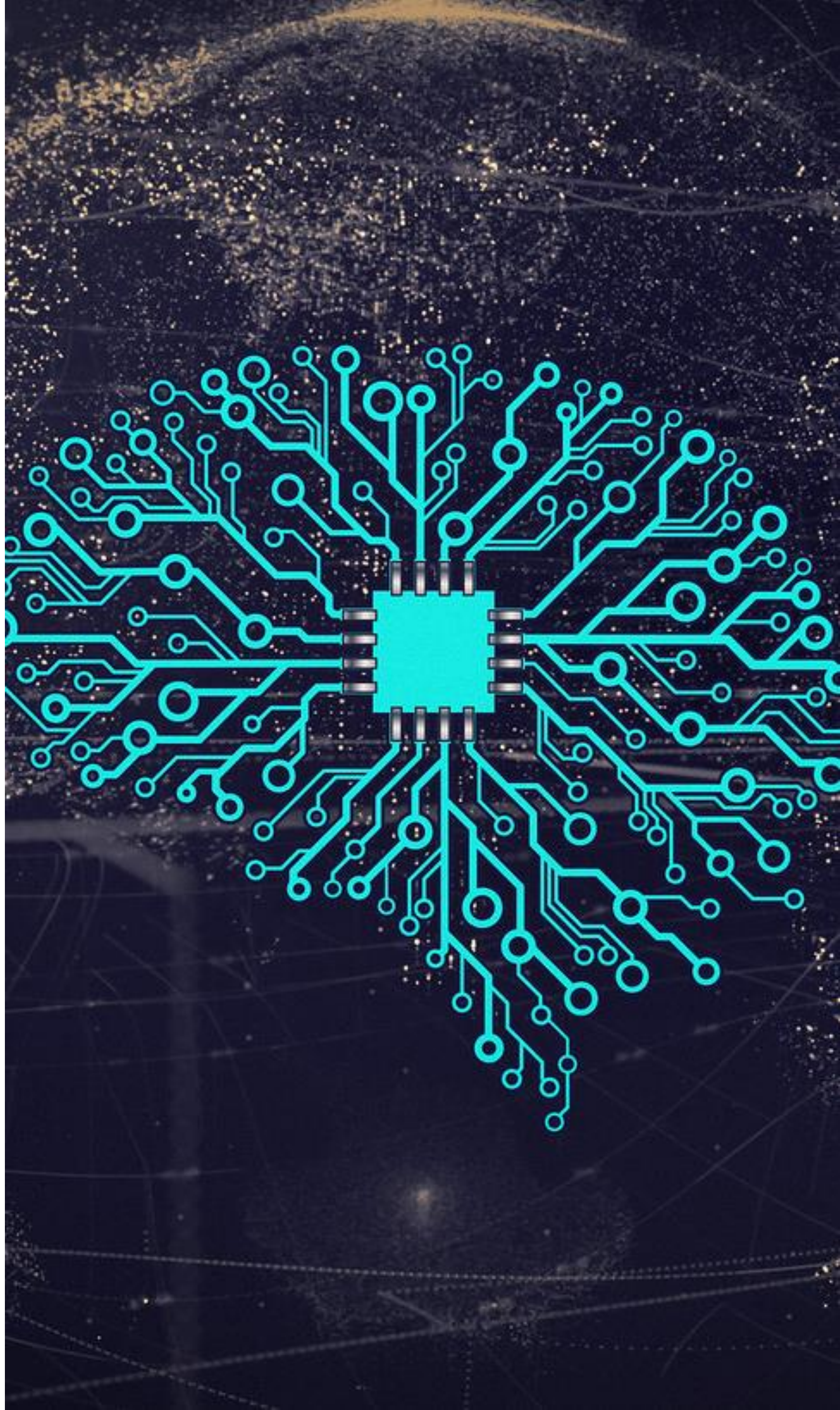
User-user collaborative filtering

Machine Learning with Big Data

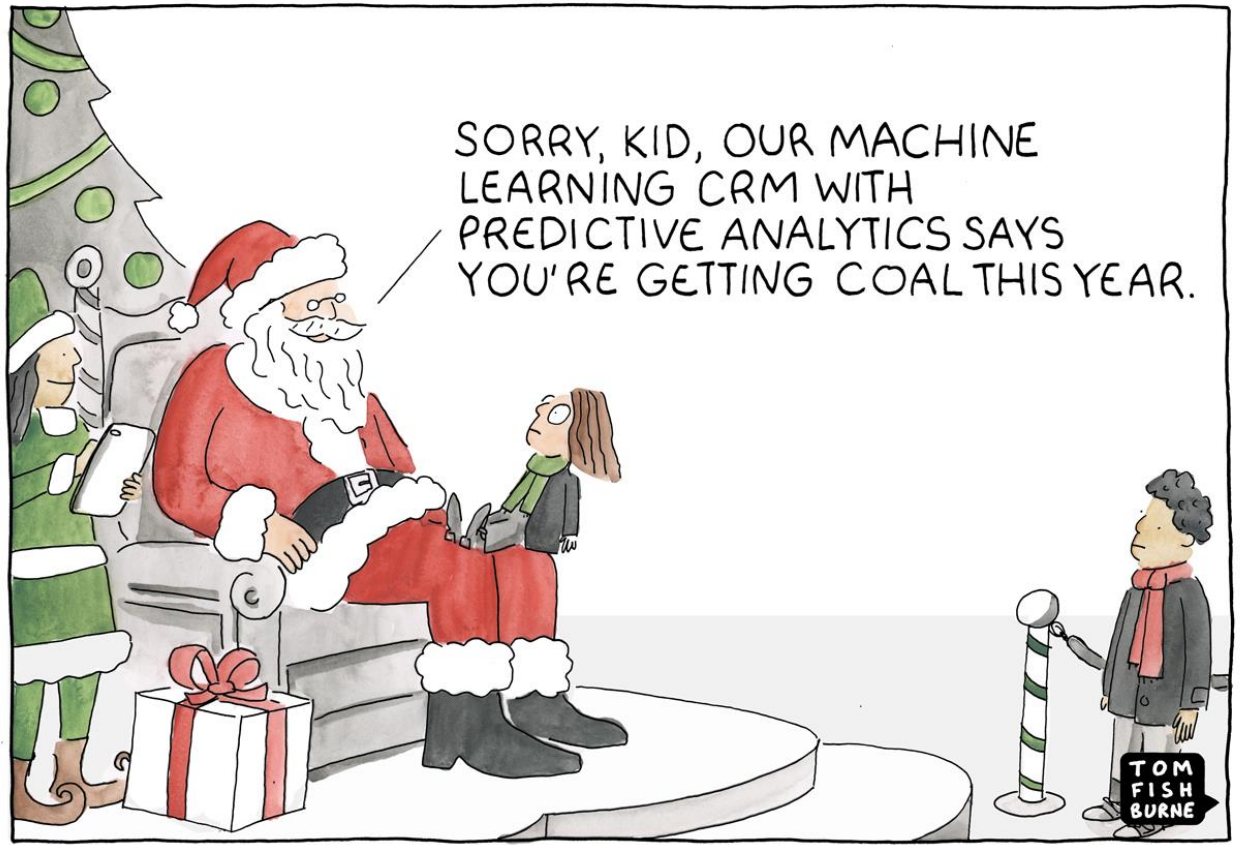
Evert Duipmans
Jeroen Linssen
Etto Salomons

SAXION UNIVERSITY OF APPLIED SCIENCES



Contents

- Recommender systems recap
- Content-based recommendations questions
- Collaborative filtering
- Similarity metrics
- User-user collaborative filtering


SORRY, KID, OUR MACHINE LEARNING CRM WITH PREDICTIVE ANALYTICS SAYS YOU'RE GETTING COAL THIS YEAR.
TOM FISH BURNE

SAXION UNIVERSITY OF APPLIED SCIENCES

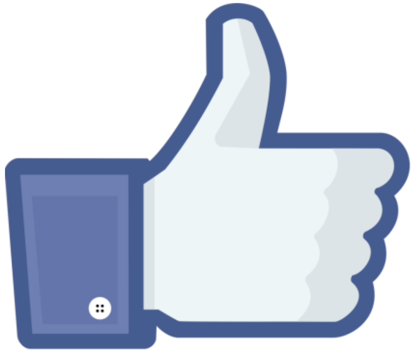
Recommender systems
What data is used?

Explicit ratings

- Rate content (stars, like/dislike, …)
- Requires extra work for the user
- Cultural differences
- People rate different
- Data is often sparse

Implicit ratings

- Things you do: click on links, read article, add to cart, buy things, how long did you watch a video?
- Lots of companies use sales data
- Things you consume



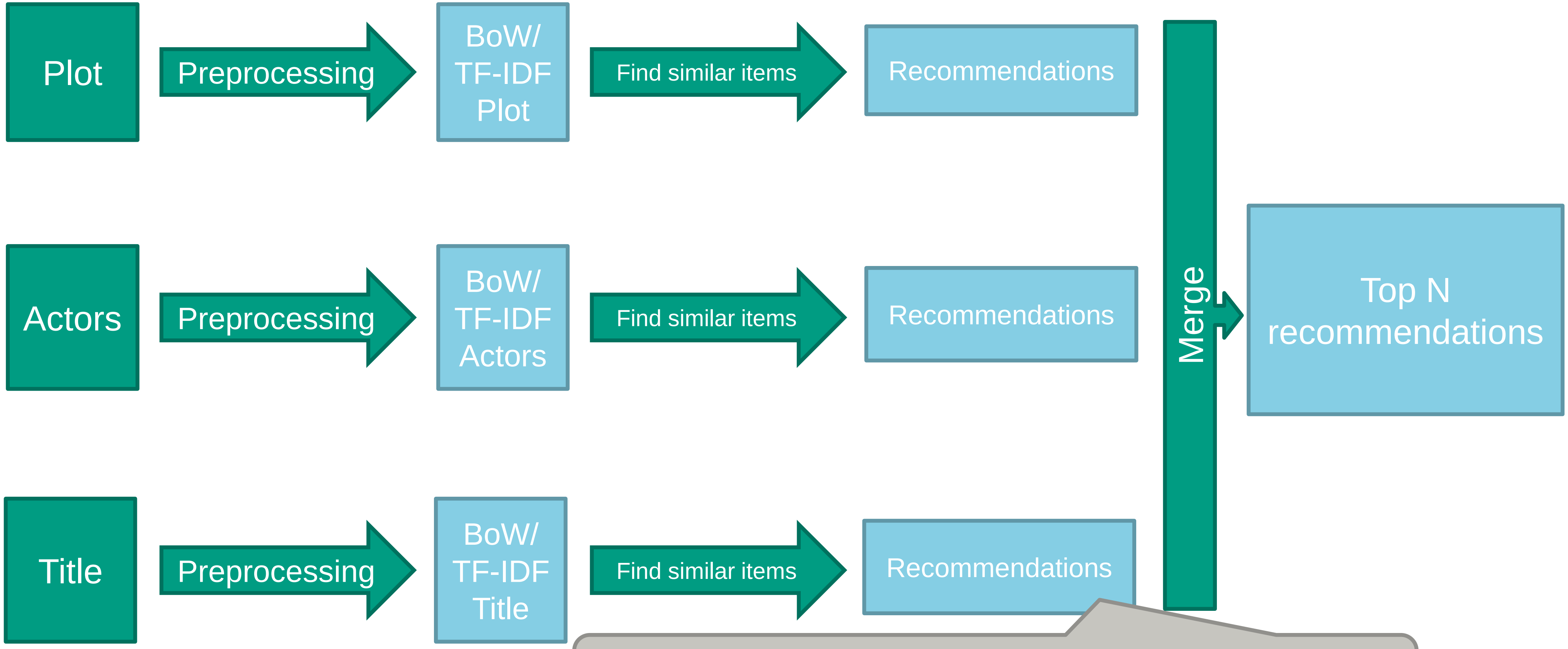
Recommender systems
Top-N list

The overall goal of a recommendation system is to:

Recommend n relevant items to the user

Question from last week:

‘How to deal with multiple attributes for recommending items?’



The merging phase is the most interesting. This is where you can add multiple merging rules to optimize your recommendations

Recommender systems
Approaches

Two possible approaches to build recommender systems:

1. **Content-based**
   Recommend items with the same properties

2. **Collaborative filtering**
   Recommend based on ratings of similar users

Collaborative filtering

CUSTOMERS WHO BOUGHT THIS ITEM:
ALSO BOUGHT:
piccolo

Collaborative filtering

The idea: recommending based on other peoples’ behavior

How to do it

Find users with similar taste

From these users, find items that they like and you haven't purchased yet

Limitations

Usual problem is that the data is very sparse (not enough ratings)

Assignment

For our assignment we have enough information (MovieLens data set)

Collaborative filtering by intuition

Movies

Users

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- Suppose we have the following user-item matrix (with ratings between 1 and 5 stars)
- Which user looks the most similar to user A?
- What would you recommend?

Collaborative filtering
2 types

User-user collaborative filtering

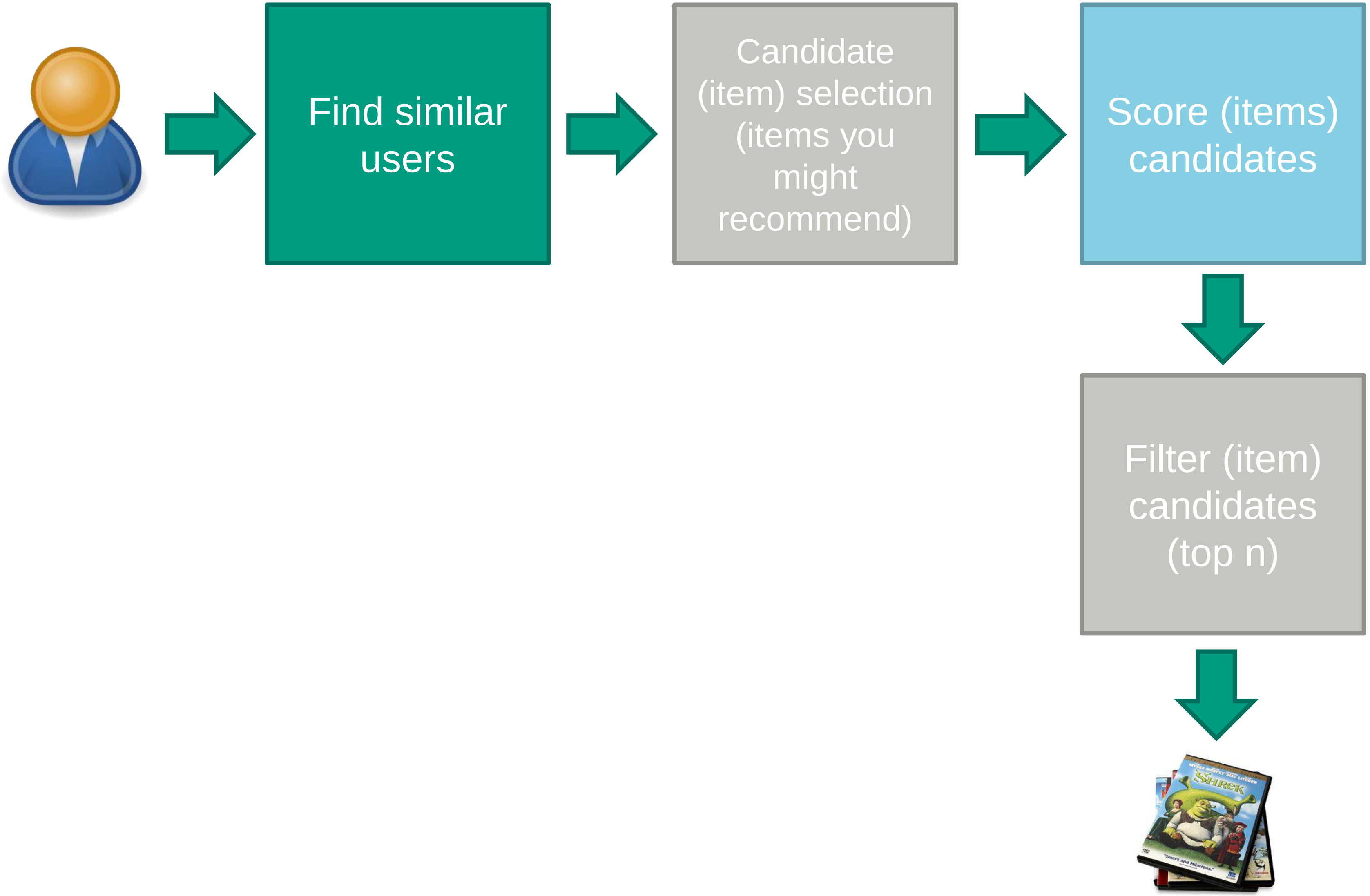
Find similar users (users that rate items the same way you do) and recommend items they liked

Item-item collaborative filtering

Find similar items (items with the same ratings) and recommend similar items that you have not yet seen

The two types are almost the same, especially when you *transpose* the user-item matrix

(User-User) Collaborative filtering



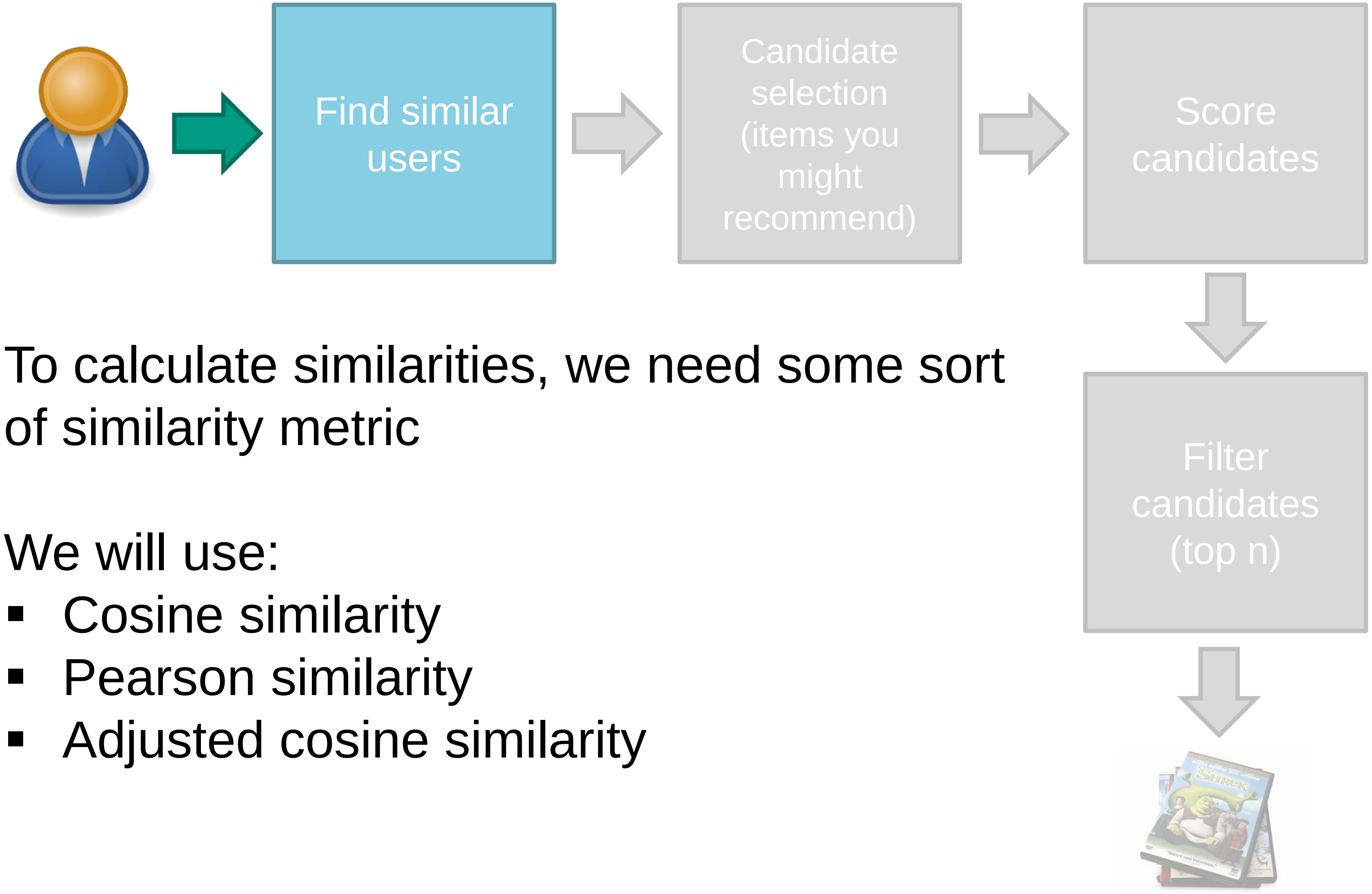
Find similar users

Candidate (item) selection (items you might recommend)

Score (items) candidates

Filter (item) candidates (top n)

Finding similar users

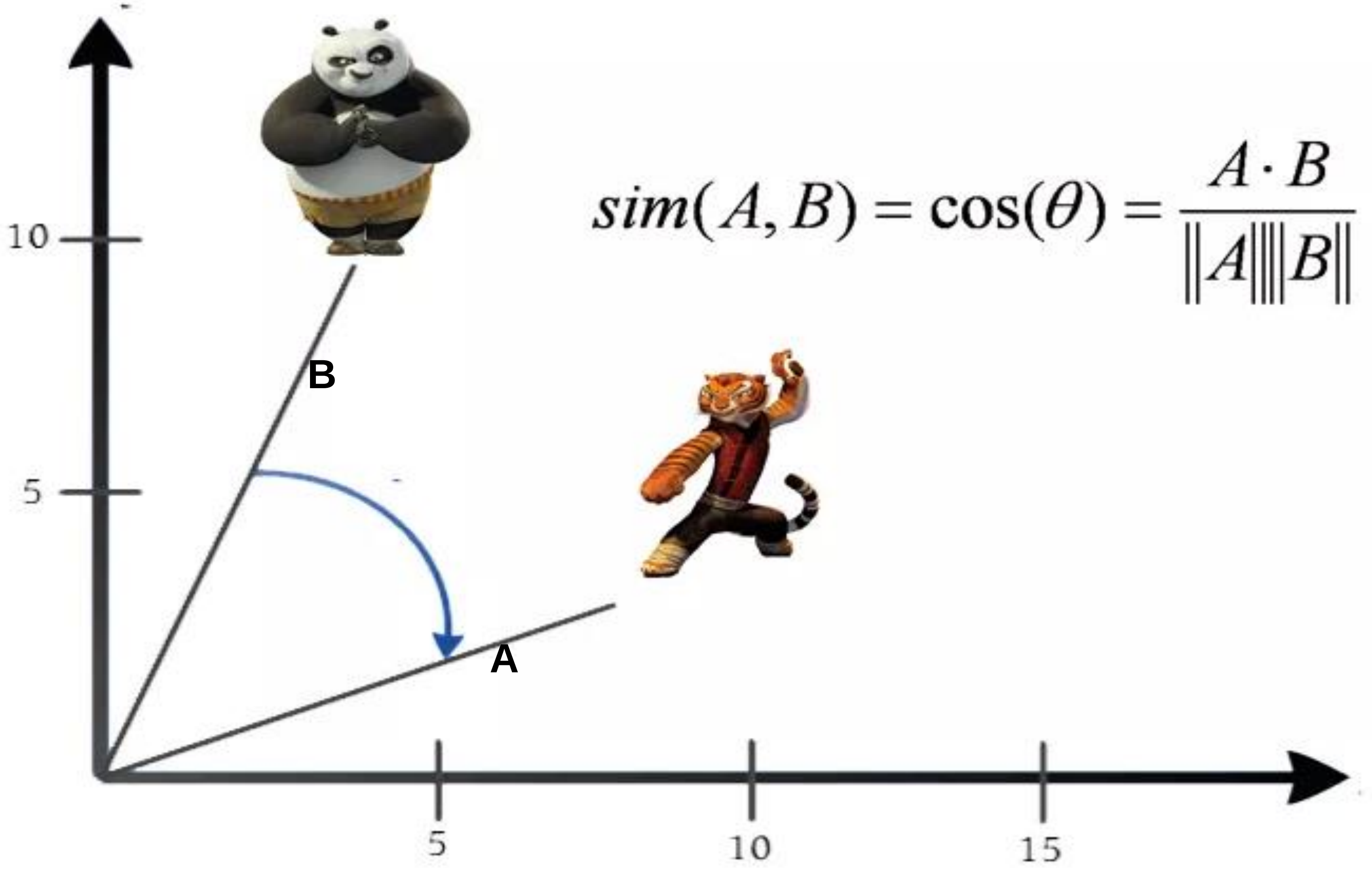

Find similar users
Candidate selection (items you might recommend)
Score candidates
Filter candidates (top n)

To calculate similarities, we need some sort of similarity metric

We will use:

- Cosine similarity
- Pearson similarity
- Adjusted cosine similarity

Similarity metrics



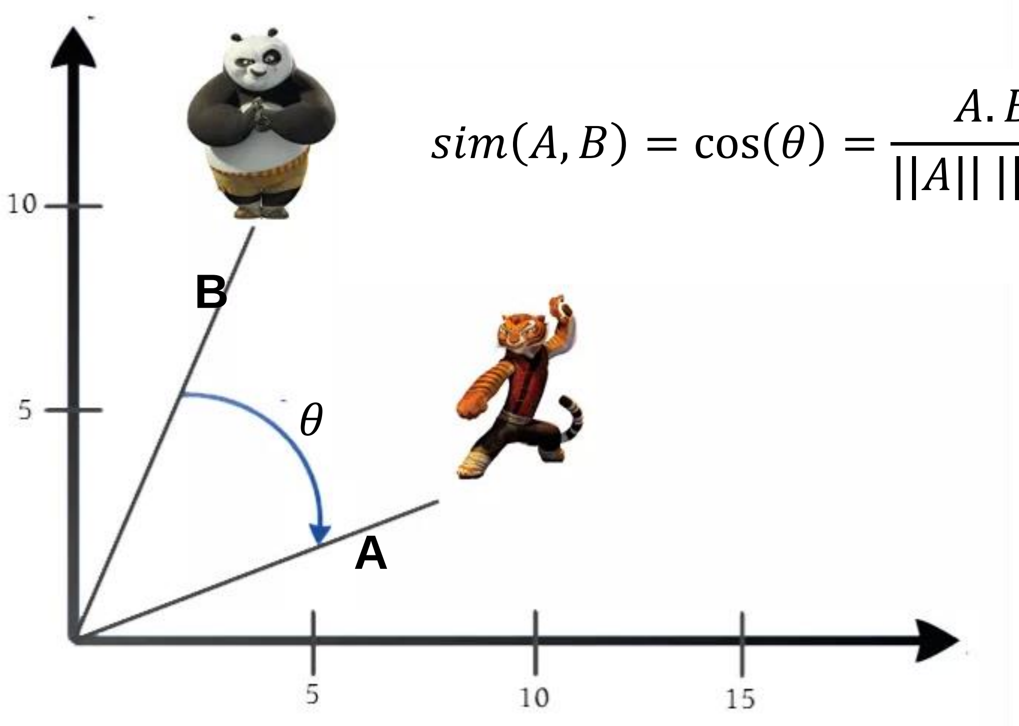
$$sim(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

Recap: Cosine similarity

The angle between vectors

Small angles are similar items

Cosine similarity = 1 means perfect match



$$sim(A,B) = \cos(\theta) = \frac{A.B}{||A||\,||B||} = \frac{\sum_i (A_i \times B_i)}{\sqrt{\sum_i A_i}\sqrt{\sum_i B_i}}$$

Cosine similarity
Formula

$i \in I$	: for every item i in list of items
u, v	: users u and v
u_i	: rating given by user u for item i

$$cosSim(u, v) = \frac{\sum_{i \in I}(u_i \times v_i)}{\sqrt{\sum_{i \in I} {u_i}^2}\sqrt{\sum_{i \in I} {v_i}^2}}$$

Cosine similarity
Example

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C	0	0	0	2	4	5	0
D	0	3	0	0	0	0	3

To calculate the cosine similarity, we need to fill in all the missing values

Let’s fill in 0 for all missing values and calculate the cosine similarity

Cosine similarity
Example

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C	0	0	0	2	4	5	0
D	0	3	0	0	0	0	3

Cosine similarity(A,B) = $\frac{4\times5+0\times5+0\times4+5\times0+1\times0+0\times0+0\times0}{\sqrt{4^2+0+0+5^2+1^2+0+0}\ \sqrt{5^2+5^2+4^20+0+0+0}} = \frac{20}{\sqrt{42}\sqrt{66}} = \mathbf{0.38}$

Cosine similarity(A,C) = **0.32**

0.38 > 0.32, so A and B are more similar than A and C

Cosine similarity
Problems with cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4	0	0	5	1	0	0
B	5	5	4	0	0	0	0
C	0	0	0	2	4	5	0
D	0	3	0	0	0	0	3

- Problem: low ratings and high ratings will have the same angle when calculating the cosine similarity
- Difference between ***A and B*** and ***A and C*** is small, whereas A and C are almost opposite users
- Solution: center around 0

Pearson similarity

Also known as Pearson correlation

Used for normalizing vectors (users’ ratings will be centered around 0)

How to calculate:

1. Subtract the mean (average) user rating from each user’s rating
2. Calculate the normal cosine similarity

$$pearsonSimilarity(u, v) = \frac{\sum_{i \in I}(u_i - \bar{u})\,(v_i - \bar{v})}{\sqrt{\sum_{i \in I}(u_i - \bar{u})^2}\,\sqrt{\sum_{i \in I}(v_i - \bar{v})^2}}$$

$i \in I$	: for every item i in list of items
u, v	: users u and v
u_i	: rating given by user u for item I
$\bar{u}$	: average rating of user u

Pearson similarity
1). Calculating the user average

	HP1	HP2	HP3	TW	SW1	SW2	SW3	Average
A	4			5	1			10/3
B	5	5	4					14/3
C				2	4	5		11/3
D		3					3	6/3

Calculate the average rating for each user

Example:
Average of A = (4 + 5 + 1) / 3 = 10/3
Average of B = (5 + 5 + 4) / 3 = 14/3

Pearson similarity

2). Subtract the averages and fill in zeros

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	$4 - \frac{10}{3}$ =**2/3**	0	0	5/3	-7/3	0	0
B	1/3	1/3	-2/3	0	0	0	0
C	0	0	0	-5/3	1/3	4/3	0
D	0	0	0	0	0	0	0

Average
10/3
14/3
11/3
3

1. Subtract the average of a user from its ratings
2. The negative values represent negative ratings, positive values represent positive ratings
3. The value 0 is now the average rating for a user

Pearson similarity
3). Calculating the cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3	0	0	5/3	-7/3	0	0
B	1/3	1/3	-2/3	0	0	0	0
C	0	0	0	-5/3	1/3	4/3	0
D	0	0	0	0	0	0	0

Cosine similarity(A,B) = 0.092
Cosine similarity(A,C) = -0.559

0.092 > -0.559, so A and B are more similar than A and C

Adjusted Cosine Similarity Formulas

Slight variation on the Pearson similarity

Now, we subtract the average item rating from each user rating for a given item (when we calculate the difference between users)

How much does a user deviate from the average?

$$adjustedCosineSimilarity(u, v) = \frac{\sum_{i \in I}(u_i - \bar{\imath})\,(v_i - \bar{\imath})}{\sqrt{\sum_{i \in I}(u_i - \bar{\imath})^2}\,\sqrt{\sum_{i \in I}(v_i - \bar{\imath})^2}}$$

$i \in I$	: for every item i in list of items
u, v	: users u and v
u_i	: rating given by user u for item I
$\bar{\imath}$	: average rating of item i

Adjusted Cosine Similarity
1). Calculating the item average

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3
AVG	9/2	4	4	7/2	5/2	5	3

Calculate the average rating for each item

Example:

Average of HP1 = (4 + 5) / 2 = 9/2

Average of HP2 = (5 + 3) / 2 = 4

Adjusted Cosine Similarity
2). Subtracting the average

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	-1/2			3/2	-3/2		
B	1/2	1	0				
C				-3/2	3/2	0	
D		-1					0
AVG	9/2	4	4	7/2	5/2	5	3

Calculate the average rating for each item

Example:

Average of HP1 = (4 + 5) / 2 = 9/2
Average of HP2 = (5 + 3) / 2 = 4

Adjusted Cosine Similarity
3). Calculating the cosine similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	-1/2			3/2	-3/2		
B	1/2	1	0				
C				-3/2	3/2	0	
D		-1					0

AVG	9/2	4	4	7/2	5/2	5	3

Cosine similarity(A,B) = -0.1025
Cosine similarity(A,C) = -0.9733

-0.1025 > -0.9733, so A and B are more similar than A and C

Pearson similarity vs. adjusted cosine similarity

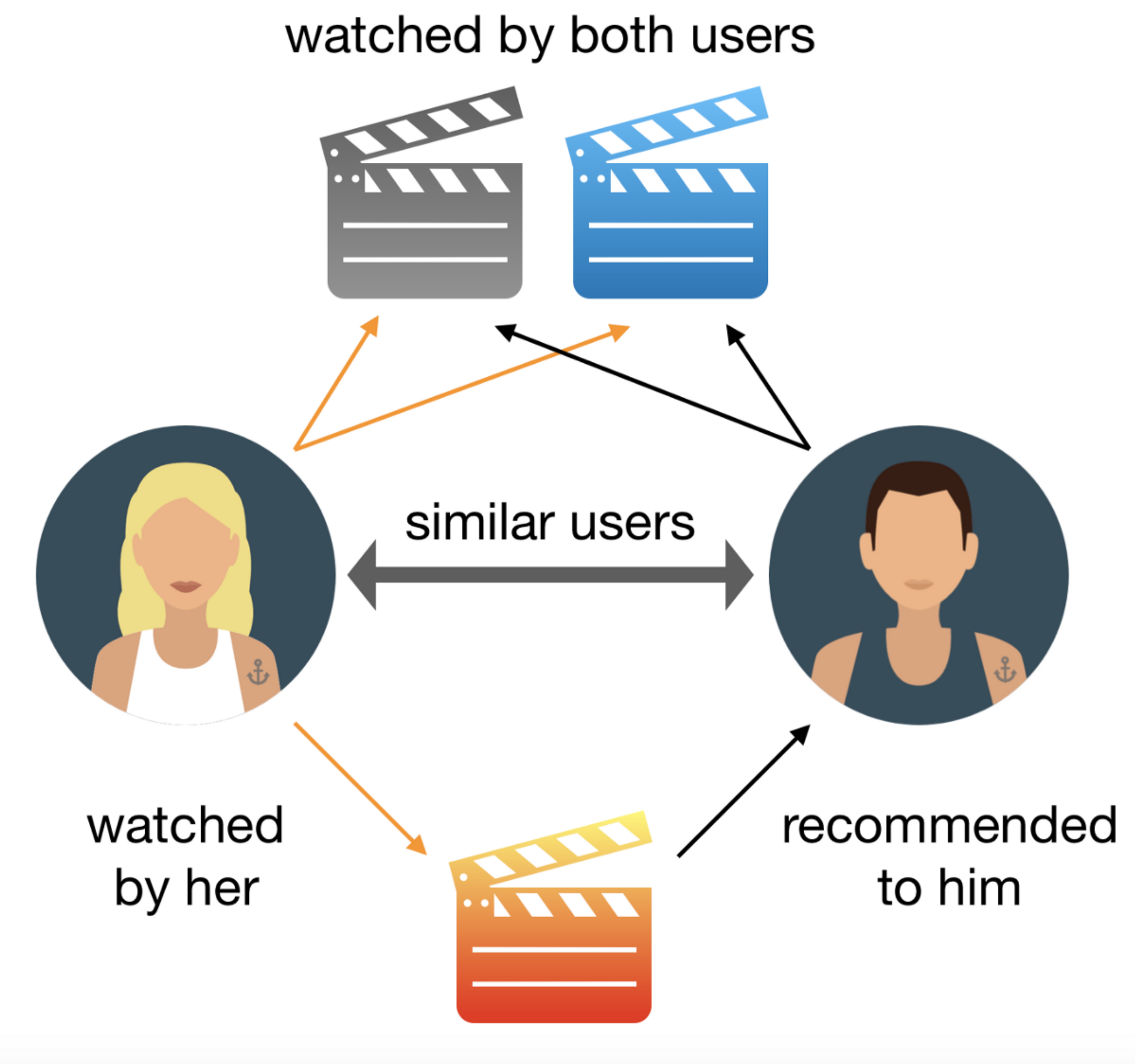
Almost the same measures

- Pearson subtracts the row mean
- Adjusted cosine subtracts the column mean

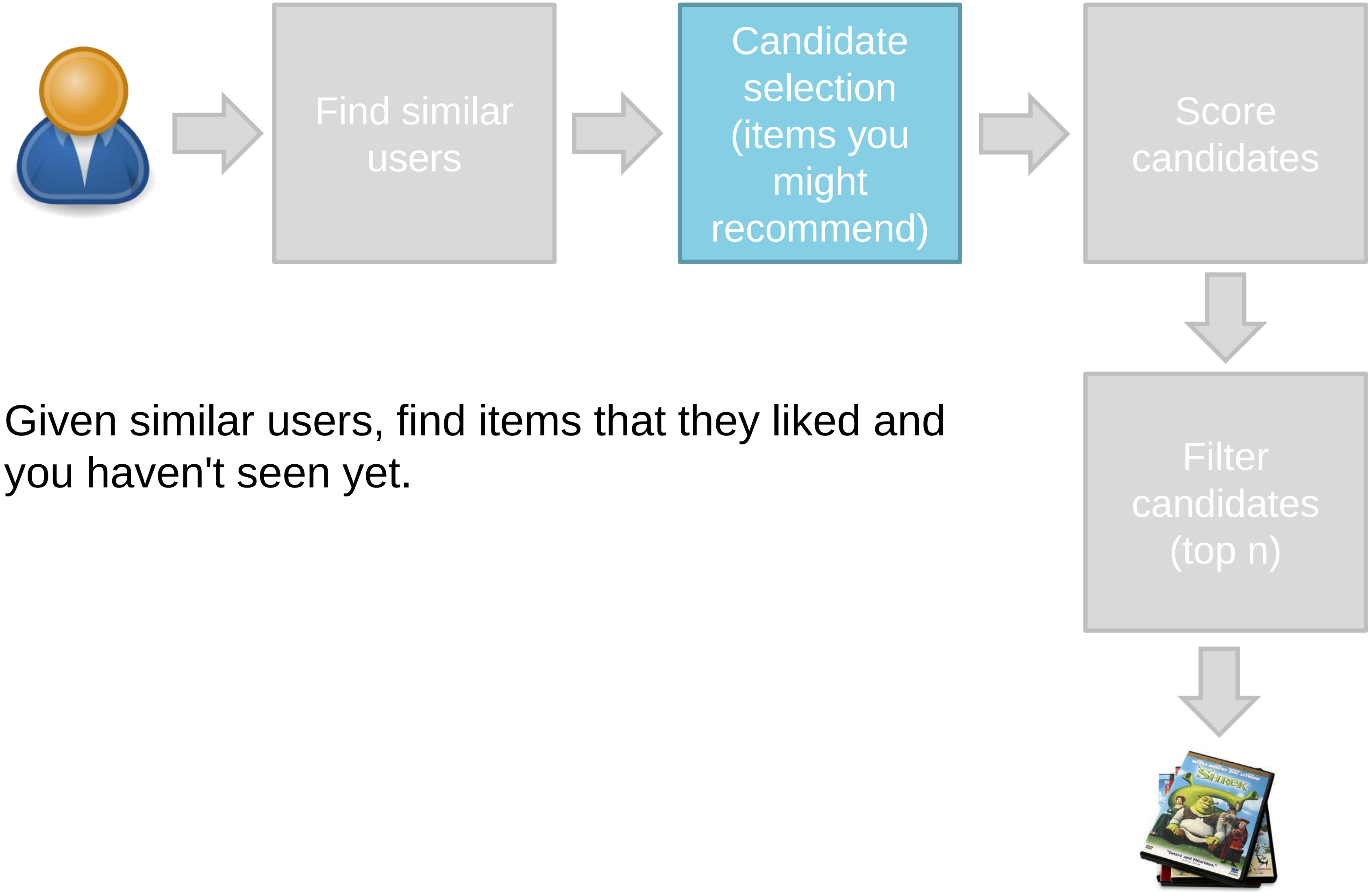
Both applicable for user-user and item-item recommendations

Simply switch around users and items in the formulas!

User-user collaborative filtering

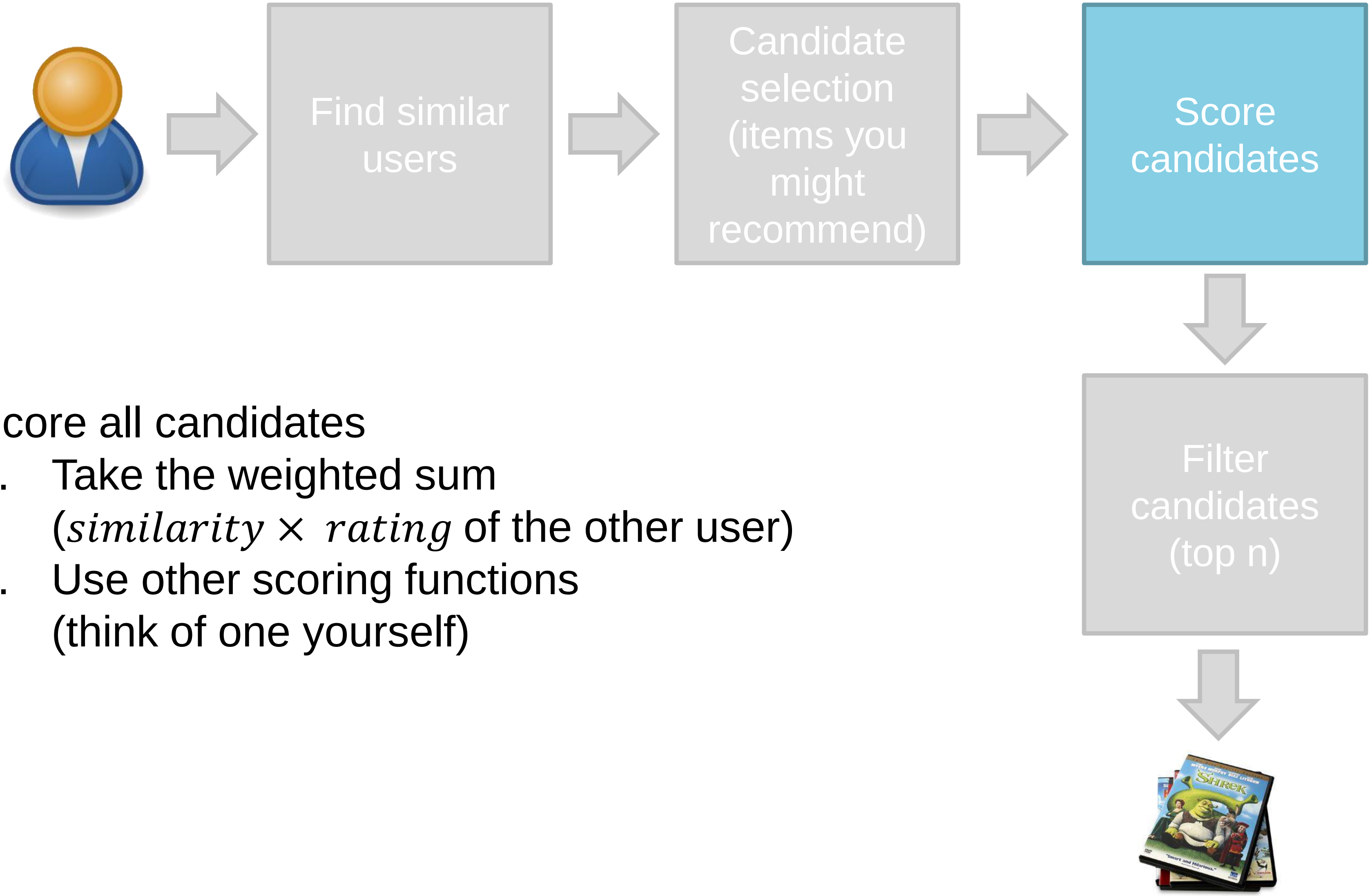


(User-User) Collaborative filtering


Find similar users
Candidate selection (items you might recommend)
Score candidates
Filter candidates (top n)

Given similar users, find items that they liked and you haven't seen yet.

(User-User) Collaborative filtering



Find similar users

Candidate selection (items you might recommend)

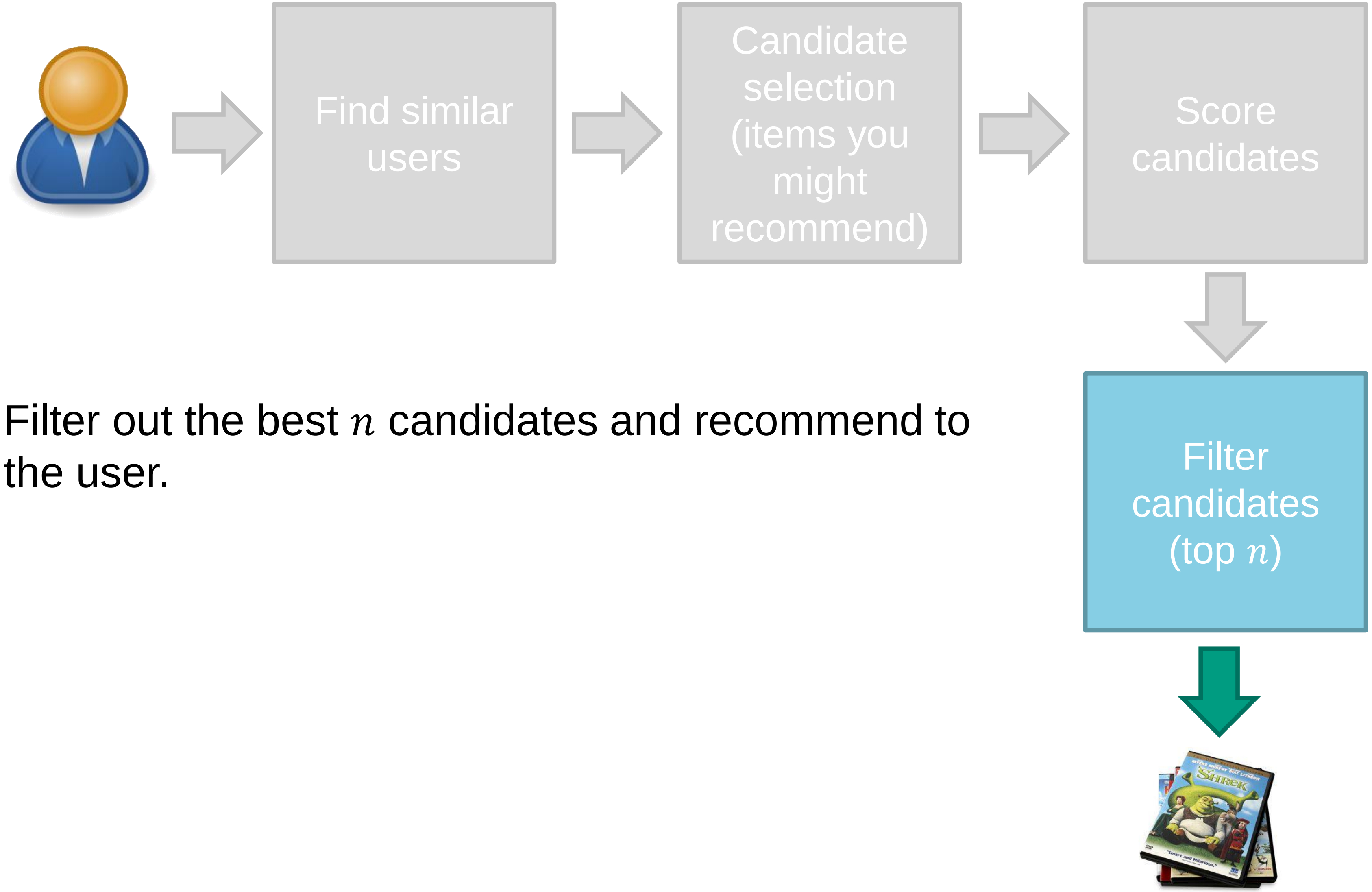
Score candidates

Filter candidates (top n)

Score all candidates

1. Take the weighted sum ($similarity \times rating$ of the other user)
2. Use other scoring functions (think of one yourself)

(User-User) Collaborative filtering


Find similar users
Candidate selection (items you might recommend)
Score candidates
Filter candidates (top n)

Filter out the best n candidates and recommend to the user.

User-user collaborative filtering: Example

Movies

User

	1	2	3	4	5	6
A	1		2			1
B			4	2		
C	3	5		4	4	3
D		4	1		3	
E			2	5	4	3
F	5				2	
G		4	3			
H				4		2
I	5		4			
J		2	3			
K	4	1	5	2	2	4
L		3		5		

The yellow cells represent known ratings

Question: Recommend movies to user A by finding the $N = 2$ most similar users

User-user collaborative filtering: Example

Movies

User

	1	2	3	4	5	6
A	1		2			1
B			4	2		
C	3	5		4	4	3
D		4	1		3	
E			2	5	4	3
F	5				2	
G		4	3			
H				4		2
I	5		4			
J		2	3			
K	4	1	5	2	2	4
L		3		5		

First, we calculate the user-user similarity matrix (see earlier slides) with Pearson correlation

Users

Users

	A	B	C	D	E	F	G	H	I	J	K	L
A	1	0.57	0.39	-0.63	-0.46	-0.29	-0.58	0.29	-0.87	0.58	0.24	0
B	0.57	1	…	…	…	…	…	…	…	…	…	…
C	0.39	…	1	…	…	…	…	…	…	…	…	…
D	-0.63	…	…	1	…	…	…	…	…	…	…	…
E	-0.46	…	…	…	1	…	…	…	…	…	…	…
F	-0.29	…	…	…	…	1	…	…	…	…	…	…
G	-0.58	…	…	…	…	…	1	…	…	…	…	…
H	0.29	…	…	…	…	…	…	1	…	…	…	…
I	-0.87	…	…	…	…	…	…	…	1	…	…	…
J	0.58	…	…	…	…	…	…	…	…	1	…	…
K	0.24	…	…	…	…	…	…	…	…	…	1	…
L	0	…	…	…	…	…	…	…	…	…	…	1

User-user collaborative filtering: Calculate similarities

Movies

User

	1	2	3	4	5	6	A
A	1		2			1	1
B			4	2			0.57
C	3	5		4	4	3	0.39
D		4	1		3		-0.63
E			2	5	4	3	-0.46
F	5				2		-0.29
G		4	3				-0.58
H				4		2	0.29
I	5		4				-0.87
J		2	3				0.58
K	4	1	5	2	2	4	0.24
L		3		5			0

In this case, we are only interested in the similarities between user A and the other users

User-user collaborative filtering: Calculate similarities

Movies

User

	1	2	3	4	5	6	A
A	1		2			1	1
B			4	2			0.57
C	3	5		4	4	3	0.39
D		4	1		3		-0.63
E			2	5	4	3	-0.46
F	5				2		-0.29
G		4	3				-0.58
H				4		2	0.29
I	5		4				-0.87
J		2	3				0.58
K	4	1	5	2	2	4	0.24
L		3		5			0

User B and User J are most similar to User A

User-user collaborative filtering: Calculate similarities

Movies

User

	1	2	3	4	5	6	A
A	1		2			1	1
B			4	2			0.57
C	3	5		4	4	3	0.39
D		4	1		3		-0.63
E			2	5	4	3	-0.46
F	5				2		-0.29
G		4	3				-0.58
H				4		2	0.29
I	5		4				-0.87
J		2	3				0.58
K	4	1	5	2	2	4	0.24
L		3		5			0

In this case, movie 4 and movie 2 could be recommended

When there are more movies (more than N), we could score the results and select the N best:

E.g., $similarity\ *\ movie\ score$

(0.57 * 2 and 0.58 * 2)

Questions

Assignment

Assignment 1: recommender systems
Many websites give users the possibility to rate items nowadays. Companies such as Amazon, Netflix, YouTube, IMDB and Bol.com use this information to recommend similar items to their users. The MovieLens dataset is a free dataset with a collection of movie ratings.

In this assignment you will build two recommendation systems, using the following techniques: content-based and **collaborative filtering**.

Pro-tip: finish this assignment by week 4

References

- Mining Massive Datasets (mmds.org)
- Building recommender systems with Machine Learning and AI
- https://md.ekstrandom.net/blog/2015/06/item-similarity