for user $u3$ on item $i1$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	i2	i3	i4	i5	i6
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.894
0.939
1.0
-1.0
-0.817

- Users $u1$ and $u2$ are the most similar users to $u3$
  - $P_{u3}(i1) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i1} = 2 + \frac{0.944 \times (7 - 5.5) + 0.939}{}$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

Example

- What is the predicted rating for user $u3$ on item $i1$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	i2	i3	i4	i5	i6
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.894
0.939
1.0
-1.0
-0.817

- Users $u1$ and $u2$ are the most similar users to $u3$
  - $P_{u3}(i1) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i1} = 2 + \frac{0.944 \times (7 - 5.5) + 0.939 \times (6 - 4.8)}{}$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

Example

- **What is the predicted rating for user $u3$ on item $i1$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to u3
0.894
0.939
1.0
-1.0
-0.817

- **Users $u1$ and $u2$ are the most similar users to $u3$**
  - $P_{u3}(i1) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i1} = 2 + \frac{0.944 \times (7 - 5.5) + 0.939 \times (6 - 4.8)}{0.944 + 0.939} \approx 3.35$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u,v) \times (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u,v)|}$$

Example

- What is the predicted rating for user $u3$ on item $i6$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

$Similarity\ to\ u3$
0.944
0.939
1.0
-1.0
-0.817

- Users $u1$ and $u2$ are the most similar users to $u3$
  - $P_{u3}(i6) = \{u1, u2\}$
- $\mu_{u3} = 2$

$$\hat{r}_{u3i6} = 2 + \frac{0.944\times(4-5.5) + 0.939\times(4-4.8)}{0.944 + 0.939} \approx 0.86$$

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v\in P_u(j)} Sim(u,v)\times(r_{vj}-\mu_v)}{\sum_{v\in P_u(j)} |Sim(u,v)|}$$

Example

- **Given**
  - $\hat{r}_{u3i1} = 3.35$
  - $\hat{r}_{u3i6} = 0.86$

- **Item $i1$ should be prioritized over item $i6$ as a recommendation to user $u3$.**

Item-based Collaborative Filtering

- **Neighbors are constructed in terms of *items* rather than *users*.**

- **Similarities need to be computed between items (or columns in the ratings matrix).**

- **The basic idea is to leverage the user's own ratings on similar items in the final step of making the prediction.**

Notations

- r_{uk}: **rating given by user u to item k.**
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$

	$i1$	$i2$	$i3$	$i4$	$i5$	$i6$
$u1$	7	6	7	4	5	4
$u2$	6	7	?	4	3	4
$u3$	?	3	3	1	1	?
$u4$	1	2	2	3	3	4
$u5$	1	?	1	2	3	3

Notations

- r_{uk}: **rating given by user** u **to item** k**.**
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$
- U_i: **set of users who rated item** i**.**
  - $U_{i2} = \{u1, u2, u3, u4\}$
  - $U_{i6} = \{u1, u2, u4, u5\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Notations

- r_{uk}: **rating given by user u to item k.**
  - $r_{u2i4} = 4$
  - $r_{u5i3} = 1$
- U_i: **set of users who rated item i.**
  - $U_{i2} = \{u1, u2, u3, u4\}$
  - $U_{i6} = \{u1, u2, u4, u5\}$
- $U_i \cap U_j$: **set of users who rated both items i and j.**
  - $U_{i2} \cap U_{i6} = \{u1, u2, u4\}$

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Step 1: Similarity computation

- **We compute the similarity values between the *target item* and all the other items.**

- **Cosine Similarity is another well-known similarity metric which is used to compute how much two items are similar.**

- **The similarity value between items $\boldsymbol{i}$ and $\boldsymbol{j}$:**

$$\boldsymbol{Sim(i,j) = Cosine(i,j) = \frac{\sum_{u \in U_i \cap U_j} r_{ui} \times r_{uj}}{\sqrt{\sum_{u \in U_i \cap U_j} r_{ui}^2} \times \sqrt{\sum_{u \in U_i \cap U_j} r_{uj}^2}}}$$

Step 1: Similarity computation

$$Sim(i,j) = Cosine(i,j) = \frac{\sum_{u \in U_i \cap U_j} r_{ui} \times r_{uj}}{\sqrt{\sum_{u \in U_i \cap U_j} r_{ui}^2} \times \sqrt{\sum_{u \in U_i \cap U_j} r_{uj}^2}}$$

What is $Cosine(i2, i6)$?

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

$$U_{i2} \cap U_{i6} = \{u1, u2, u4\}$$

$$Cosine(i2, i6) = \frac{\overbrace{(6\times4)}^{u1} + \overbrace{(7\times4)}^{u2} + \overbrace{(2\times4)}^{u4}}{\sqrt{6^2+7^2+2^2} \times \sqrt{4^2+4^2+4^2}} = \frac{60}{\sqrt{89}\times\sqrt{48}} = 0.917$$

Step 1: Similarity computation

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Simlarity values between $i2$ and all other items

$Cosine(i2, j)$
0.983
1
0.998
0.951
0.914
0.917

Step 2: Neighbor Selection

- **This step is the same as User-based Collaborative Filtering.**

- **The set of *k* items with the highest Cosine similarity with the target item are selected as the neighbors.**

- **The value for *k* is often experimentally determined.**

Step 3: Rating Prediction

- **Predicting the rating that a target user u might give to a target item i:**

The set of k closest items to target item i, which have received rating from user u.

The similarity degree of item j and target item i

Rating by user u on item j

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} \overbrace{Sim(i,j)} \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Example

- **What is the predicted rating for user $u5$ on item $i2$?**

	$i1$	$i2$	$i3$	$i4$	$i5$	$i6$
$u1$	7	6	7	4	5	4
$u2$	6	7	?	4	3	4
$u3$	?	3	3	1	1	?
$u4$	1	2	2	3	3	4
$u5$	1	?	1	2	3	3

$Similarity\ to\ i2$
0.983
1
0.998
0.951
0.914
0.917

Example

- What is the predicted rating for user $u5$ on item $i2$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

$Similarity\ to\ i2$
0.983
1
0.998
0.951
0.914
0.917

- Items $i1$ and $i3$ are the most similar items to $i2$
  - $Q_{i2}(u5) = \{i1, i3\}$

Example

- **What is the predicted rating for user $u5$ on item $i2$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to i2
0.983
1
0.998
0.951
0.914
0.917

- **Items $i1$ and $i3$ are the most similar items to $i2$**
  - $Q_{i2}(u5) = \{i1, i3\}$

$$\hat{r}_{u5i2} = \frac{0.983}{} = 1$$

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} Sim(i,j) \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Example

- **What is the predicted rating for user $u5$ on item $i2$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to i2
0.983
1
0.998
0.951
0.914
0.917

- **Items $i1$ and $i3$ are the most similar items to $i2$**
  - $Q_{i2}(u5) = \{i1, i3\}$

$$\hat{r}_{u5i2} = \frac{0.983 \times 1}{} = 1$$

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} Sim(i,j) \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Example

- What is the predicted rating for user $u5$ on item $i2$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to i2
0.983
1
0.998
0.951
0.914
0.917

- Items $i1$ and $i3$ are the most similar items to $i2$
  - $Q_{i2}(u5) = \{i1, i3\}$

$$\hat{r}_{u5i2} = \frac{0.983 \times 1 + 0.998}{} = 1$$

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} Sim(i,j) \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Example

- What is the predicted rating for user $u5$ on item $i2$?
- Assume $k = 2$ (only consider 2 neighbors)

	i1	i2	i3	i4	i5	i6
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to i2
0.983
1
0.998
0.951
0.914
0.917

- Items $i1$ and $i3$ are the most similar items to $i2$
  - $Q_{i2}(u5) = \{i1, i3\}$

$$\hat{r}_{u5i2} = \frac{0.983 \times 1 + 0.998 \times 1}{} = 1$$

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} Sim(i,j) \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Example

- **What is the predicted rating for user $u5$ on item $i2$?**
- **Assume $k = 2$ (only consider 2 neighbors)**

	i1	*i2*	*i3*	*i4*	*i5*	*i6*
u1	7	6	7	4	5	4
u2	6	7	?	4	3	4
u3	?	3	3	1	1	?
u4	1	2	2	3	3	4
u5	1	?	1	2	3	3

Similarity to i2
0.983
1
0.998
0.951
0.914
0.917

- **Items $i1$ and $i3$ are the most similar items to $i2$**
  - $Q_{i2}(u5) = \{i1, i3\}$

$$\hat{r}_{u5i2} = \frac{0.983 \times 1 + 0.998 \times 1}{0.983 + 0.998} = 1$$

$$\hat{r}_{ui} = \frac{\sum_{j \in Q_i(u)} Sim(i,j) \times r_{uj}}{\sum_{j \in Q_i(u)} |Sim(i,j)|}$$

Comparing User-based and Item-based

- **Item-based method often provide more relevant recommendations**

Comparing User-based and Item-based

- **Item-based method often provide more relevant recommendations**
  - because of the fact that a user’s own ratings are used to perform the recommendation.

Comparing User-based and Item-based

- **Item-based method often provide more relevant recommendations**
  - because of the fact that a user's own ratings are used to perform the recommendation.
  - For example, similar items to a target *historical movie* might be a set of other historical movies.

Comparing User-based and Item-based

- **Item-based method often provide more relevant recommendations**
  - because of the fact that a user's own ratings are used to perform the recommendation.
  - For example, similar items to a target *historical movie* might be a set of other historical movies.
  - This is not the case for user-based methods in which the ratings are extrapolated from other users, who might have overlapping but different interests.

Comparing User-based and Item-based

- **Item-based method often provide more relevant recommendations**
  - because of the fact that a user's own ratings are used to perform the recommendation.
  - For example, similar items to a target *historical movie* might be a set of other historical movies.
  - This is not the case for user-based methods in which the ratings are extrapolated from other users, who might have overlapping but different interests.
  - **As a result, item-based methods often exhibit better accuracy**

Comparing User-based and Item-based

- **User-based method often provide more diverse recommendations**

Comparing User-based and Item-based

- **User-based method often provide more diverse recommendations**
  - Diversity refers to the fact that the items in the ranked list tend to be somewhat different.

Comparing User-based and Item-based

- **User-based method often provide more diverse recommendations**
  - Diversity refers to the fact that the items in the ranked list tend to be somewhat different.
  - If the items are not diverse, then if the user does not like the first item, she might not also like any of the other items in the list.

Comparing User-based and Item-based

- **User-based method often provide more diverse recommendations**
  - Diversity refers to the fact that the items in the ranked list tend to be somewhat different.
  - If the items are not diverse, then if the user does not like the first item, she might not also like any of the other items in the list.
  - Item-based methods might sometimes recommend obvious items, or items which are not novel from previous user experiences.

Comparing User-based and Item-based

- **User-based method often provide more diverse recommendations**
  - Diversity refers to the fact that the items in the ranked list tend to be somewhat different.
  - If the items are not diverse, then if the user does not like the first item, she might not also like any of the other items in the list.
  - Item-based methods might sometimes recommend obvious items, or items which are not novel from previous user experiences.
  - Without sufficient novelty and diversity, users might become bored with very similar recommendations to what they have already watched.

Comparing User-based and Item-based

- **Item-based methods can also provide a concrete reason for the recommendation.**

Comparing User-based and Item-based

- **Item-based methods can also provide a concrete reason for the recommendation.**
  - For example, Netflix often provides recommendations with statements such as
    - *Because you watched "Secrets of the Wings," [the recommendations are] <List>.*

Comparing User-based and Item-based

- **Item-based methods can also provide a concrete reason for the recommendation.**
  - For example, Netflix often provides recommendations with statements such as
    - *Because you watched "Secrets of the Wings," [the recommendations are] <List>.*
  - Such explanations can be concretely addressed with item-based method by using the item neighbourhoods.

Comparing User-based and Item-based

- **Item-based methods can also provide a concrete reason for the recommendation.**
  - For example, Netflix often provides recommendations with statements such as
    - *Because you watched "Secrets of the Wings," [the recommendations are] <List>.*
  - Such explanations can be concretely addressed with item-based method by using the item neighbourhoods.
  - On the other hand, these explanations are harder to address with user-based method, because the peer group is simply a set of anonymous users and not directly usable in the recommendation process.

Weaknesses of Neighborhood-Based Methods

- **The offline phase can sometimes be impractical in large-scale settings.**
  - Nevertheless, the online phase of neighborhood methods is always efficient.

Weaknesses of Neighborhood-Based Methods

- **The offline phase can sometimes be impractical in large-scale settings.**
  - Nevertheless, the online phase of neighborhood methods is always efficient.

- **The rating prediction is sometimes impossible due to *sparsity* of rating matrix.**
  - For example, if none of John's nearest neighbors have rated Terminator, it is not possible to provide a rating prediction of Terminator for John.

Weaknesses of Neighborhood-Based Methods

- **The offline phase can sometimes be impractical in large-scale settings.**
  - Nevertheless, the online phase of neighborhood methods is always efficient.
- **The rating prediction is sometimes impossible due to *sparsity* of rating matrix.**
  - For example, if none of John's nearest neighbors have rated Terminator, it is not possible to provide a rating prediction of Terminator for John.
- **Sparsity also creates challenges for robust similarity computation when the number of mutually rated items between two users is small.**

Strengths of Neighborhood-Based Methods

- **Easy to implement and debug.**

Strengths of Neighborhood-Based Methods

- **Easy to implement and debug.**

- **Easy to justify why a specific item is recommended**
  - The interpretability of item-based methods is particularly notable.

Strengths of Neighborhood-Based Methods

- **Easy to implement and debug.**
- **Easy to justify why a specific item is recommended**
  - The interpretability of item-based methods is particularly notable.
- **The recommendations are relatively stable with the addition of new items and users.**

Strengths of Neighborhood-Based Methods

- **Easy to implement and debug.**
- **Easy to justify why a specific item is recommended**
  - The interpretability of item-based methods is particularly notable.
- **The recommendations are relatively stable with the addition of new items and users.**
- **It is also possible to create incremental approximations of these methods.**

Neighborhood-Based Recommendation Methods

Masoud Mansoury
AMLab, University of Amsterdam
Discovery Lab, Elsevier