The background of the slide features a large, irregularly shaped circle filled with a dark blue color. This circle is set against a lighter blue background that has white, textured splatters and speckles, resembling paint or liquid droplets. The overall effect is artistic and modern.

# An Introduction to Recommender System

Poo Kuan Hoong

# Agenda

- Introduction
  - Why do we need recommender system?
  - What is recommendation and personalization?
- Interactions
  - User/Items
- Recommender System Approaches
  - Content Based
  - Collaborative Filtering
    - Nearest Neighbor
    - Matrix Factorization
- Performance Metrics
- Simple Demo

# Introduction

- Recommender System is widely used to provide relevant recommendations to users
- Websites such as Amazon, YouTube, Netflix, eBay and many more use recommendation engine



NETFLIX

# Why do we need recommender system?

## Relevant recommendation

- Most people do not know what is available and what they are looking for

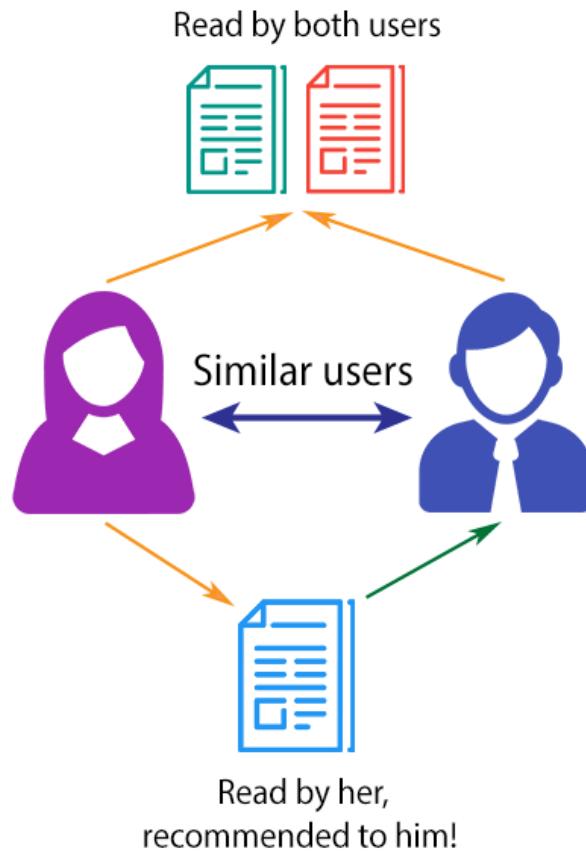


## The Long Tail Phenomenon

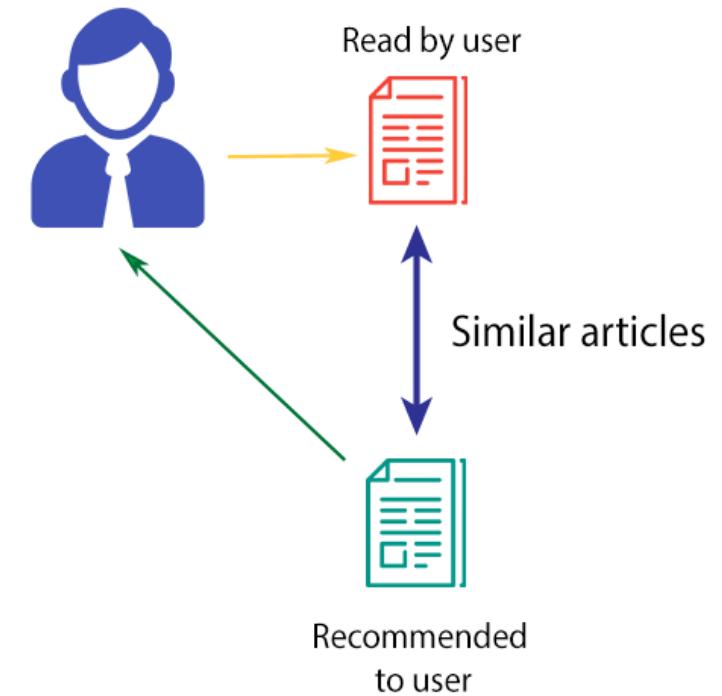
## Recommendation

- A recommendation system presents **items** to **users** in a relevant way
- The definition of **relevant** is product/context-specific

### COLLABORATIVE FILTERING



### CONTENT-BASED FILTERING



## Your recently viewed items and featured recommendations

Inspired by your browsing history

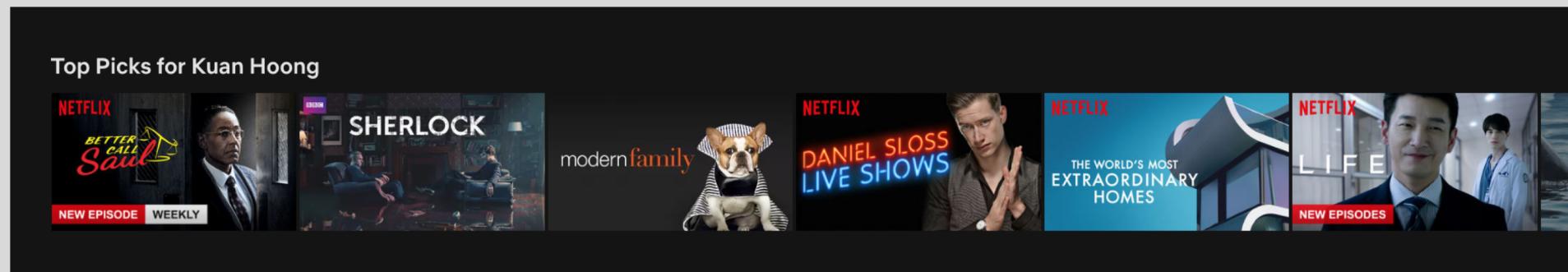
Page 1 of 7



Related to items you viewed



Amazon



Netflix

# Personalization

- A **personalization** system presents recommendations in a way that is **relevant** to the **individual user**
- The **user** expects their experience to change based on their **interactions** with the system
- **Relevance** can be in the context or product specific



# Browse

OVERVIEW PODCASTS CHARTS GENRES & MOODS NEW RELEASES DISCOVER CONCERTS

## Playlists made just for you



### Discover Weekly

Your weekly mixtape of fresh music. Enjoy new discoveries and deep cuts chosen just for you. Updated every Monday, so save your...

PLAYLIST • BY SPOTIFY



### Release Radar

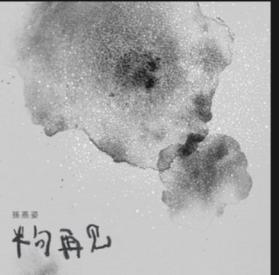
Never miss a new release! Catch all the latest music from artists you care about, plus new singles picked just for you. Updates every...

PLAYLIST • BY SPOTIFY

## Top recommendations for you



Soar on Wings Like an Eagle  
Alina Nin



半句再見  
Stefanie Sun



五月天自傳  
Mayday



長大  
The Freshman

Kuan Hoong

MADE FOR KUAN

## Discover Weekly

Your weekly mixtape of fresh music. Enjoy new discoveries and deep cuts chosen just for you. Updated every Monday, so save your favorites!

Made for Kuan Hoong by Spotify • 30 songs, 1 hr 52 min

PLAY FOLLOWING ...

Filter

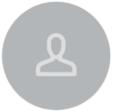
TITLE	ARTIST	ALBUM	DATE
+ Break My Heart (feat. Thomas Daniel)	Mattilo, Thomas Daniel	Break My Heart (feat. Thomas Da...	5 days ago
+ This is love	Eric Chou	愛,教會我們的事	5 days ago
+ 莎賓娜,不要再結婚了!	Joanna Wang	摩登悲劇	5 days ago
+ Sometimes When We Touch	Lion	Replay	5 days ago
+ 當你	Cyndi Wang	Begin	5 days ago
+ Summertime Love	Vanness Wu	Summertime Love	5 days ago
+ Proud of You	Emi Fujita	Camomile Best Audio	5 days ago
+ LOVE SCENARIO	iKON	Return	5 days ago
+ When I Saw You	Bumkey	A Korean Odyssey (Original Telev...	5 days ago
+ DDU-DU DDU-DU - KR Ver.	BLACKPINK	SQUARE UP	5 days ago
+ 心中的日月 - Live	Leehom Wang, 韓紅	我想和你唱3 Episode 3	5 days ago
+ 星火(電影《鬥魚》主題曲)	Lydia	星火(電影《鬥魚》主題曲)	5 days ago

# Personalization

## People you may know



**Justin Lim**  
Android Developer at Gobike  
Philippe Bertrand and 20 others

[Connect](#)

**Insa Lohmann**  
insa\_lohmann@web.de

[Invite](#)

**Ismail Ibrahim**  
Geo Sense, Making Sense to your Geo Data  
Kin Peng Chan and 13 others

[Connect](#)

**Ban Weng**  
kwban@mmu.edu.my

[Invite](#)

**Thirusagthy**  
Operation Specialist at NCS Group  
khairul anwar and 1 other

[Connect](#)

**Mahathir Malek**  
mahathirmalek@gmail.com

[Invite](#)

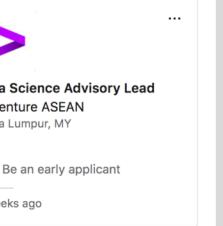
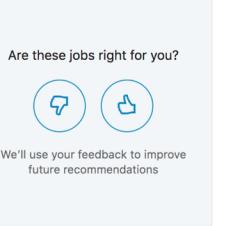
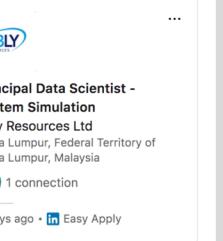
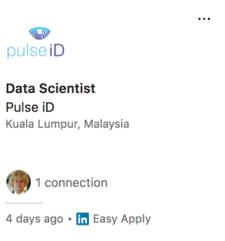
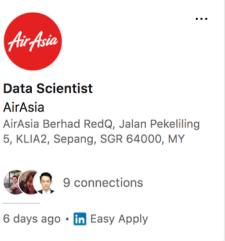
### Based on your Profile and Career interests

3 job titles · Any location · Any industry · 1 to 10,000+ employees ... [Update Career interests](#)



**Muhammad J**  
CTO at BYT Tech  
Data Scientist  
AirAsia  
AirAsia Berhad RedQ, Jalan Pekeling 5, KLIA2, Sepang, SGR 64000, MY

9 connections  
6 days ago • [Easy Apply](#)



## People you may know



**Daren Tan**

Multimedia University  
Chee Siang Wong and 2 other mutual friends

[Add Friend](#)[Remove](#)

**Jared Chan**

Adelaide, South Australia  
Kai Ping Yong and 13 other mutual friends

[Add Friend](#)[Remove](#)

**Clarence Lcy (梁涇瑜)**

Penang, Malaysia  
Chen Ruyue and 29 other mutual friends

[Add Friend](#)[Remove](#)

**Jing Mun**

SMK Kepong  
Henry Tong and 11 other mutual friends

[Add Friend](#)[Remove](#)

**Su-Lynn Tan**

Ian Tan and Lee Tang Ching are mutual friends.

[Add Friend](#)[Remove](#)

**Keith Hiew**

Kuala Lumpur, Malaysia  
Pablo Pablo and 2 other mutual friends

[Add Friend](#)[Remove](#)

# LinkedIn

# Facebook

# Users vs Items

## Users

- A **user** in a recommender system is the party that is **receiving and acting on** the recommendations.
- Sometimes the user is the **context**, not an actual person.

## Items

- An **item** in a recommender system is the **passive party** that is being recommended to the users.

The line between these two can be blurry.

# Quiz

- What is the matching user vs items for the followings?
  - LinkedIn      (Users: Members, Items: Jobs)
  - Facebook      (Users: Members, Items: Friends)
  - Amazon      (Users: Members, Items: Books/Products)
  - Netflix      (Users: Members, Items: Movies/Shows)
  - Spotify      (Users: Members, Items: Songs)



# Interactions

## Positive

- Thumb up, hearts, stars, listens, watches, follows, bids, purchases, reads, views, upvotes...



## Negative

- Thumb down, skips, angry-face, 0-star review, unfollows, rejections, returns, downvotes..



## Explicit vs Implicit

- Explicit actions are those that a user expects or intends to impact their personalized experience
- Implicit actions are all other interactions between users and items

# Recommender System (RS) Approaches

- **Content-Based (CB)** – use personal preferences to match and filter items
  - E.g. what sort of books do I like?
- **Collaborative Filtering (CF)** – match ‘like-minded’ people
  - E.g. if two people have similar ‘taste’ they can recommend items to each other
- **Social Software** – the recommendation process is supported but not automated
  - E.g. Weblogs provide a medium for recommendation
- **Social Data Mining** – Mine log data of social activity to learn group preferences
  - E.g. web usage mining
- For this talk, we will concentrate on CB and CF

# Content-Based

- Find me things that I liked in the past.
- Machine *learns* preferences through user feedback and builds a user *profile*
- Explicit feedback – user rates items
- Implicit feedback – system records user activity
  - Clickstream data classified according to page category and activity, e.g. browsing a product page
  - Time spent on an activity such as browsing a page
- Recommendation is viewed as a search process, with the user profile acting as the query and the set of items acting as the documents to match.

# Example: Content Based



Animated	Yes	Yes	No	No
Superheroes	No	No	Yes	Yes
Super Villains	Yes	No	Yes	Yes
Came out on 2015	Yes	Yes	No	No
Marvel	No	No	Yes	Yes

2

1

1

# Collaborative Filtering (CF)

- Match people with similar interests as a basis for recommendation.
- The most prominent approach to generate recommendations
  - used by large, commercial e-commerce sites
  - well-understood, various algorithms and variations exist
  - applicable in many domains (book, movies, DVDs, ..)
- Approach
  - use the "wisdom of the crowd" to recommend items
- Basic assumption and idea
  - Users give ratings to catalog items (implicitly or explicitly)
  - Customers who had similar tastes in the past, will have similar tastes in the future

# Example



John	Yes	Yes	Yes	Yes
Peter	No	No	Yes	Yes
Ian	Yes	No	Yes	No
Tom	No	Yes	No	No
Marcus	Yes	?	?	?

# Example: Collaborative Filtering (CD)



John	Yes	Yes	Yes	Yes
Peter	No	No	Yes	Yes
Ian	Yes	No	Yes	No
Tom	No	Yes	No	No
Marcus	Yes	?	?	?

1 vote

2 votes

1 vote

# Example: Collaborative Filtering (CD)



John	5	5	4	3
Peter	0	0	3	5
Ian	3	0	4	4
Tom	1	5	5	1
Marcus	?	?	?	?

	item1	item2	item3	item4
user1	5	5	4	3
user2	0	0	3	5
user3	3	0	4	4
user4	1	5	5	1
user5	?	?	?	?

# Collaborative Filtering (CF)

- 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

# User-based k-nearest neighbor collaborative filtering

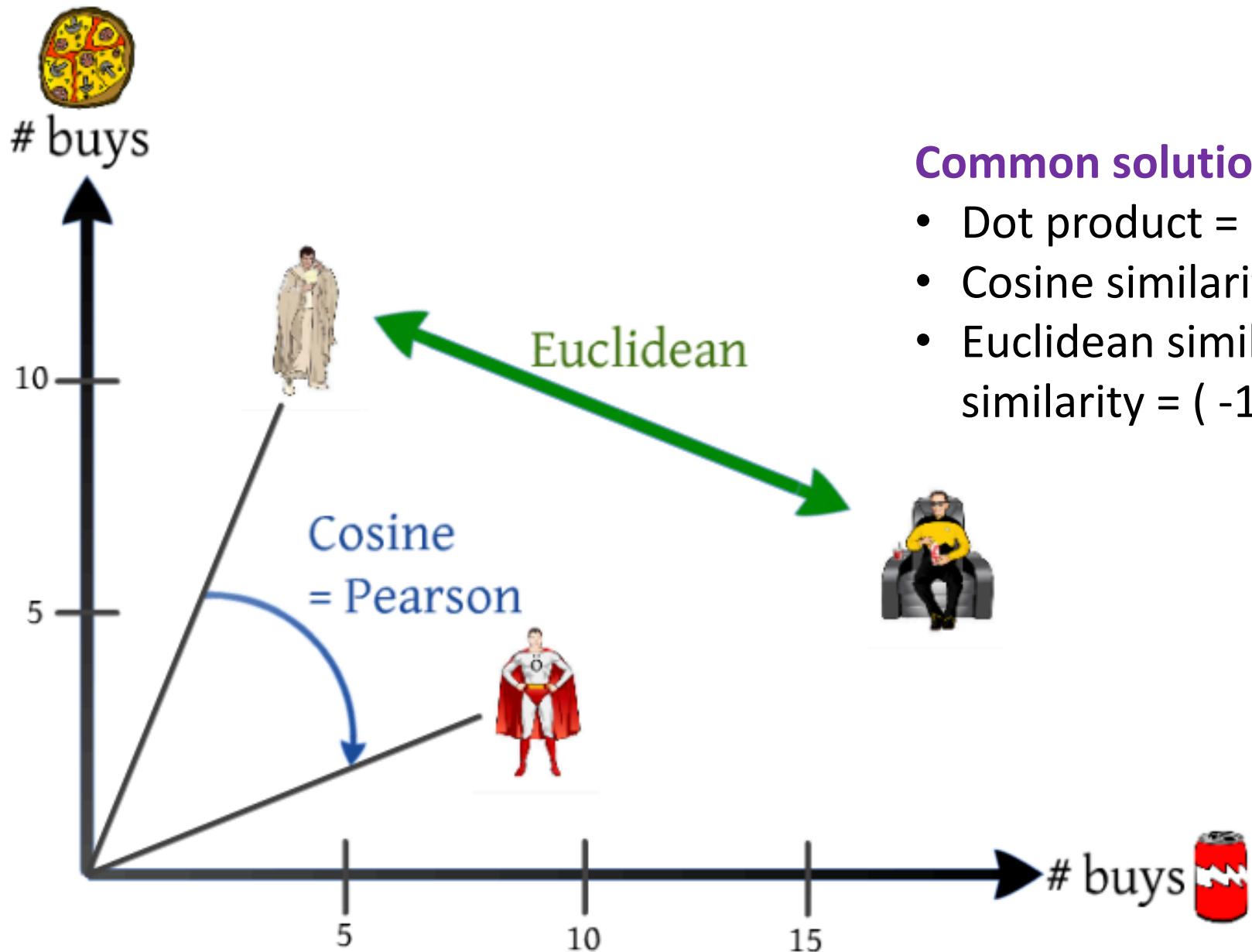
- The basic technique
  - Compute similarity of users
  - Find  $k$  most similar users to user  $a$
  - Recommend movies not seen by  $a$
- Basic assumption and idea
  - If users had similar tastes in the past they will have similar tastes in the future
  - User preferences remain stable and consistent over time



John	5	5	4	3
Peter	0	0	3	5
Ian	3	0	4	4
Tom	1	5	5	1
Marcus	?	?	?	?

Cosine Similarity

$$sim(a, b) = \frac{a \cdot b}{\|a\| \cdot \|b\|}$$

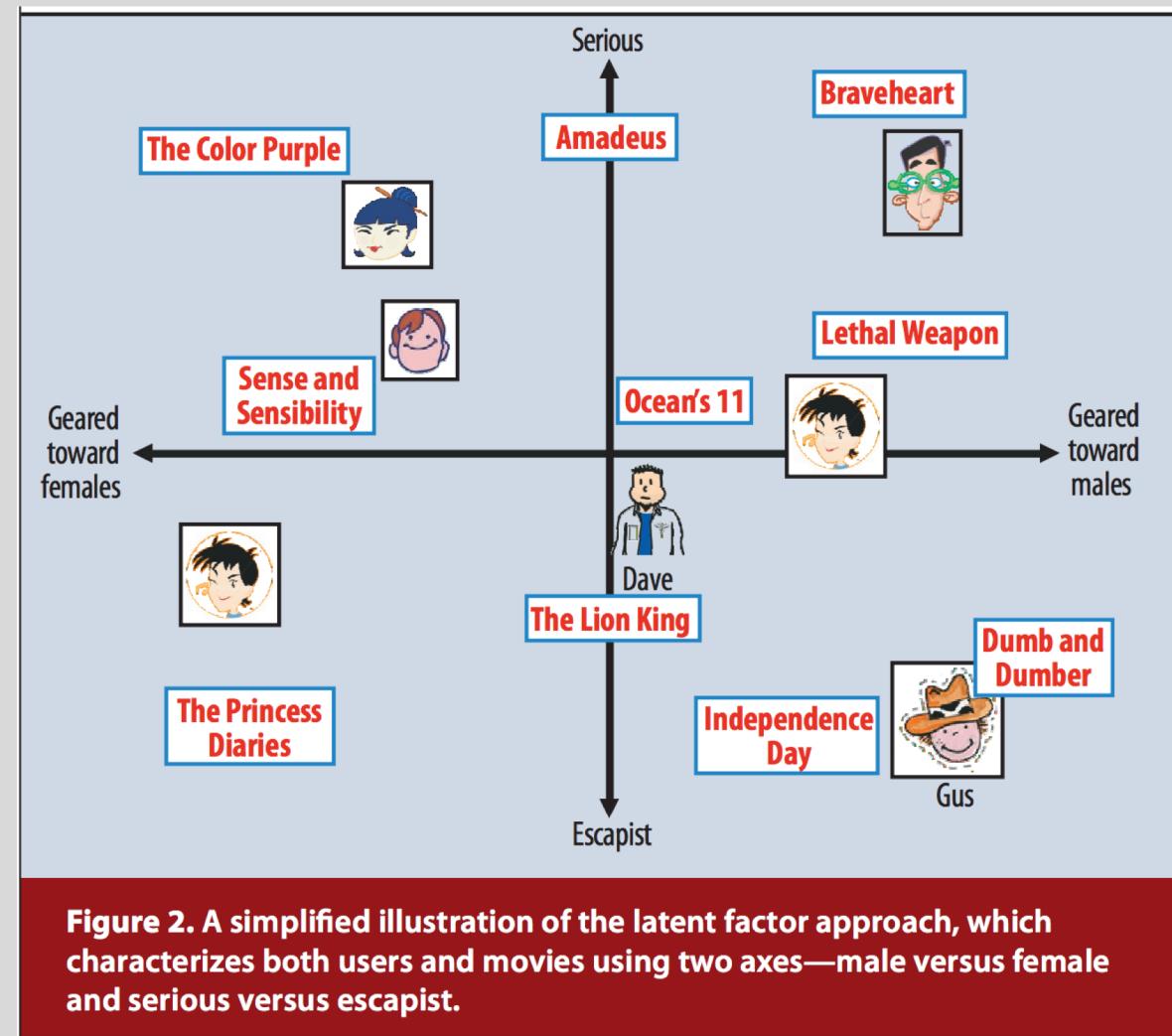


### Common solutions:

- Dot product =  $\text{User} \cdot \text{Item}$
- Cosine similarity =  $\cos(\Theta)$
- Euclidean similarity\* = ( -1 Manhattan similarity = ( -1 \*  $|\text{User} - \text{Item}|$  )

# Matrix Factorization – Collaborative Filtering

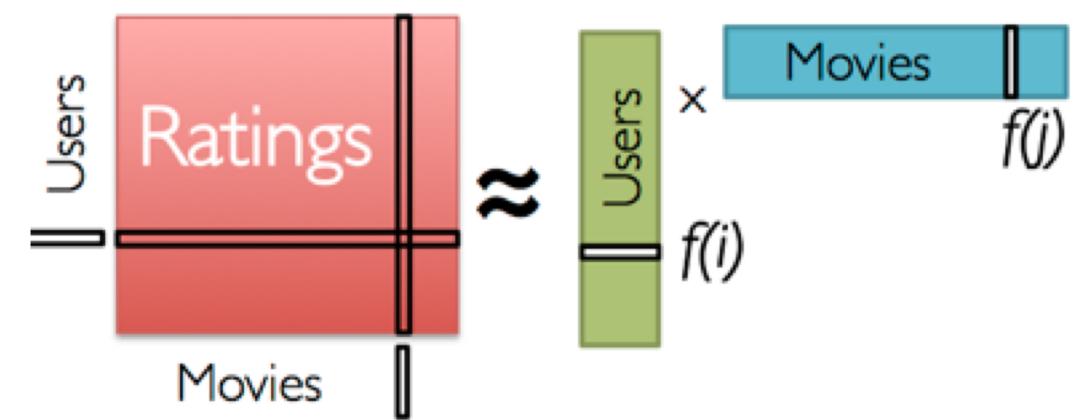
- Identify latent (hidden) features from the input user x itemRatings matrix to represent users and items as vectors in N dimensional space.
- Matrix factorization based methods attempt to reduce dimensionality of the interaction matrix and approximate it by two or more small matrices with k latent components.



# Matrix Factorization – Collaborative Filtering

- **Training:** Use Matrix factorization approaches (Eg. Singular value Decomposition or SVD) to split the Rating Matrix into constituent User Matrix and Item Matrix with minimum Sum of squared error (SSE).
- **Goal:** Predict unknown ratings for the remaining set of movies using the learned User Matrix and Item Matrix

	item1	item2	item3	item4
user1	5	5	4	3
user2	0	0	3	5
user3	3	0	4	4
user4	1	5	5	1
user5	?	?	?	?



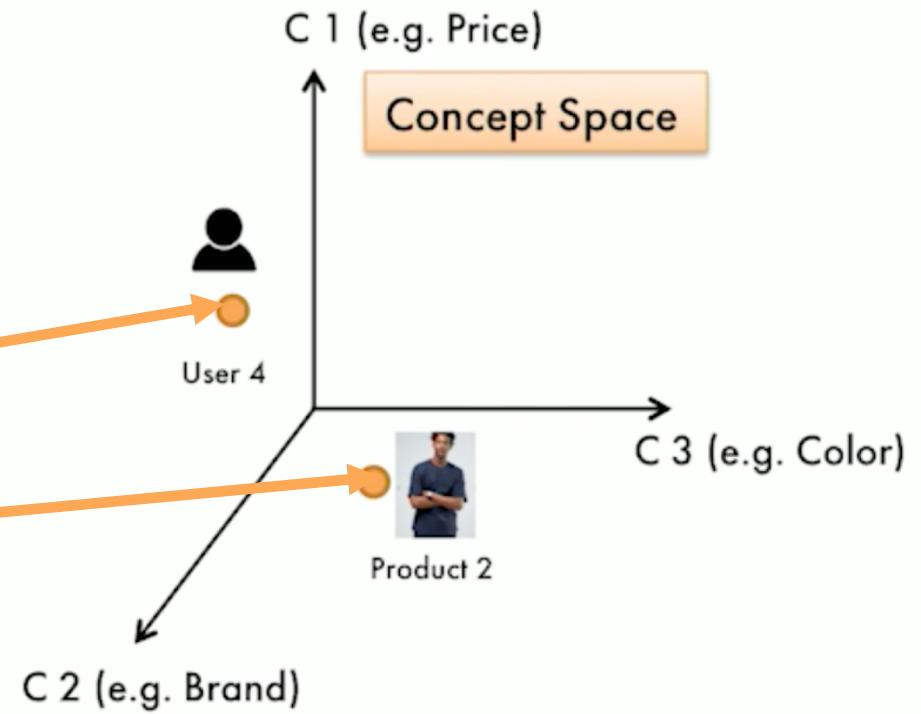
# MATRIX FACTORISATION

users	1	3	5	5	5	4		
items		5	4	4	2	1	3	
users	2	4	1	2	3	4	3	5
items	2	4	5		4		2	
users		4	3	4	2		2	5
items	1	3	3		2		4	

items	factors	.1	-.4	.2	
items		-.5	.6	.5	
items		-.2	.3	.5	
items		1.2	2.1	.3	
items		-.7	2.1	-2	
items		-1	.7	.3	

users	1.1	-2	.3	.5	-.2	-.5	8	-.4	.3	
users		.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7
users		2.1	-4	.6	1.7	2.4	.9	-.3	.4	.8



## Interactions

Identify interaction values

Types of behaviors

Explicit vs Implicit

Do you need negative interactions?

## Features

Identify user/item features

Indicator features

Use metadata

Feature Engineering

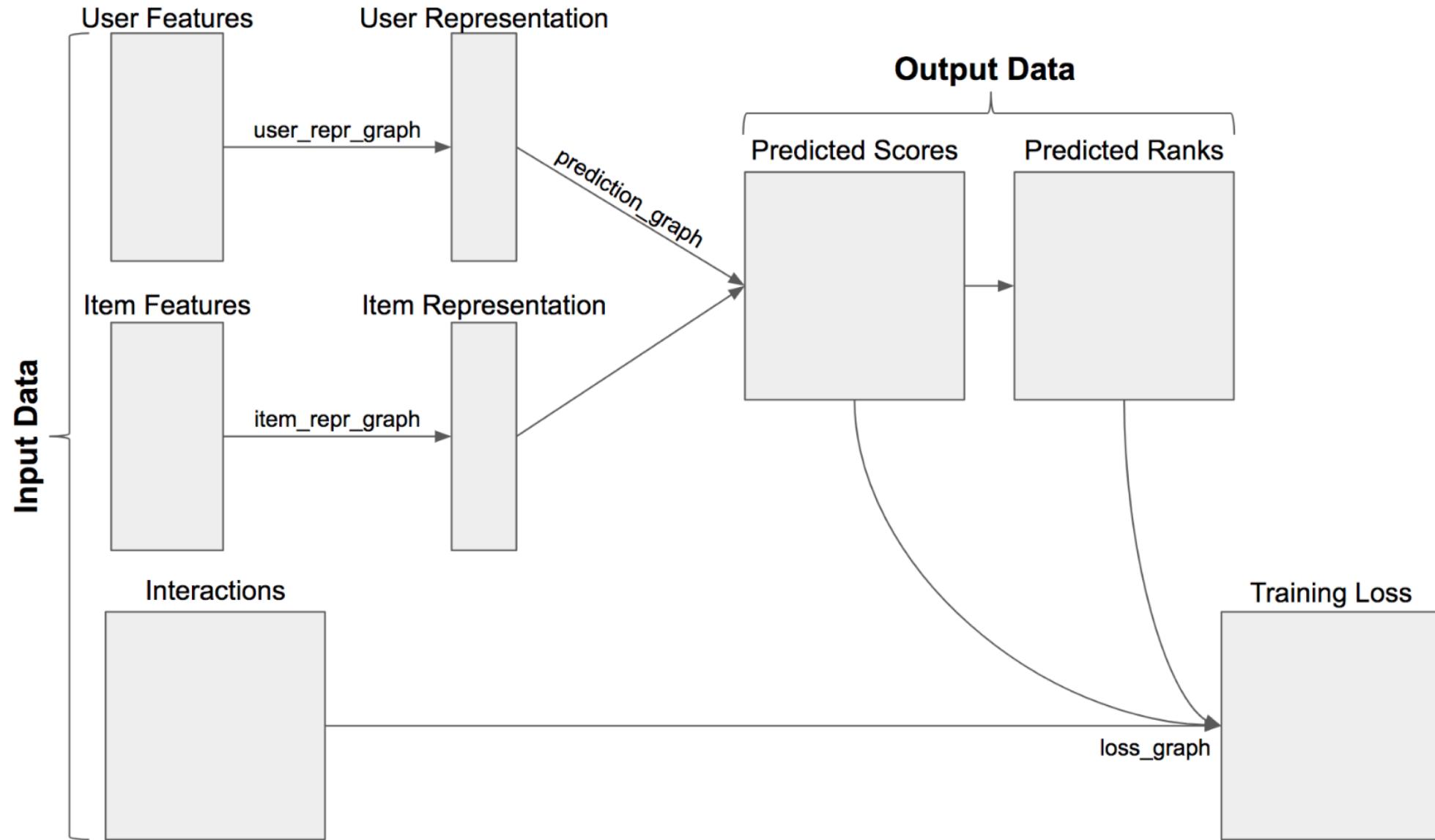
## Learning

Identify learning process

Which prediction method

Identify loss function

How do you evaluate?



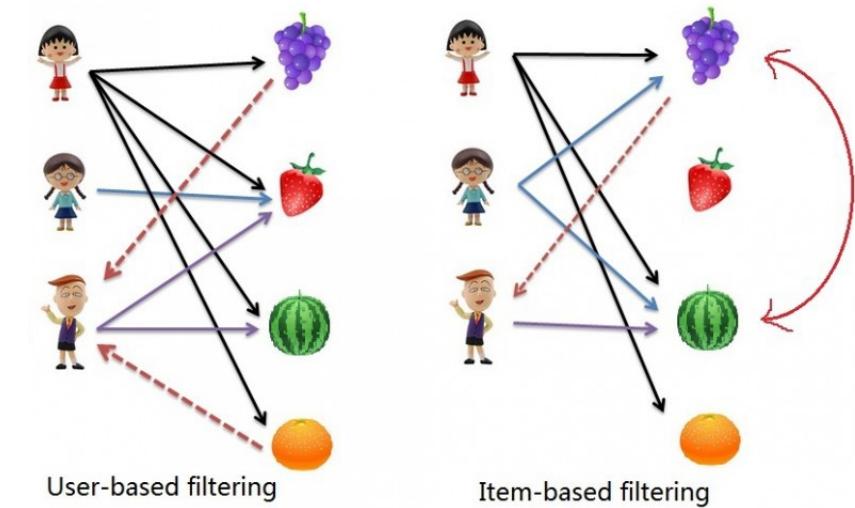
# Identify Interaction Values

- Do we allow negative interactions?
  - Negative interactions can be valuable statements of what content to avoid
  - Negative interactions can be confusing when learning-to-rank
  - Not all loss functions accommodate negative interactions



# Identify user/item features

- Indicator features allow for powerful personalization but are as numerous as user/item
- recommenders with user interactions can not effectively make recommendations for new users (cold-start problem)
- Many users means many indicator features – this may not scale
- Identify preprocessing processes to improve recommender learning
- Remove useless/misleading features
- The choice of representation function impacts the usefulness of feature engineering



# Identify learning process

- What is the effective ways for the recommender system to learn?
- Use of linear kernel or well engineered features (matrix factorization) are effective
- How about complicated models such as neural networks or deep learning?
- Can the learning process will represented without interactions such as auto-encoders, word2vec, etc
  - Identify loss functions – Root Mean Square Error (RMSE), Weighted approximately ranked pairwise (WARP), Weighted margin-rank batch (WMRB), etc
- What is the evaluation process?



# Performance Metrics

- What do we want to measure?
  - Clicks
  - Likes/Saves
  - Actions – Download, Install, etc

## Loss Function

- The process that **converts predictions and interactions** in to error for learning
- Common examples:
  - Root-mean-square error (RMSE)
  - Kullback-Leibler divergence (KLD)
  - Alternating least squares\* (ALS)
  - Bayesian personalized ranking\* (BPR)
  - Weighted approximately ranked pairwise (WARP)
  - Weighted margin-rank batch (WMRB)



# Other considerations

- Balancing Personalization vs Other Objectives
- Balancing Popularity vs Novelty
- Temporal effects
- Cold start problems

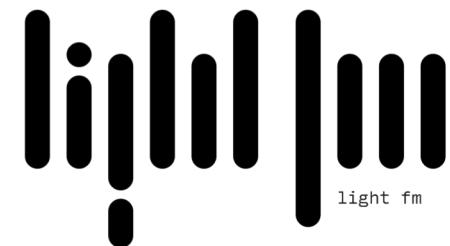
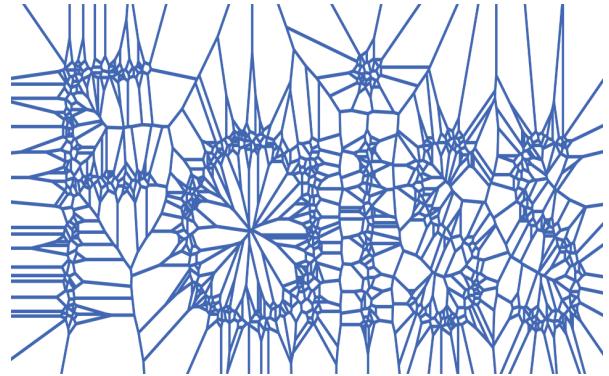


# Libraries

- [Python Scikit-Learn – SVD, PCA etc](#)
- [LightFM](#)
- [TensorRec](#)
- [FAISS \(Facebook AI Similarity Search\)](#)
- [NMSLib \(Non-Metric Space Library\)](#)

# Tutorial

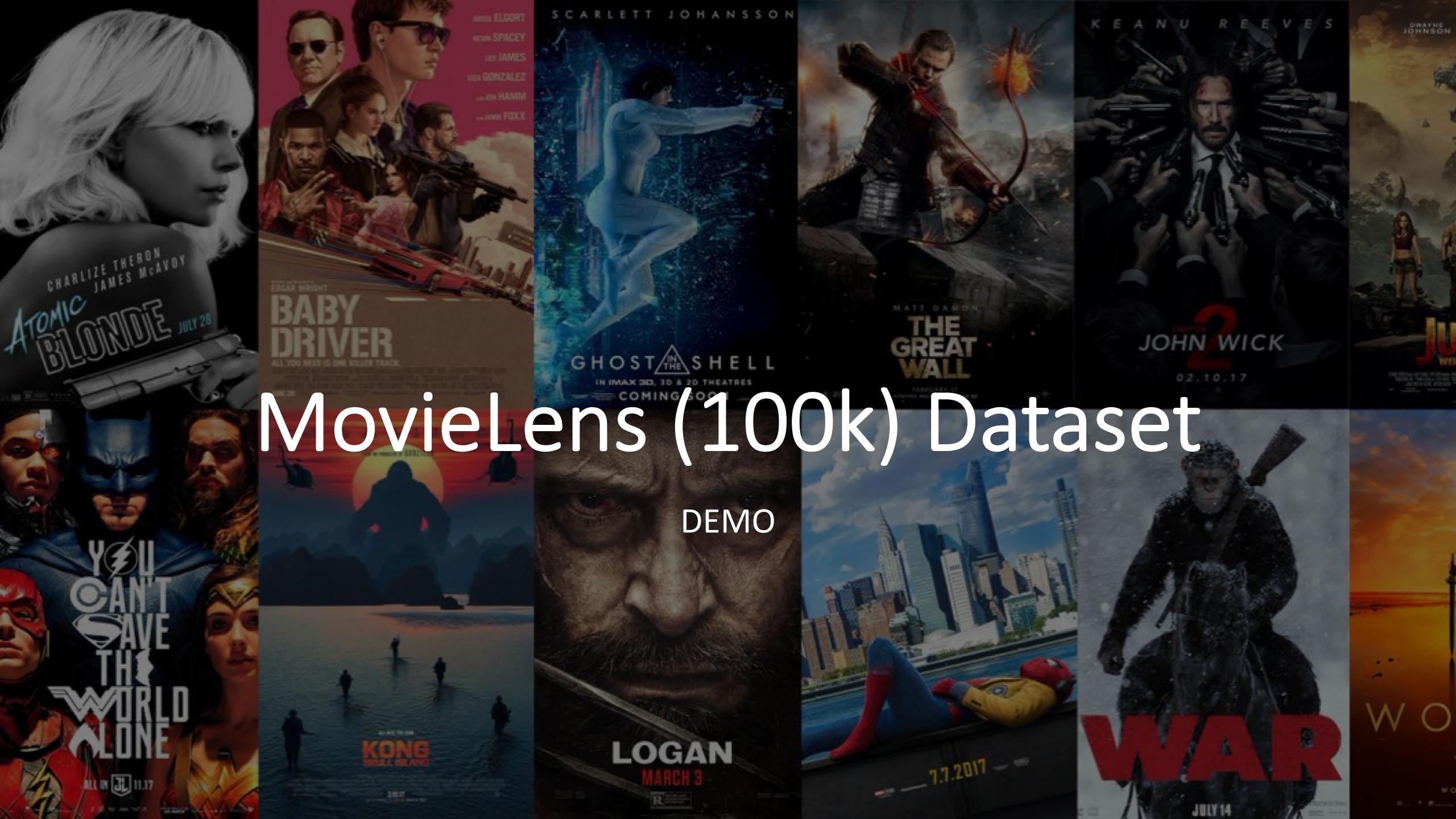
- [Building a Recommendation System in TensorFlow: Overview](#)



# Q&A

---





# MovieLens (100k) Dataset

DEMO



[http://bit.ly/TFDLM recommender](http://bit.ly/TFDLM_recommender)

Github

# Thanks!

## Questions?



@kuanhoong



<https://www.linkedin.com/in/kuanhoong>



kuanhoong@gmail.com



# Group Photo