Outline

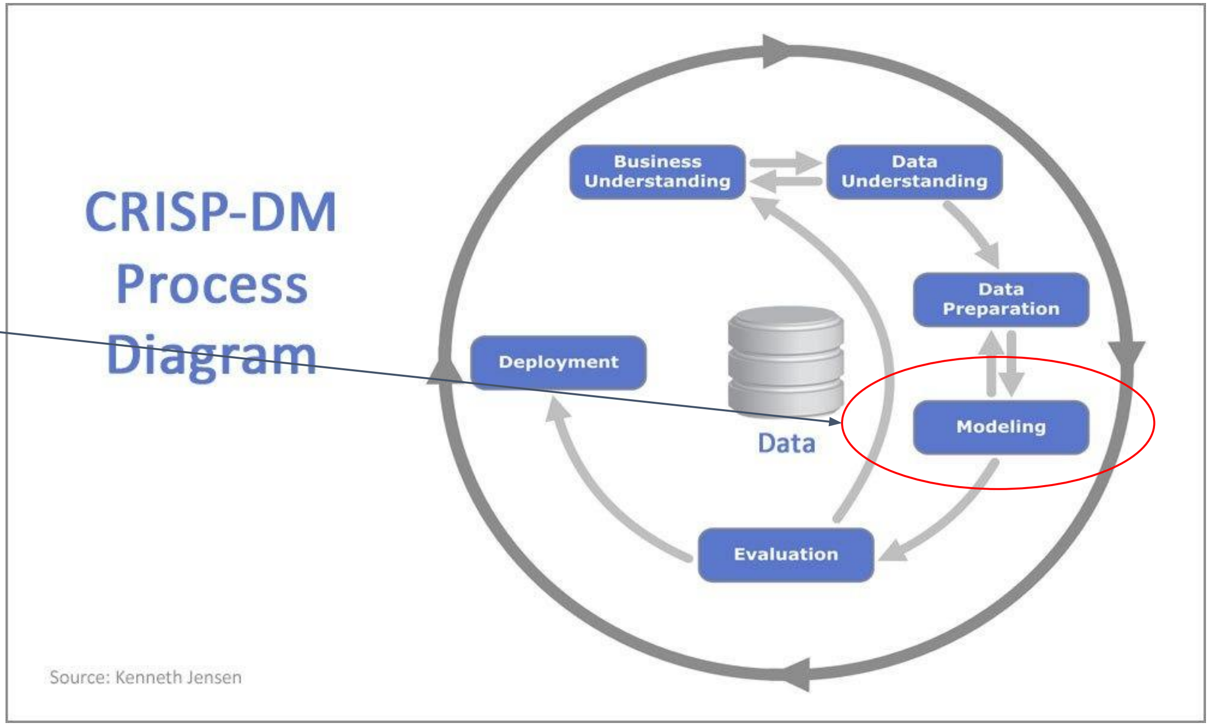
What Is Recommendation System ?

Content based

- similarity measure
- content based filtering

Collaborative Filtering

- SVD
- ALS
- WALS



What Is Recommendation System ?

Recommendation system are about user personalization

Recommendation system can allow you to sell products you could never sell before to user you could never reach before

Recommendation system can utilize any kind of data, such as:

- structured data like tabular data
- text
- images, etc

Purwadhika
Startup and Coding School

Who Use Recommendation System ?

amazon

YouTube

NETFLIX

Why Do We Need a Recommendation System ?

- user often hard to choose items what they really want
- so many items that can be chosen
  - which book should I read ?
  - which movie should I watch ?
- make user keep using product/items. this also means more money

Type Of Recommendation System

Common type :

- Content Based Recommendation System
- Collaborative Filtering

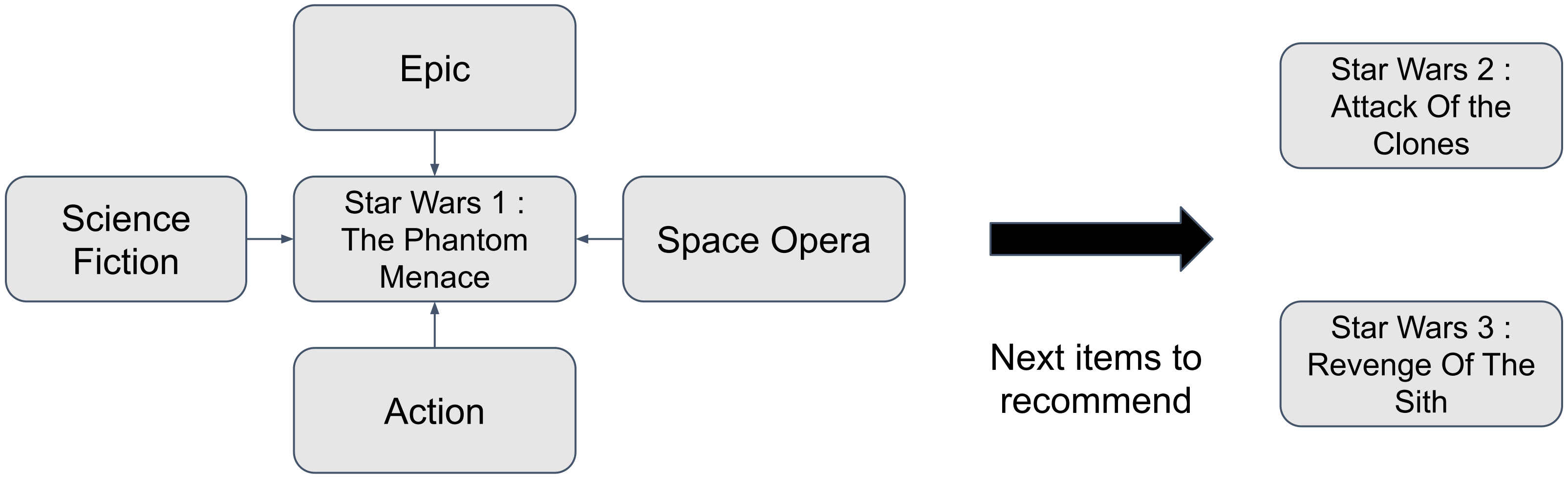
There is also another type like hybrid, knowledge based and demographic filtering. We will focus to content based and collaborative filtering

Content Based Recommendation System

Purwadhika
Startup and Coding School

Content Based Recommendation System

- uses item features to recommend other items similar to what the user likes, based on their previous actions or explicit feedback
- we must construct for each item a profile



- The film's genre (action, sci-fi, space opera, epic) is refer to as metadata/contents
- for this case, metade you can use are not only limited to genre
- you can use another metadata such as:
  - cast
  - studio
  - director, etc

What Content to Represent These Items ?

Author, year of publication, genre

Artist, composer, genre

Image data

Text data

Purwadhika
Startup and Coding School

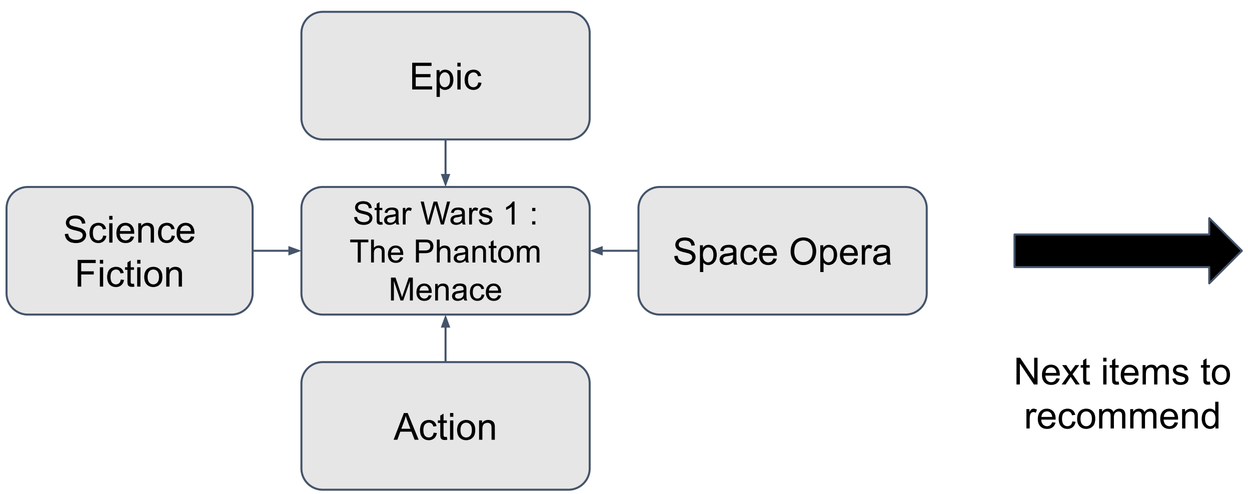
Content Based Recommendation System

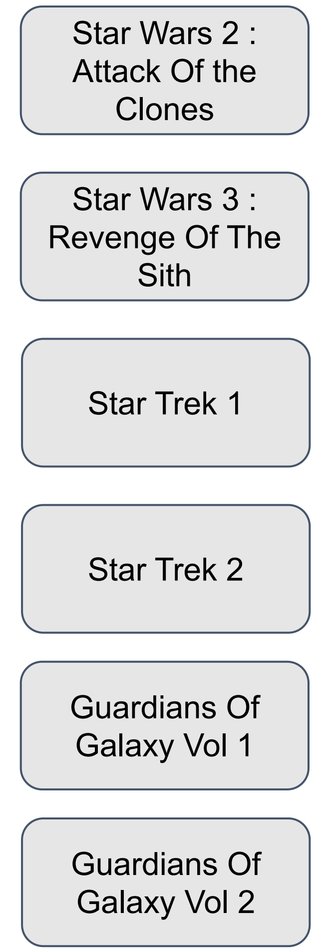
- Similarity Measure
- Content Based Filtering

Content Based Recommendation with Similarity Measure

Content Based Recommendation with Similarity Measure

Utilize an item that we already used before to recommend next items.

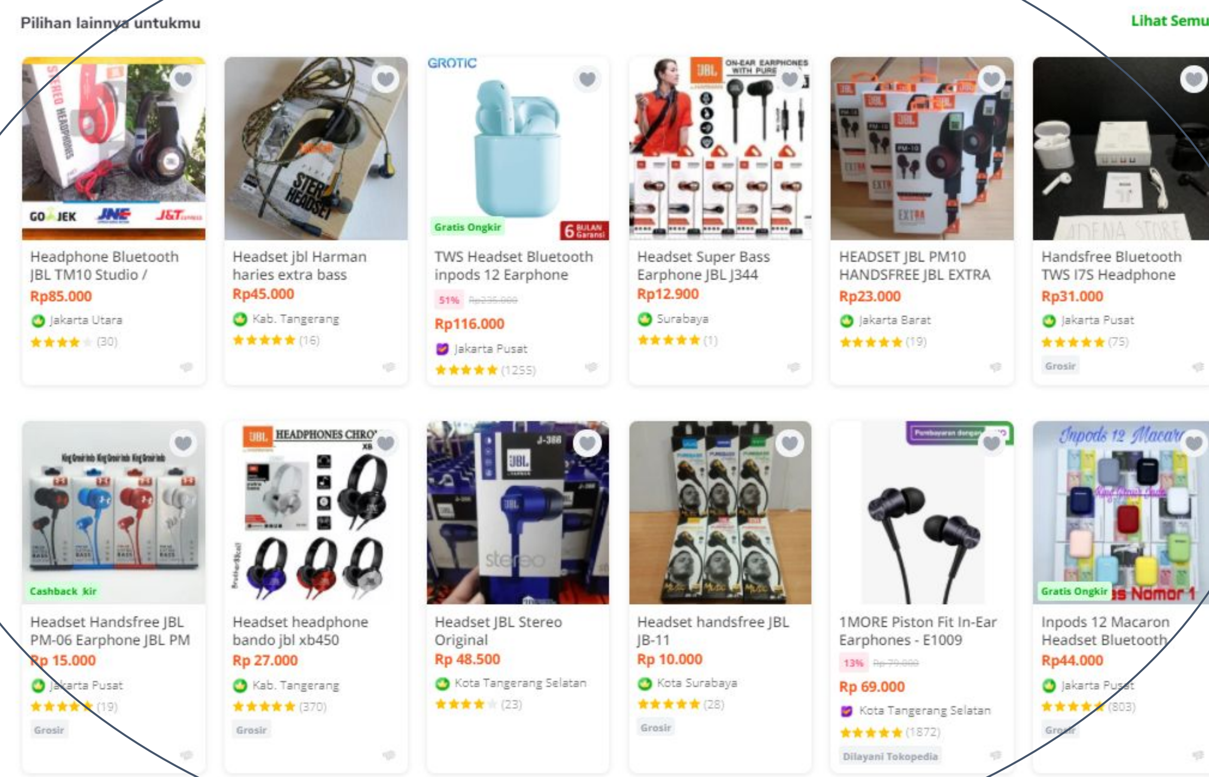




Purwadhika
Startup and Coding School

Example

Recommendation Result



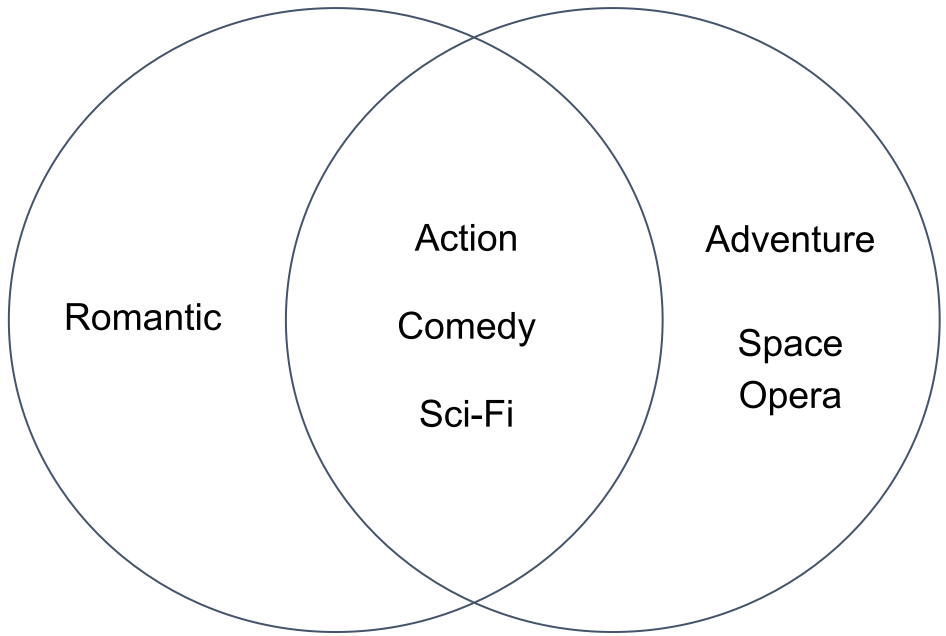
Source :
https://www.tokopedia.com/charger/headset-jbl-stereo-ori

Similarity Measure

Measure similarity between two items based on certain content:

- **cosine similarity**
- pearson correlation
- spearman correlation
- Jaccard distance
- etc

What It Means to Be Similar

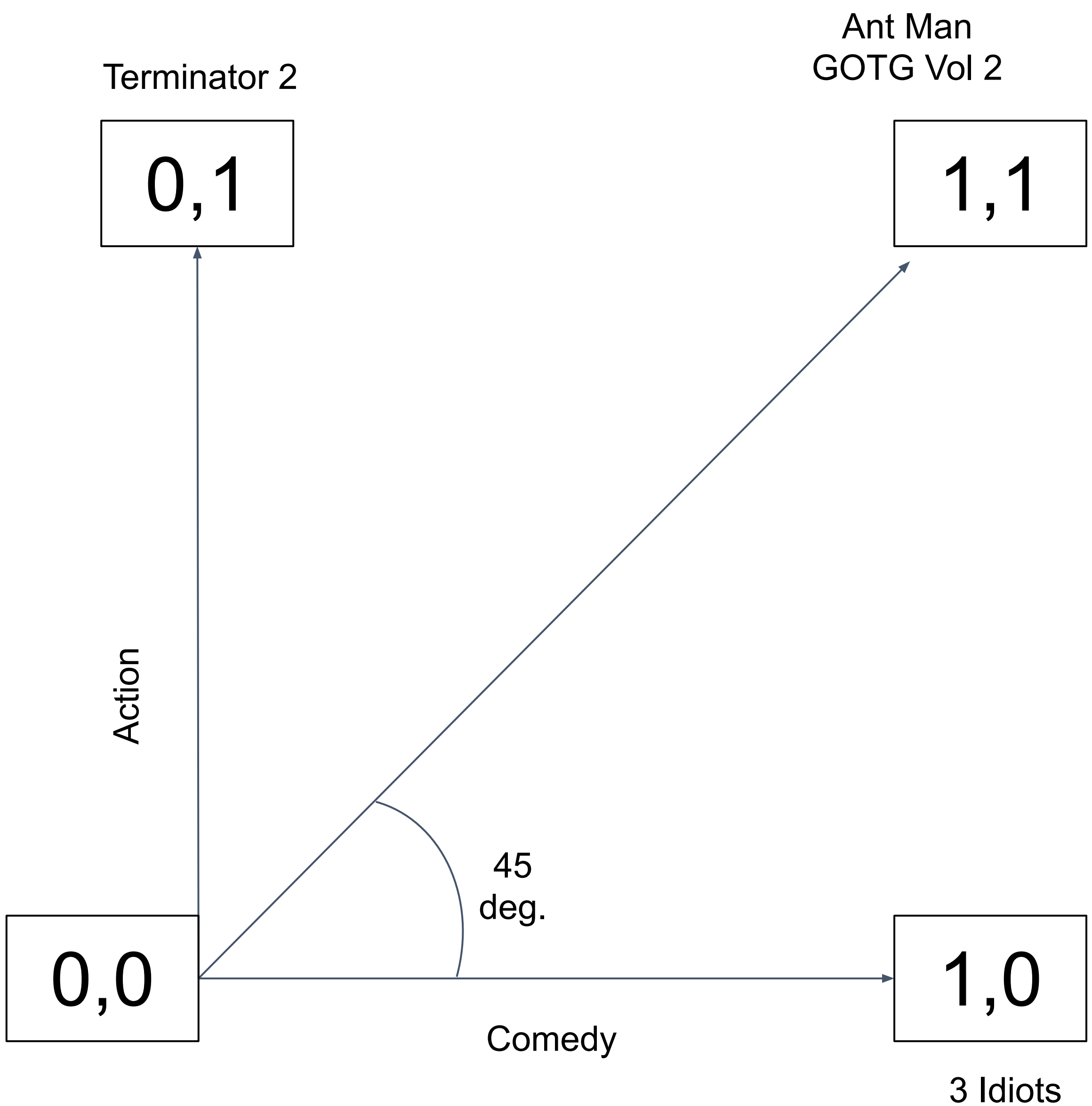


Purwadhika
Startup and Coding School

Cosine Similarity

Movie	Genre	
	Comedy	Action
Terminator 2	0	1
Ant Man 2	1	1
GOTG Vol 2	1	1
3 idiots	1	0

Movie That already watched



Purwadhika
Startup and Coding School

Cosine Similarity Formula

Movie	Genre			
	Comedy	Action	Sci-Fi	Romantic
Terminator 2	0	1	1	0
Ant Man 2	1	1	1	1
GOTG Vol 2	1	1	1	0
3 idiots	1	0	0	0

Movie	Genre			
	Genre 1	Genre 2	...	Genre k
Movie 1	1	1	...	1
Movie 2	0	0	...	0
….	...	...	...	...
Movie n	1	1	0	0

$$Cosine(AntMan2, GotgVol2) = \frac{1*1+1*1+1*1+1*0}{\sqrt{1^2+1^2+1^2+1^2}\sqrt{1^2+1^2+1^2+0^2}} = 0.8660...$$

$$Cosine(AntMan2, Terminator2) = \frac{1*0+1*1+1*1+1*0}{\sqrt{1^2+1^2+1^2+1^2}\sqrt{0^2+1^2+1^2+0^2}} = 0.7071...$$

$$Cosine(AntMan2, 3Idiots) = \frac{1*1+1*0+1*0+1*0}{\sqrt{1^2+1^2+1^2+1^2}\sqrt{1^2+0^2+0^2+0^2}} = 0.5$$

$$Cosine(x, y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

value between 0 and 1
1 : exactly the same
0 : exactly different
closer to 1 two items are getting similar

Purwadhika
Startup and Coding School

Recommendation

Let's say someone already watched Ant-Man 2. Movies that will be recommended (in order) are GOTG Vol 2, Terminator 2, dan 3 Idiots.

Watched Movies

Ant Man 2

Recommendation

GOTG Vol 2 (0.8660..)

Terminator 2 (0.7071…)

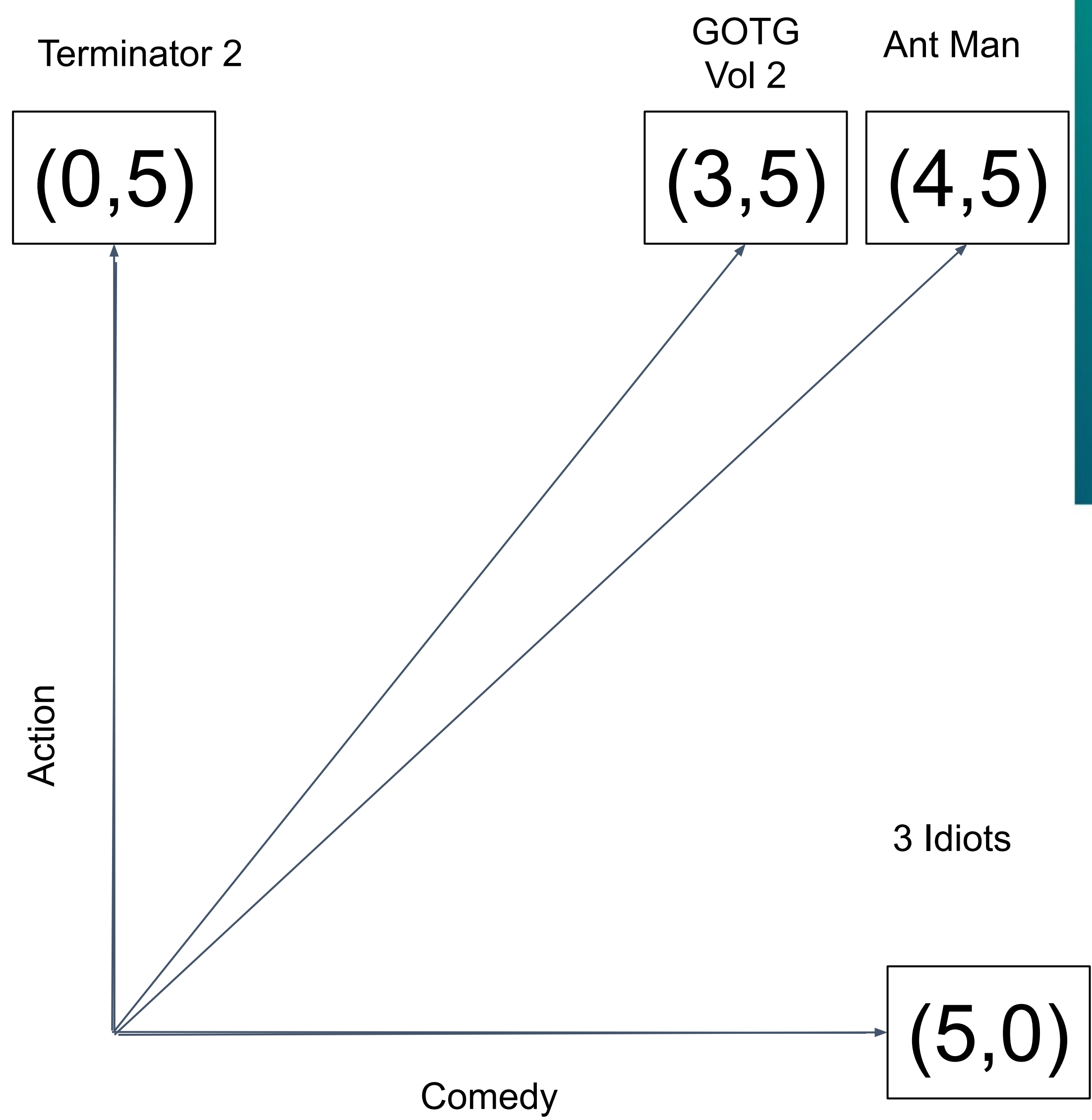
3 Idiots (0.5)

Cosine Similarity

Degree of the content

<table>
<tr><th rowspan="2">Movie</th><th colspan="2">Genre</th></tr>
<tr><th>Comedy</th><th>Action</th></tr>
<tr><td>Terminator 2</td><td>0</td><td>5</td></tr>
<tr><td>Ant Man 2</td><td>4</td><td>5</td></tr>
<tr><td>GOTG Vol 2</td><td>3</td><td>5</td></tr>
<tr><td>3 idiots</td><td>5</td><td>0</td></tr>
</table>

Movie That already watched



Purwadhika
Startup and Coding School

Cosine Similarity Formula

Movie	Genre			
	Comedy	Action	Sci-Fi	Romantic
Terminator 2	0	5	5	0
Ant Man 2	4	5	5	4
GOTG Vol 2	3	5	5	0
3 idiots	4	0	0	0

Movie	Genre			
	Genre 1	Genre 2	...	Genre k
Movie 1	a11	a21	...	ak1
Movie 2	a12	a22	...	ak2
….	...	...	...	...
Movie n	a1n	a2n	...	akn

$$Cosine(AntMan2, GotgVol2) = \frac{4*3+5*5+5*5+4*0}{\sqrt{4^2+5^2+5^2+4^2}\sqrt{3^2+5^2+5^2+0^2}} = 0.8913...$$

$$Cosine(AntMan2, Terminator2) = \frac{4*0+5*5+5*5+4*0}{\sqrt{4^2+5^2+5^2+4^2}\sqrt{0^2+5^2+5^2+0^2}} = 0.7808...$$

$$Cosine(AntMan2, 3Idiots) = \frac{4*5+5*0+5*0+4*0}{\sqrt{4^2+5^2+5^2+4^2}\sqrt{4^2+0^2+0^2+0^2}} = 0.5521...$$

$$Cosine(x,y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

value between 0 and 1
1 : exactly the same
0 : exactly different
closer to 1 two items are getting similar

Purwadhika
Startup and Coding School

Recommendation

Movies that will be recommended (in order) are GOTG Vol 2, Terminator 2, dan 3 Idiots.

Recommendation

1. GOTG Vol 2
2. Terminator 2
3. 3 Idiots

Watched Movies

Ant Man 2

Summary

Determine which content to use for profiling

↓

Choose an item that already liked before

↓

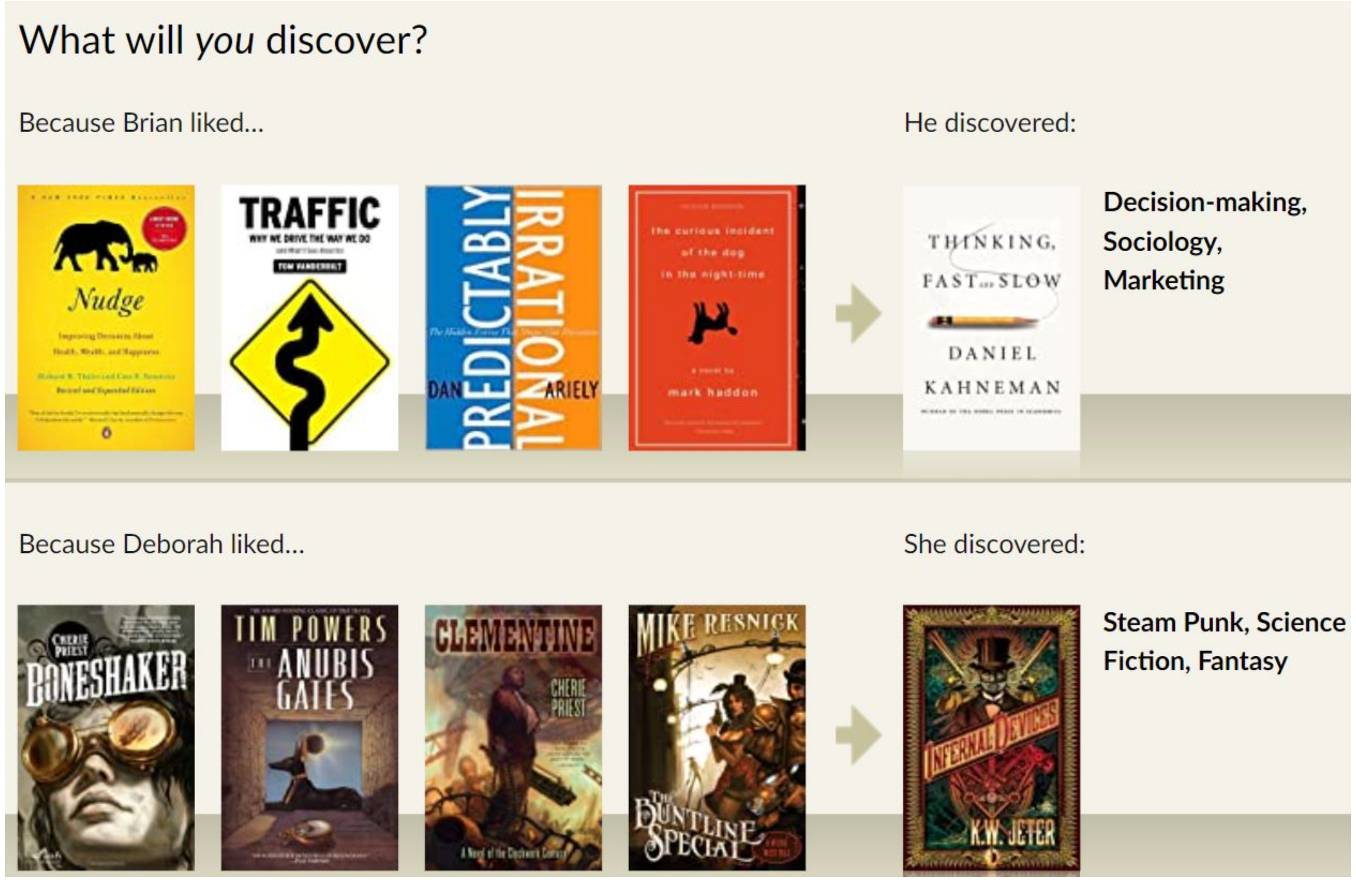
Find the most similar items

Purwadhika
Startup and Coding School

Content Based Filtering

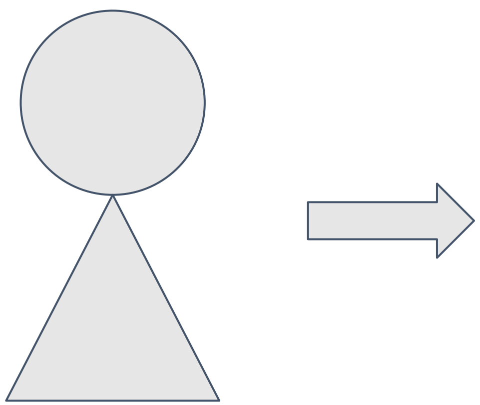
Purwadhika
Startup and Coding School

Content Based Filtering



Utilize several items that we already used before to recommend next items.

Content Based Filtering



User 1

Movie	Score	Action	Sci-Fi	Adventure	Comedy	Drama
Terminator 2	7	1	1	0	0	0
Interstellar	9	0	1	1	0	1
Ant Man 2	8	1	1	1	1	0
3 Idiots	9	0	0	0	1	1
Titanic	?	0	0	0	0	1
Martian	?	0	1	1	0	1
GOTG Vol 2	?	1	1	1	1	0

User Score Rating

Purwadhika
Startup and Coding School

Movie	Score	Action	Sci-Fi	Adventure	Comedy	Drama
Terminator 2	7	1	1	0	0	0
Interstellar	9	0	1	1	0	1
Ant Man 2	8	1	1	1	1	0
3 Idiots	9	0	0	0	1	1

Action	Sci-Fi	Adventure	Comedy	Drama
7	7	0	0	0
0	9	9	0	9
8	8	8	8	0
0	0	0	9	9

Action	Sci-Fi	Adventure	Comedy	Drama
15	24	17	17	18

Action	Sci-Fi	Adventure	Comedy	Drama
0.164	0.263	0.186	0.186	0.197

User Feature Vector

- Sci-Fi has the strongest influence for this user
- Action has the least influence for this user

Recommendation

	Action	Sci-Fi	Adventure	Comedy	Drama
	0.164	0.263	0.186	0.186	0.197

Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Titanic	1	1	0	0	0
Martian	0	1	1	0	1
GOTG Vol 2	1	1	1	1	0

Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Titanic	0.164	0.263	0	0	0
Martian	0	0.263	0.186	0	0.197
GOTG Vol 2	0.164	0.263	0.186	0.186	0

Movie	Movie Scoring
Titanic	0.429
Martian	0.648
GOTG Vol 2	0.802

Recommendation Order for the user:

1. GOTG Vol 2
2. Martian
3. Titanic

Purwadhika
Startup and Coding School

Content Based Filtering for Multiple User

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8		6	5
User 3	9			10
User 4		7		9

User-item rating matrix

Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Terminator 2	1	1			
Interstellar		1	1		1
Ant Man 2	1	1	1	1	
3 Idiots				1	1

Item-feature matrix

Purwadhika
Startup and Coding School

User Feature Matrix

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8		6	5
User 3	9			10
User 4		7		9

Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Terminator 2	1	1			
Interstellar		1	1		1
Ant Man 2	1	1	1	1	
3 Idiots				1	1

User Feature Matrix

User	Action	Sci-Fi	Adventure	Comedy	Drama
User 1	0.164	0.263	0.186	0.186	0.197
User 2	0.28	0.28	0.12	0.22	0.1
User 3	0.23	0.237	0	0.263	0.263
User 4	0	0.179	0.179	0.230	0.410

User Feature Vector

Inferred Movie Rankings

User	Action	Sci-Fi	Adventure	Comedy	Drama
User 1	0.164	0.263	0.186	0.186	0.197
User 2	0.28	0.28	0.12	0.22	0.1
User 3	0.23	0.237	0	0.263	0.263
User 4	0	0.179	0.179	0.230	0.410

Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Terminator 2	1	1			
Interstellar		1	1		1
Ant Man 2	1	1	1	1	
3 Idiots				1	1

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1				
User 2		0.5		
User 3		0.5	0.736	
User 4	0.1794		0.589	

Recommendation Order for unwatched movies

- User 3 : Ant Man 2, Interstellar
- User 4 : Ant Man 2, Terminator 2

Purwadhika

Startup and Coding School

Inferred Movie Rankings (Not Watched at All)

User	Action	Sci-Fi	Adventure	Comedy	Drama
User 1	0.164	0.263	0.186	0.186	0.197
User 2	0.28	0.28	0.12	0.22	0.1
User 3	0.23	0.237	0	0.263	0.263
User 4	0	0.179	0.179	0.230	0.410

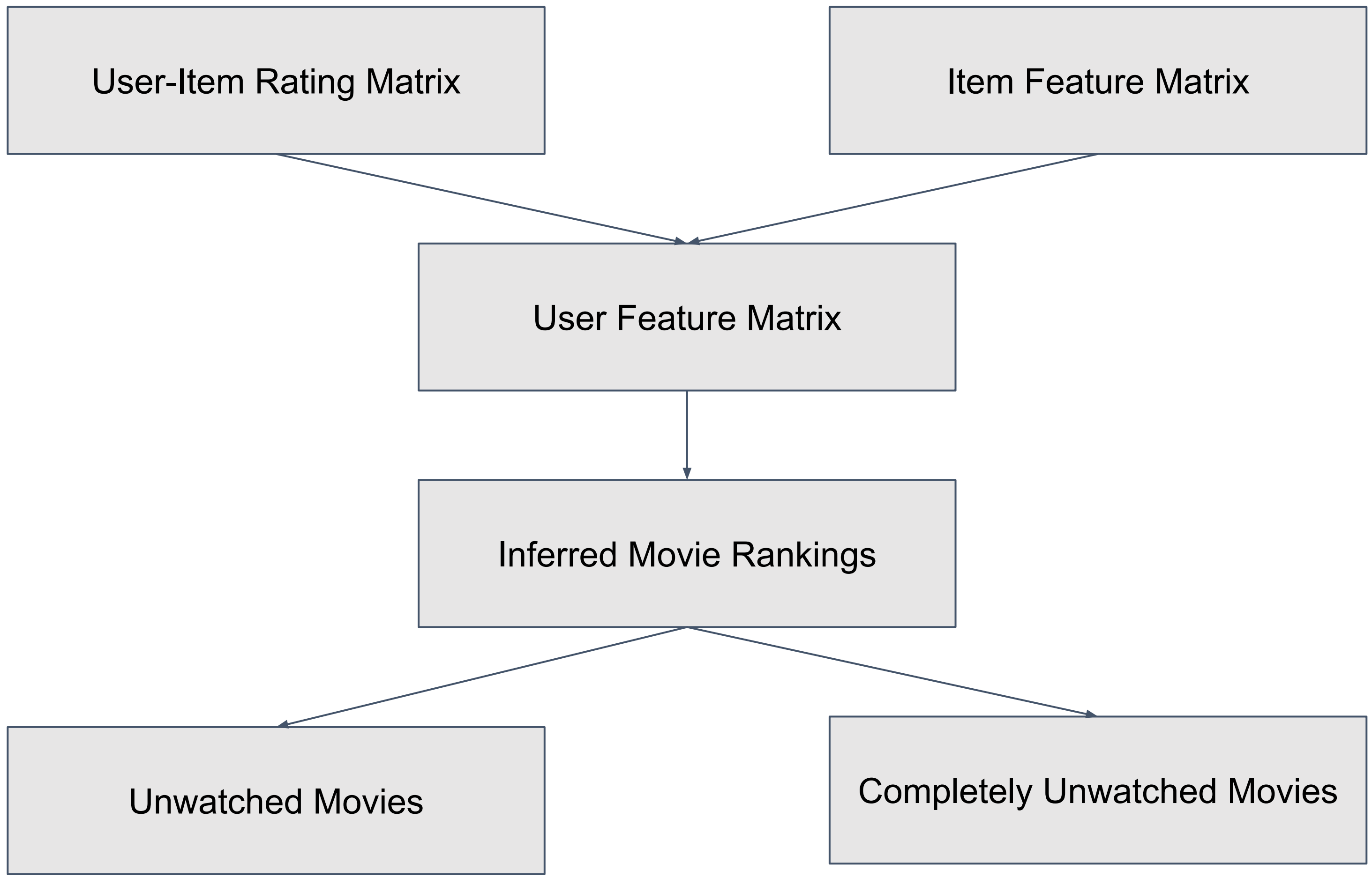
Movie	Action	Sci-Fi	Adventure	Comedy	Drama
Titanic	1	1	0	0	0
Martian	0	1	1	0	1
GOTG Vol 2	1	1	1	1	0

User	Titanic	Martian	GOTG Vol 2
User 1	0.428	0.648	0.802
User 2	0.56	0.5	0.9
User 3	0.473	0.5	0.73
User 4	0.179	0.769	0.589

Recommendation Order

- User 1 : GOTG Vol 2, Martian, Titanic
- User 2 : GOTG Vol 2, Titanic, Martian
- User 3 : GOTG Vol 2, Martian, Titanic
- User 4 : GOTG Vol 2, Martian, Titanic

Content Based Filtering Summary



Purwadhika
Startup and Coding School

Advantages and Disadvantages

Advantages:

- you can still make recommendation even if you don’t have specific information about each user
- capture specific interest of a user/user personalization

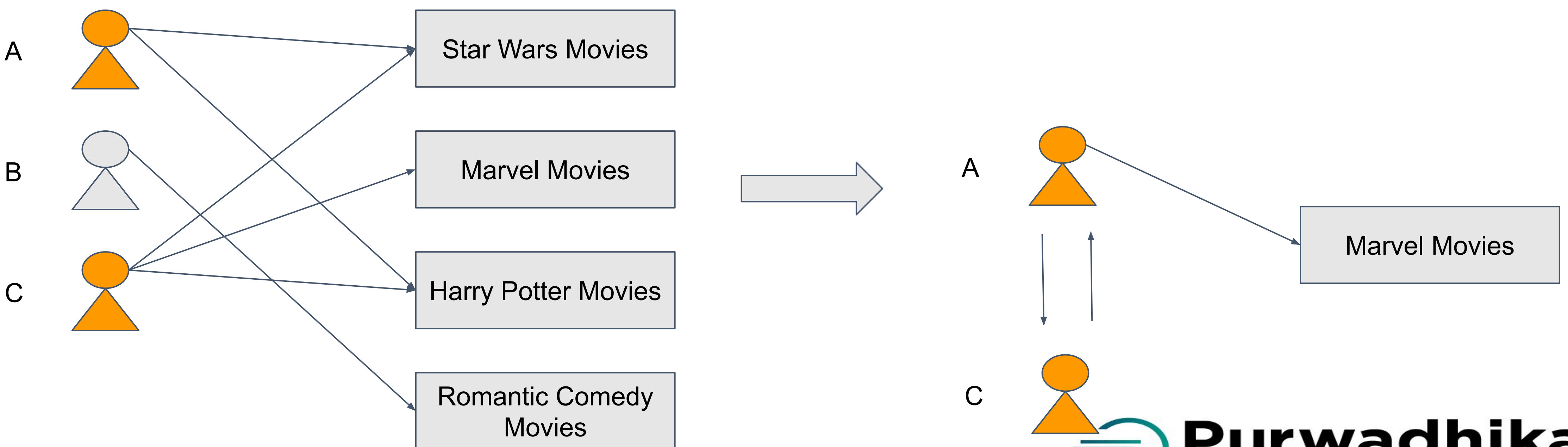
Disadvantages:

- has limited ability to expand user’s interest
- require specific domain knowledge to choose which content to use

Collaborative Filtering

Collaborative Filtering

- Use similar user's preference to recommend next items to a user
- User A and User C have similar preferences but user A haven't watch marvel movies yet
- Recommendation to A: viewers like you also views "Marvel Movies"
- We don't know the best factor to compare with



User-Item Rating Matrix

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8		6	5
User 3	9			10
User 4		7		9

User-item rating matrix (usually sparse matrix)

Rating can be :

- implicit : watch duration (inferring user interest)
- explicit : like/dislike

How To Measure User's Preferences ?

Implicit data collection examples:

- Observing the items that a user views in an online store
- Analyzing item/user viewing times
- Keeping a record of the items that a user purchases online

Explicit data collection examples:

- asking a user to rate
- asking a user to rank a collection of items from most favorite to least favorite

Purwadhika
Startup and Coding School

Steps in Collaborative Filtering

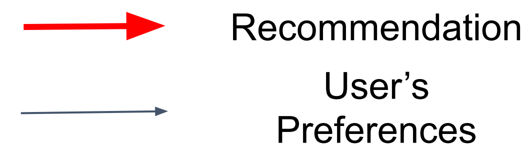
- How to determine which user or items are similar to one another ?
- How to determine the rating that a user would give to an item based on the ratings of similar user ?
- How to measure the accuracy of the ratings you calculate ?

Collaborative Filtering Method

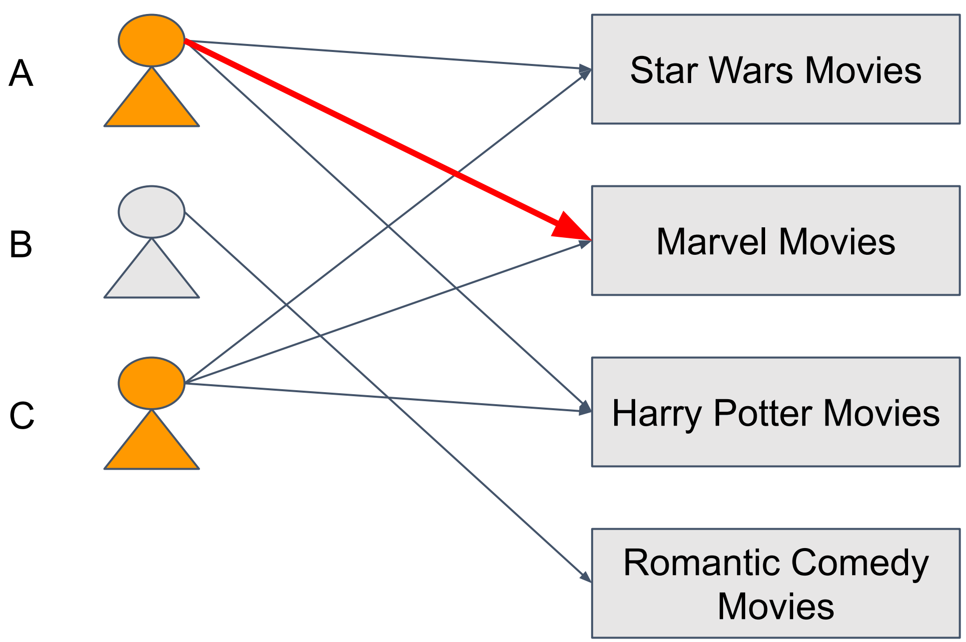
- Memory based
  - user based
  - item based
- Model based : Dimensionality Reduction
  - SVD
  - ALS
  - WLS

Memory Based

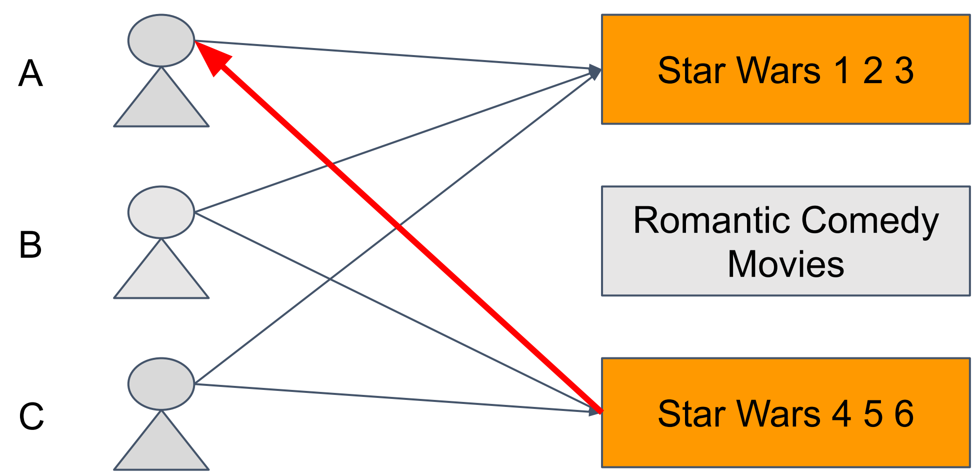
Memory Based Collaborative Filtering



user based



item based



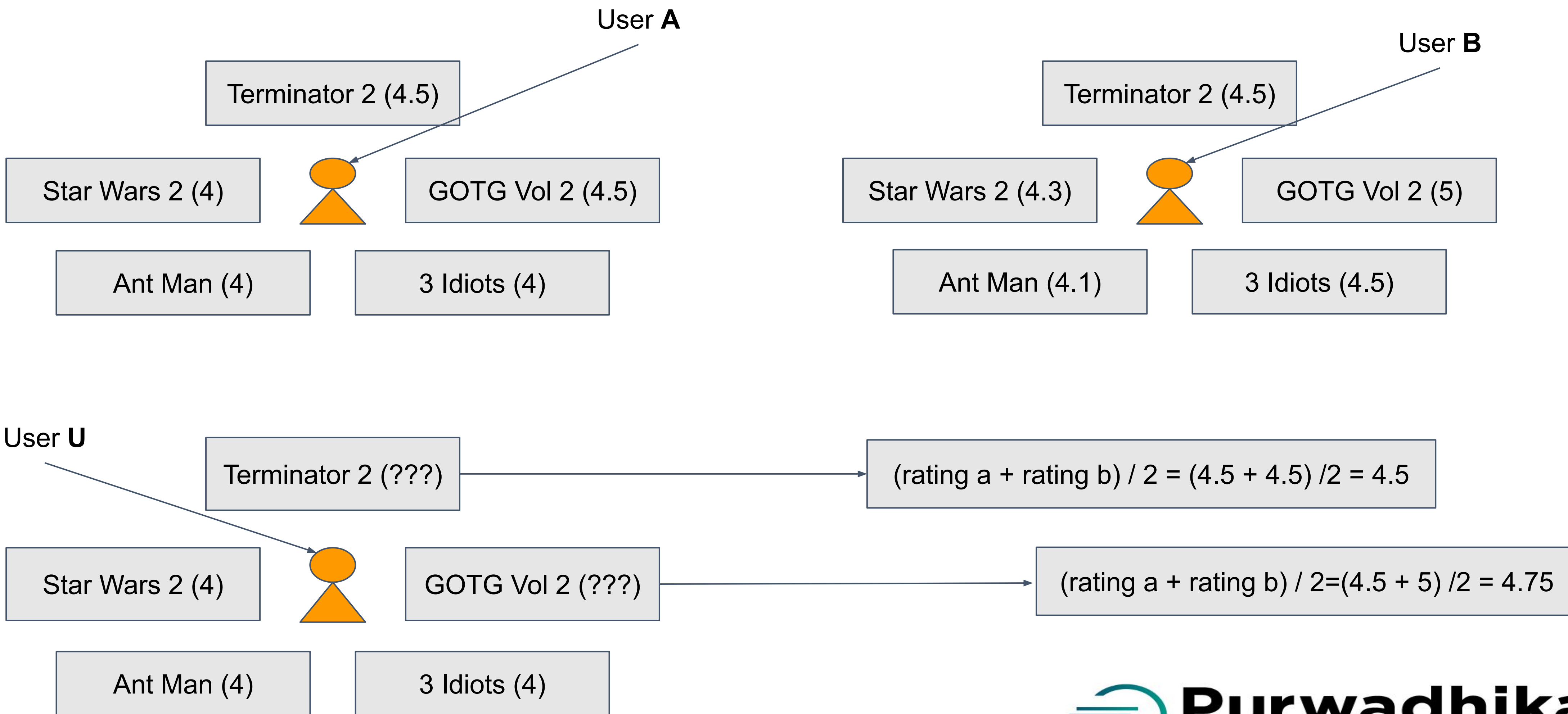
Purwadhika
Startup and Coding School

User Based Collaborative Filtering

How to recommend ?

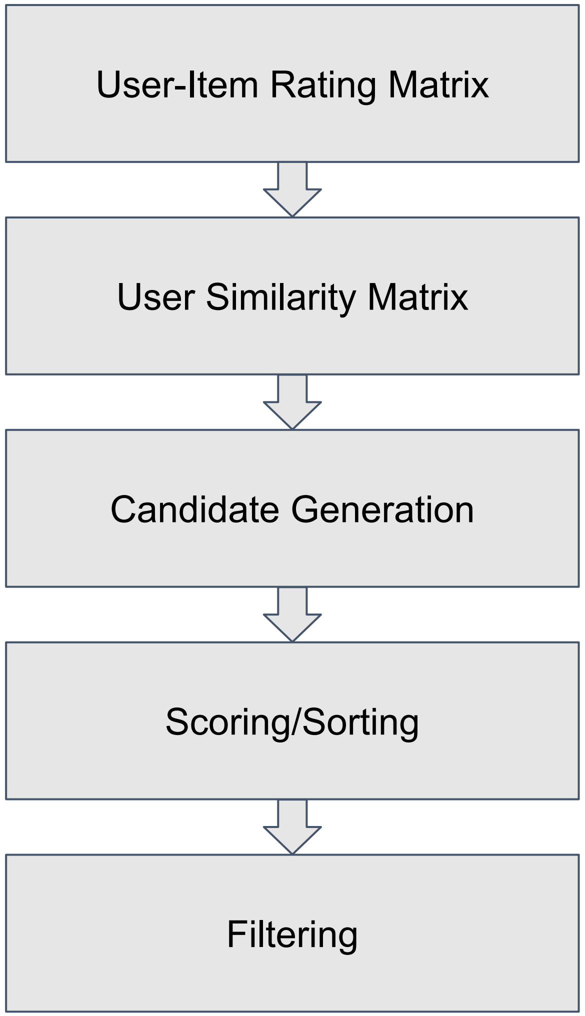
- let’s say we have a user called user **U**
- find other users **U’** that have similar preference with user **U**
- estimate rating for items **M** that user **U** hasn’t used but another user has
- use rating as recommendation order and filter

User Based Collaborative Filtering



Purwadhika
Startup and Coding School

User-based Collaborative Filtering



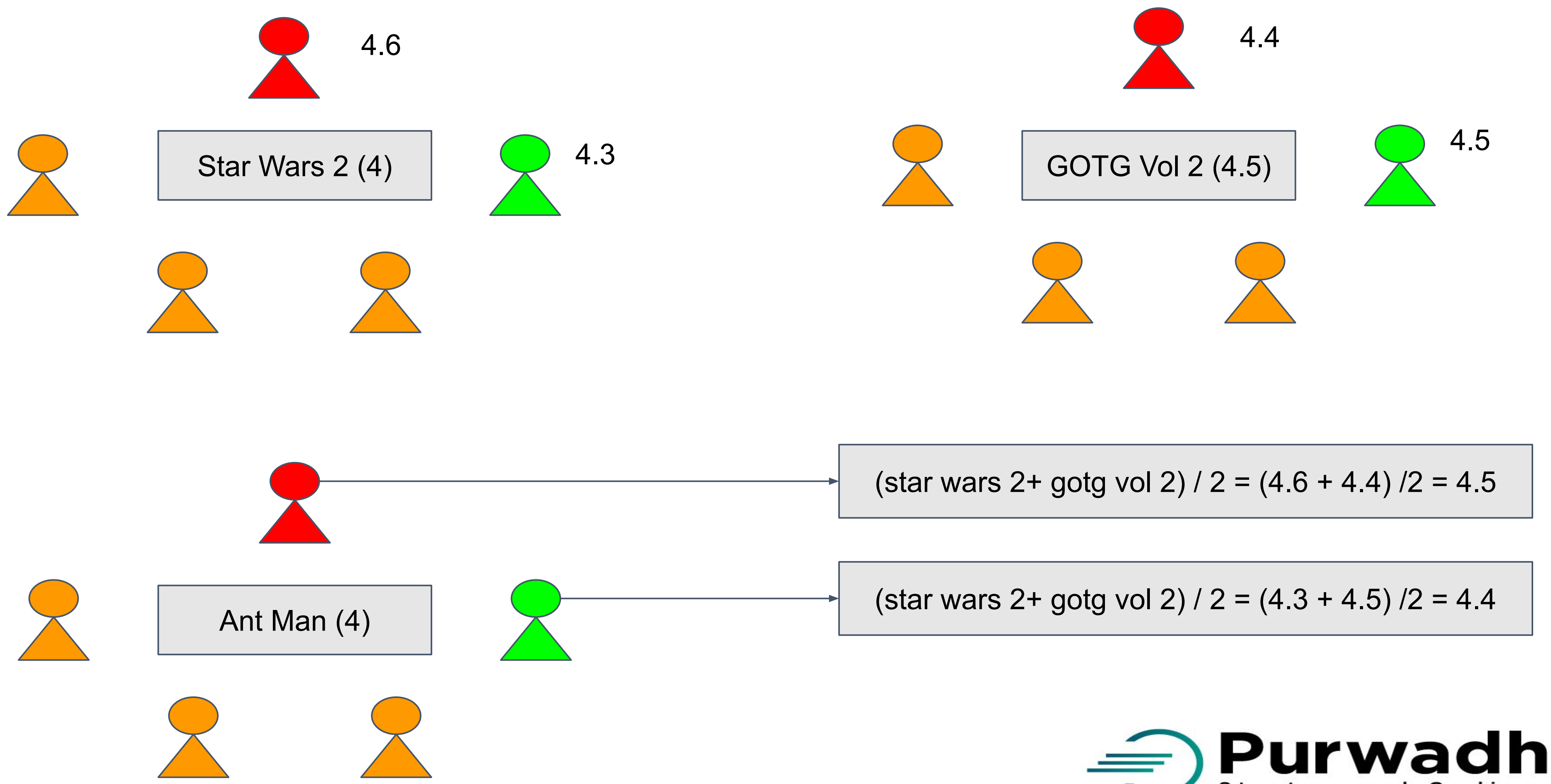
Item Based Collaborative Filtering

How to recommend ?

- find the most similar items (in term of user) for each items
- let’s say we have an item **M**
- find other items **M’** that similarly prefrenced like **M**
- find users (**U’**) of **M’** that has not used **M**
- estimate rating that will be given by **U’** to **M**
- use rating as recommendation order and filter

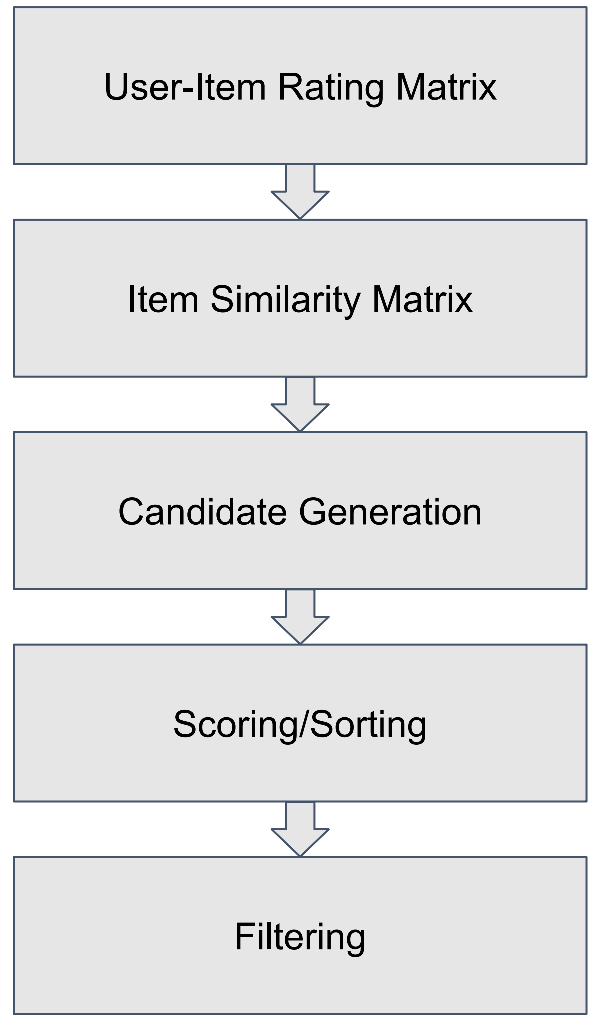
Purwadhika
Startup and Coding School

Item Based Collaborative Filtering



Purwadhika
Startup and Coding School

Item-based Collaborative Filtering



Purwadhika
Startup and Coding School

Model Based

Model Based

Main Idea : dimensional reduction

- users dimension
- items dimension

Matrix Factorization Method :

- SVD
- ALS
- WALS

Matrix Factorization

Dimension : Latent Factor x Items = 2 x 4

	1		
	-3		

Latent Factor

10	1

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8		6	5
User 3	9			10
User 4		7		9
User 5	9	9		7

Dimension : User x Latent Factor = 5 x 2

Dimension : User x Movie = 5 x 4

Matrix factorization can be seen as breaking down a large matrix into a product of smaller ones.

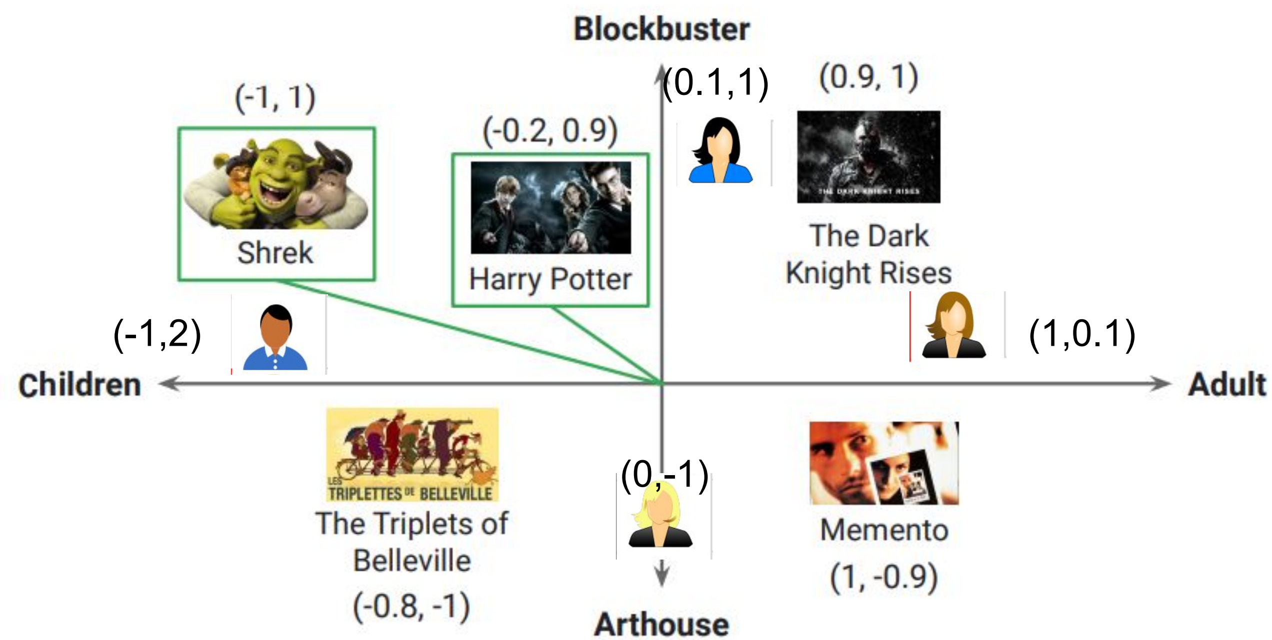
- Matrix with 5 x 4 dimension
- reduce into two matrix : user factor with 5 x 2 dimension and movie factor 2 x 4 dimension
- latent factor reveal hidden information. it can be anything
- 10 * 1 + 1 * (-3) = 7

Purwadhika
Startup and Coding School

Analogy

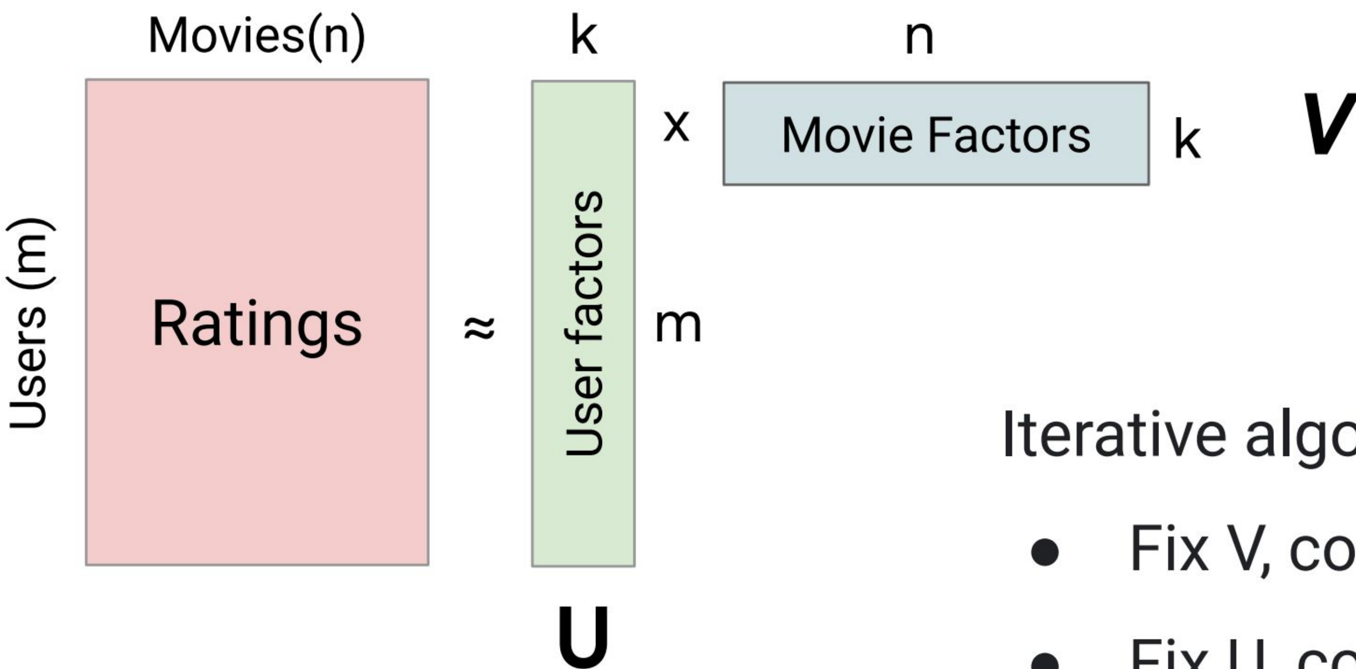
Latent Movie Factor : Interpreted as Blockbuster - Arthouse

Latent Movie Factor : Interpreted as Adult - Children



Purwadhika
Startup and Coding School

Matrix Factorization



Iterative algorithm:

- Fix V, compute U
- Fix U, compute V

Method :

- Singular Value Decomposition (SVD)
- Alternating Least Square (ALS)
- Weighted Alternating Least Square (WALS)

Singular Value Decomposition (SVD)

Explicitly sets all unobserved user-interaction matrix pairs to zero

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8	0	6	5
User 3	9	0	0	10
User 4	0	7	0	9
User 5	9	9	0	7

Purwadhika
Startup and Coding School

Alternating Least Square (ALS)

Simply ignores all unobserved user-interaction matrix pairs.

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8		6	5
User 3	9			10
User 4		7		9
User 5	9	9		7

Purwadhika
Startup and Coding School

Weighted Alternating Least Square (WALS)

Assign weights to all unobserved user-interaction matrix pairs. Weights can be thought as representing low confidence

User	Terminator 2	Interstellar	Ant Man 2	3 Idiots
User 1	7	9	8	9
User 2	8	w	6	5
User 3	9	w	w	10
User 4	w	7	w	9
User 5	9	9	w	7

Purwadhika
Startup and Coding School

Advantages and Disadvantages

Advantages:

- no domain knowledge needed
- can help to discover new interest

Disadvantages:

- cold start for new items and new user because until someone rates them, they don’t get recommended
- data sparsity
- scalability (computation can be growing due to large number of user and large number of items)

Purwadhika
Startup and Coding School

References

https://medium.com/data-folks-indonesia/recommendation-system-dengan-python-definisi-part-1-71154dc3f700

https://towardsdatascience.com/introduction-to-two-approaches-of-content-based-recommendation-system-fc797460c18c

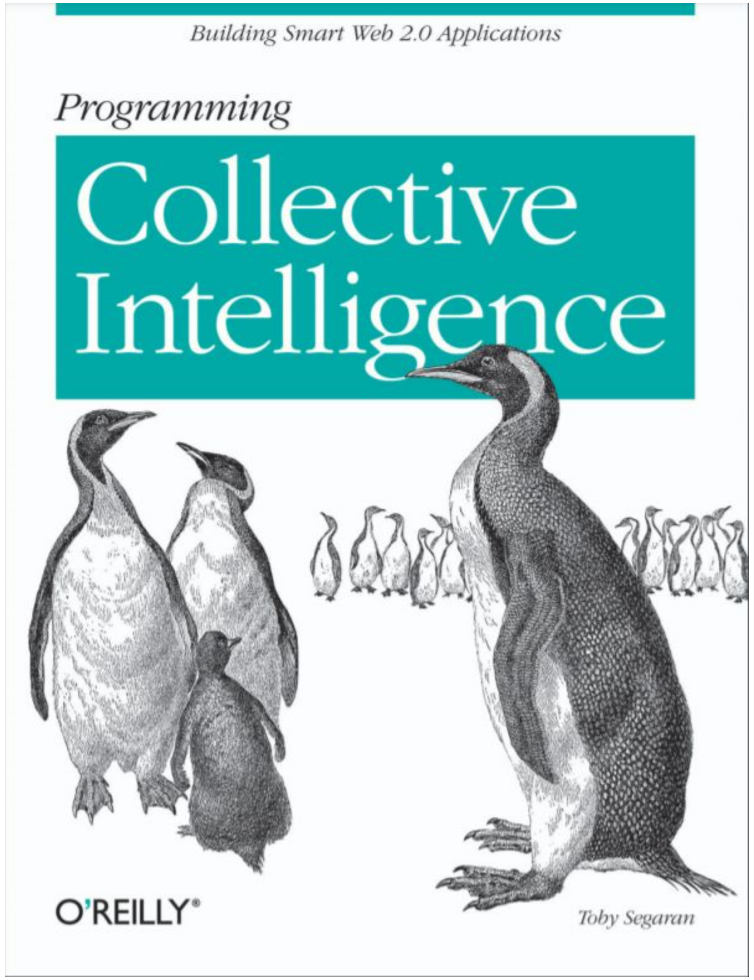
https://developers.google.com/machine-learning/recommendation/collaborative/basics

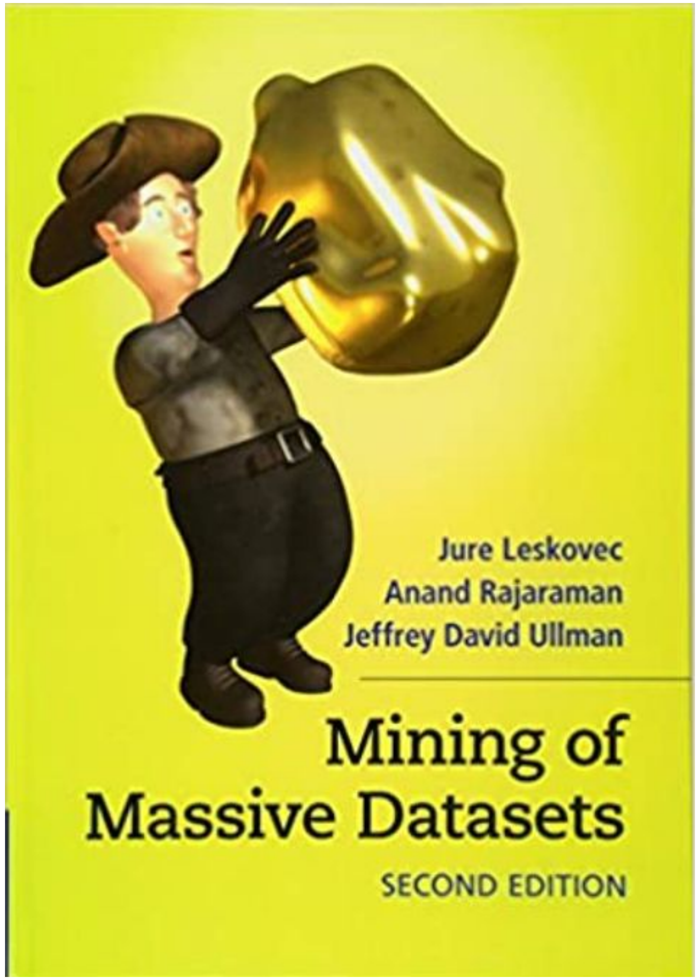
https://developers.google.com/machine-learning/recommendation/collaborative/matrix

https://realpython.com/build-recommendation-engine-collaborative-filtering/#:~:text=Collaborative%20filtering%20is%20a%20family,type%20of%20collaborative%20filtering%20approach.

Purwadhika
Startup and Coding School

References





Purwadhika
Startup and Coding School