# Special Topic: Recommender Systems

Dong-Kyu Chae

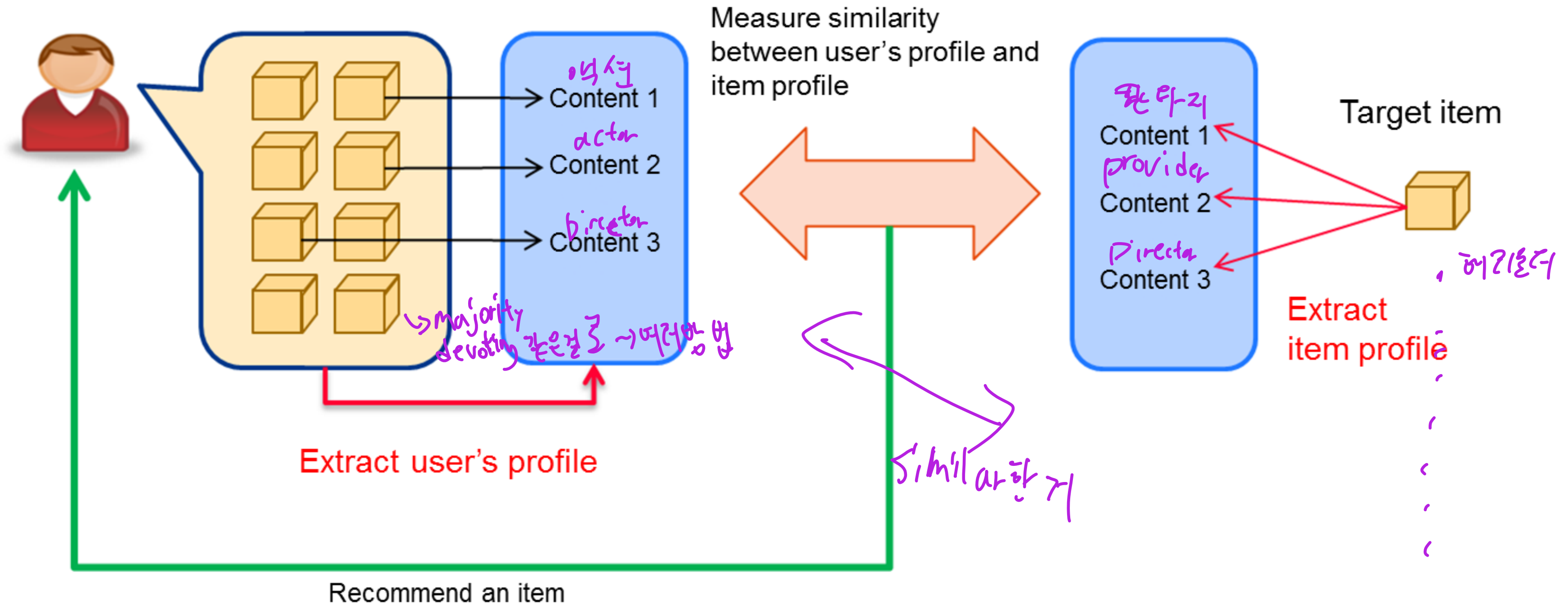
PI of the Data Intelligence Lab @HYU
Department of Computer Science & Data Science
Hanyang University

Data
Intelligence
LAB

# Content-Based Approach

- **Recommend such items that have contents similar to the profile of the target user**
  - Contents: characteristics of an item
    - Example : **genre**, **actor**, **director in movies, etc…**
  - Profile: the summarized item contents included in the target user's history

Target user's item usage history



# Content-Based Approach

- **Similarity**
  - Cosine similarity, Pearson correlation, etc.
- **How to define a user profile?**

  random, majority, devoting ...
  average ...

  - Using a single, unified user profile ← select the best item
  - Define each purchased item as a profile, and then aggregate the similarity between the target item and each purchased item 각각 다 비교해보기
- **Example of cosine similarity**

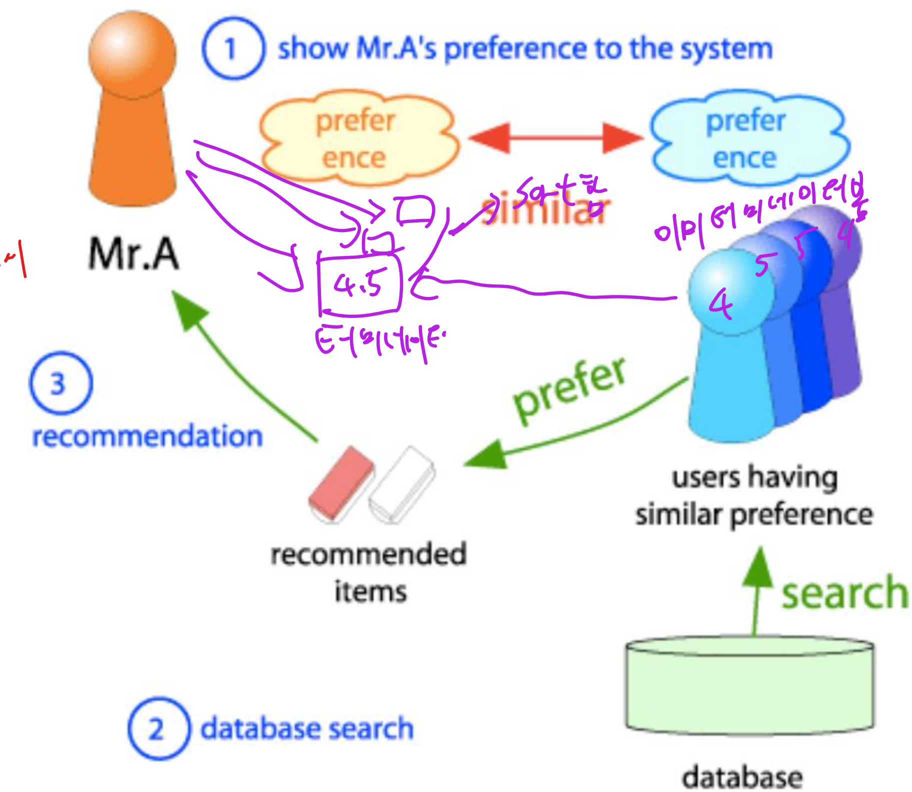
  inner product

  - $sim(u,i) = \cos(\overrightarrow{w_u}, \overrightarrow{w_i}) = \frac{\overrightarrow{w_u} \cdot \overrightarrow{w_i}}{\|\overrightarrow{w_u}\|_2 \times \|\overrightarrow{w_i}\|_2} = \frac{\sum_{k=1}^{K} w_{u,k} w_{i,k}}{\sqrt{\sum_{k=1}^{K} {w_{u,k}}^2} \sqrt{\sum_{k=1}^{K} {w_{i,k}}^2}}$
    - $K$: # of features
    - $\overrightarrow{w_u}$: user $u$'s profile
    - $\overrightarrow{w_i}$: item $i$'s contents
    - $w_{u,k}$: user profile's $k$-$th$ feature value
    - $w_{i,k}$: item content's $k$-$th$ feature value

# Collaborative Filtering (CF), KNN-based Method

- **Recommend such items rated high by *k-nearest neighbors* who have preferences similar to that of the target user**
  - **Step 1**: *Finding a k-nearest neighbors* whose preferences are similar to that of an active user *u*
  - **Step 2**: Estimating $r_{u,i}$, the rating of item i for active user u, based on the ratings given to item i by u's neighbors
  - **Step 3**: Recommending Top-N items with the ratings estimated high

A와 비슷한 prefer를 찾아서
A의 rating을 예측
neighbors의 rating을 토대로



# KNN-based Method

- **Easy example:** A database of ratings of the current user, Bob, and some other users is given:
- Determine whether Bob will like or dislike Item 5



| | Item1 | Item2 | Item3 | Item4 | Item5 |
|---|---|---|---|---|---|
| ***Bob*** | 5 | 3 | 4 | 4 | ??? |
| **User1** | 3 | 1 | 2 | 2 | 2 |
| **User2** | 4 | 3 | 4 | 3 | 5 |
| **User3** | 3 | 3 | 1 | 5 | 4 |
| **User4** | 1 | 5 | 5 | 2 | 1 |

# KNN-based Method

## Step 1-1: Similarity function between users

### Pearson correlation coefficient

- $sim(x,y) = \dfrac{\sum_{s \in S_{xy}} (r_{x,s} - \overline{r_x})(r_{y,s} - \overline{r_y})}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \overline{r_x})^2 \sum_{s \in S_{xy}} (r_{y,s} - \overline{r_y})^2}}$

### Cosine similarity

- $sim(x,y) = cos(\vec{x}, \vec{y}) = \dfrac{\vec{w}_x \cdot \vec{w}_y}{\|\vec{w}_x\|_2 \times \|\vec{w}_y\|_2} = \dfrac{\sum_{s \in S} r_{x,c} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} {r_{x,s}}^2} \sqrt{\sum_{s \in Sxy} {r_{y,s}}^2}}$

| | Item1 | Item2 | Item3 | Item4 | Item5 |
|---|---|---|---|---|---|
| ***Bob*** | 5 (+1.0) | 3 (-1.0) | 4 (0.0) | 4 (0.0) | **???** |
| User1 | 3 (+1.0) | 1 (-1.0) | 2 (0.0) | 2 (0.0) | 2 (0.0) |
| User2 | 4 (+0.2) | 3 (-0.8) | 4 (+0.2) | 3 (-0.8) | 5 (+1.2) |
| User3 | 3 (-0.2) | 3 (-0.2) | 1 (-2.2) | 5 (+1.8) | 4 (0.8) |
| User4 | 1 (-1.8) | 5 (+2.2) | 5 (+2.2) | 2 (-0.8) | 1 (-1.8) |

correlation이 높고

| | |
|---|---|
| $sim(Bob, U1)$ | 1.0 |
| $sim(Bob, U2)$ | 0.60 |
| $sim(Bob, U3)$ | 0.0 |
| $sim(Bob, U4)$ | -0.77 |

# KNN-based Method

## Step 1-2: Selecting Neighborhood

Hyperparameter

- Select $k$ neighbors sorted by the similarity values

2

Top-2 = {$U1$, $U2$}

| | Item1 | Item2 | Item3 | Item4 | Item5 | | |
|---|---|---|---|---|---|---|---|
| ***Bob*** | 5 (+1.0) | 3 (-1.0) | 4 (0.0) | 4 (0.0) | **???** | | |
| User1 | 3 (+1.0) | 1 (-1.0) | 2 (0.0) | 2 (0.0) | 2 (0.0) | $sim(Bob, U1)$ | 1.0 |
| User2 | 4 (+0.2) | 3 (-0.8) | 4 (+0.2) | 3 (-0.8) | 5 (+1.2) | $sim(Bob, U2)$ | 0.60 |
| User3 | 3 (-0.2) | 3 (-0.2) | 1 (-2.2) | 5 (+1.8) | 4 (0.8) | $sim(Bob, U3)$ | 0.0 |
| User4 | 1 (-1.8) | 5 (+2.2) | 5 (+2.2) | 2 (-0.8) | 1 (-1.8) | $sim(Bob, U4)$ | -0.77 |

Data
Intelligence
LAB

# KNN-based Method

## Step 2: Rating prediction

- Predict the rating by aggregating the neighbors' ratings on the item

$$r_{u,i} = \operatorname*{aggr}_{u' \in N} r_{u',i}^{\,2,5}$$

- $r_{u,i}$: Estimated rating on item i for user u  $i=5$
- $N$: Set of neighbors for user u ≈606  $N=\{u_1, u_2\}$

- Different methods for aggregation

$\frac{2+5}{2} = 3.5$

(a) $r_{u,i} = \frac{1}{|N|} \sum_{u' \in N} r_{u',i}$  그냥평균

(b) $r_{u,i} = \frac{1}{k} \sum_{u' \in N} sim(u,u') \times r_{u',i}$

$\underbrace{1 \cdot 2 + 0.6 \cdot 5}_{k} = \frac{5}{1+0.6} = \frac{5}{1.6} = 3.12..$

(c) $r_{u,i} = \boxed{\bar{r}_u} + \frac{1}{k} \sum_{u' \in N} sim(u,u') \times (r_{u',i} - \boxed{\bar{r}_{u'}})$

tendency 반영

Correlation은 비슷하지만 user1의 경우는 전체적으로 짜게 주니까 이러한 tendency를 반영하기 위해서

- $sim(u,u')$: similarity between user $u$ and user $u'$
- $k$: normalizing factor( $k = \underbrace{\sum_{u' \in N} |sim(u,u')|}$ )
- $\bar{r}_u$, $\bar{r}_{u'}$: Average of ratings of user $u$ and similar user $u'$

# KNN-based Method

## ❑ Variation: Item-based CF

- ❑ Search for KNN of an item i, rather than a user u

$$r_{u,i} = \operatorname*{aggr}_{i' \in N} r_{u,i'}$$

아이템에 대한 neighbor

- $r_{u,i}$: Estimated rating on item i for user u
- $N$: Set of neighbors for item i

ex) 터미네이터의 rating을 구할 때 아이언맨 ...

위의 식과 동일

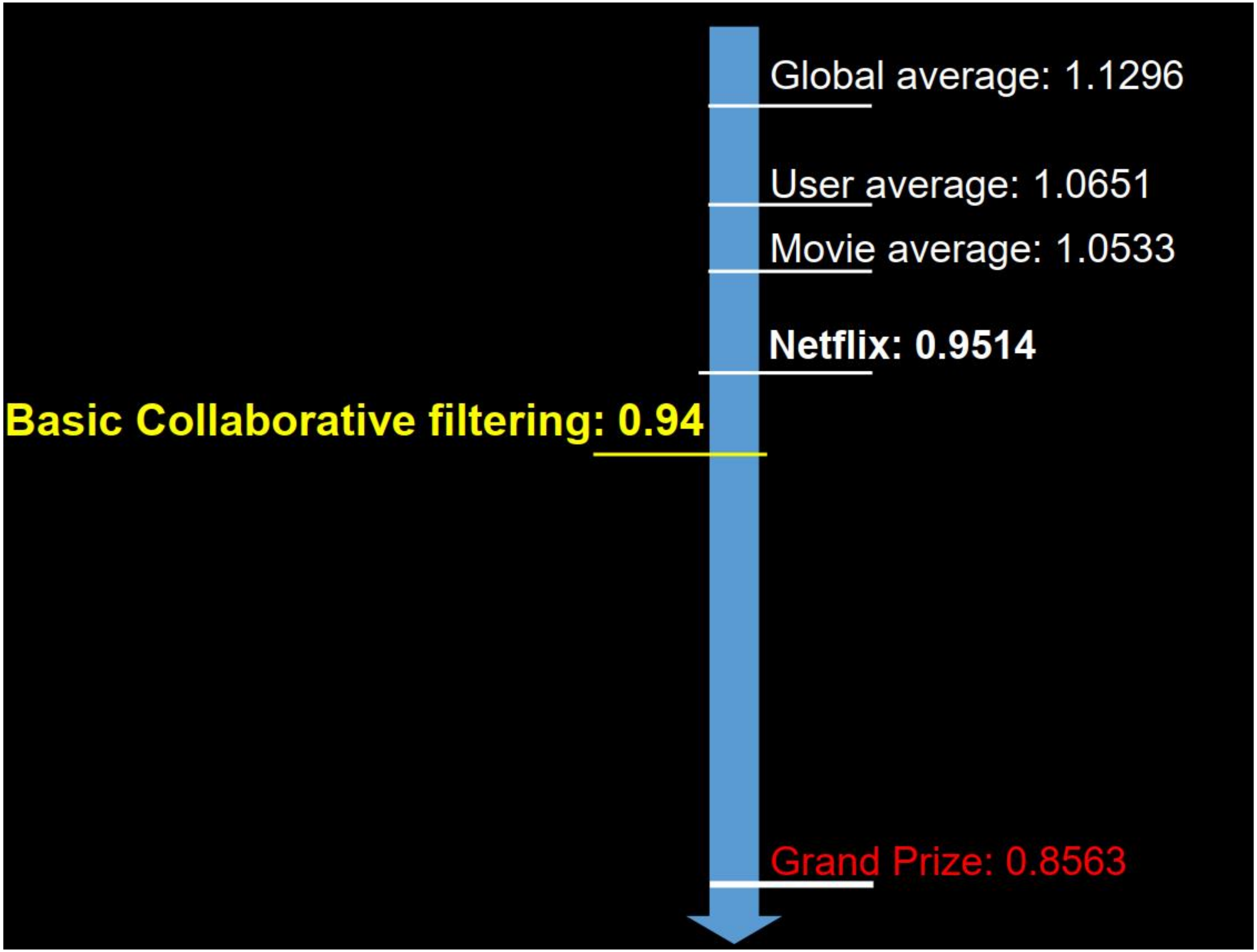
$$r_{u,i} = \frac{1}{|N|} \sum_{i' \in N} r_{u,i'}$$

$$r_{u,i} = \frac{1}{k} \sum_{i' \in N} sim(i, i') \times r_{u,i'}$$

user base에서는 이웃 유저가 터미네이터를 봤어야 함

item base에서는 A가 해리포터랑 아이언맨을 봤어야 한다

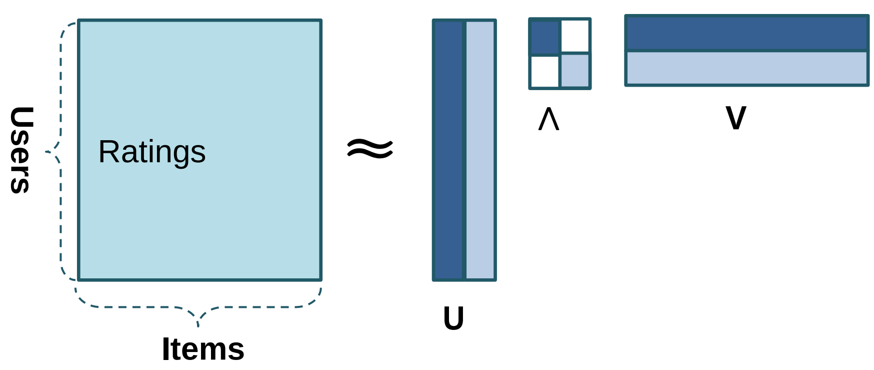
# KNN-based Method

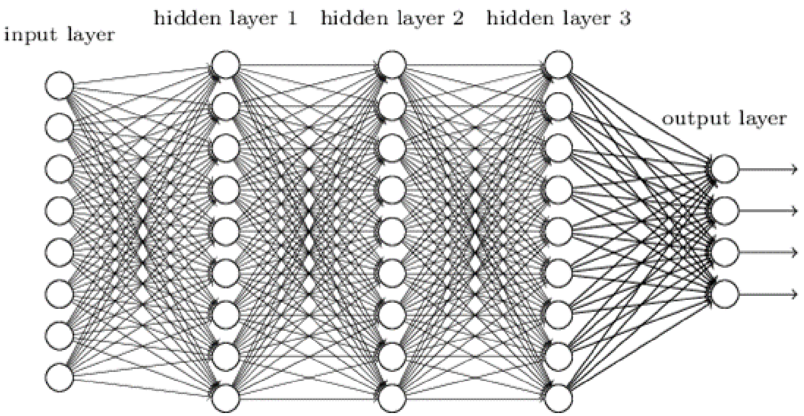


# Contents

# Latent Factor Models

- **So far…**
  - We learned KNN-based methods for recommender systems
  - However, the methods are heuristic-based, using hand-crafted functions
- **Model-based methods**
  - Latent factor models
    - Linear models: matrix factorization, SVD, …
    - Non-linear models: Autoencoder, deep neural networks, …



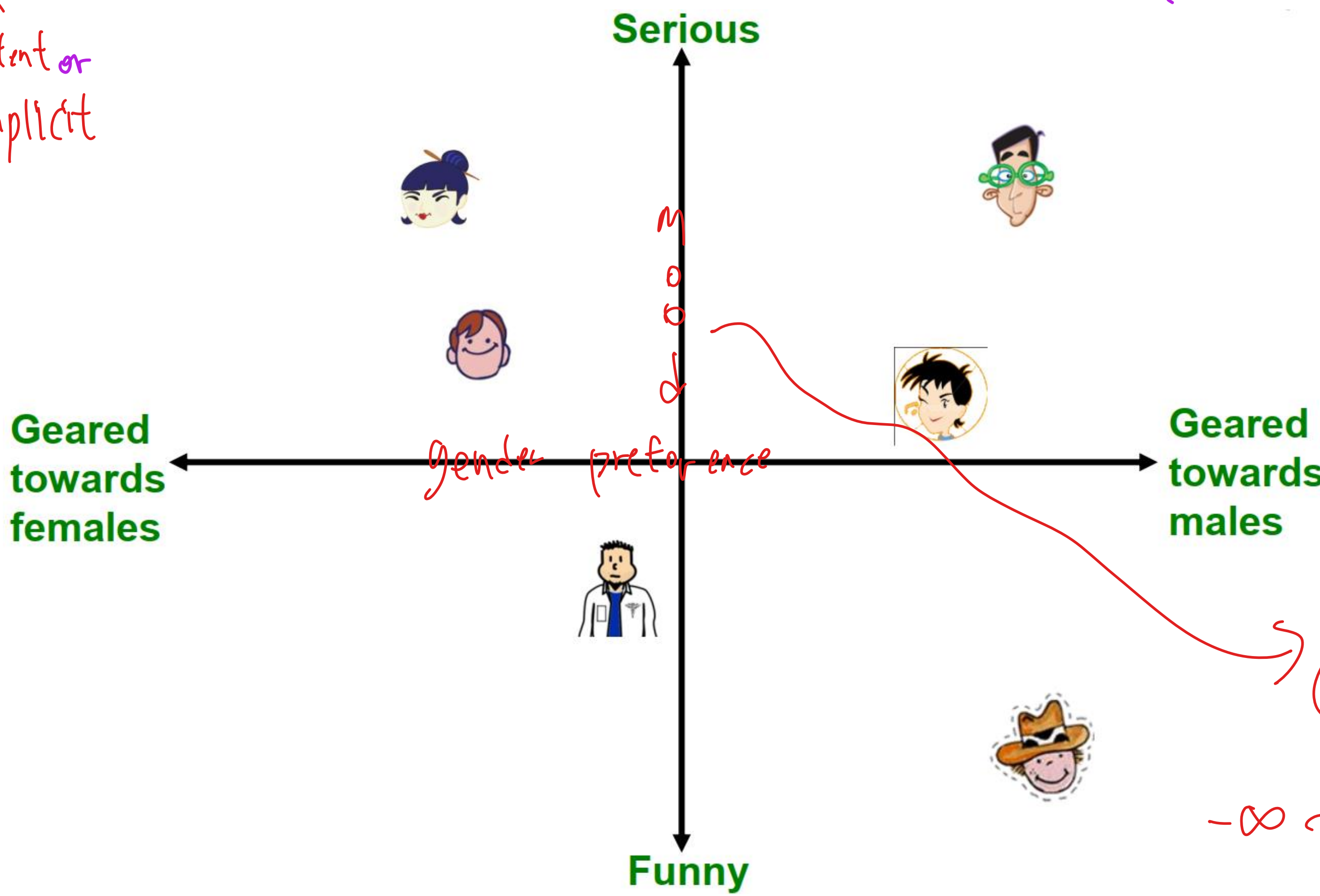



# Latent Factor Models

## ❑ What is latent factor?

- ❑ A feature that describe characteristics of users and items **hidden** in data

latent or
implicit

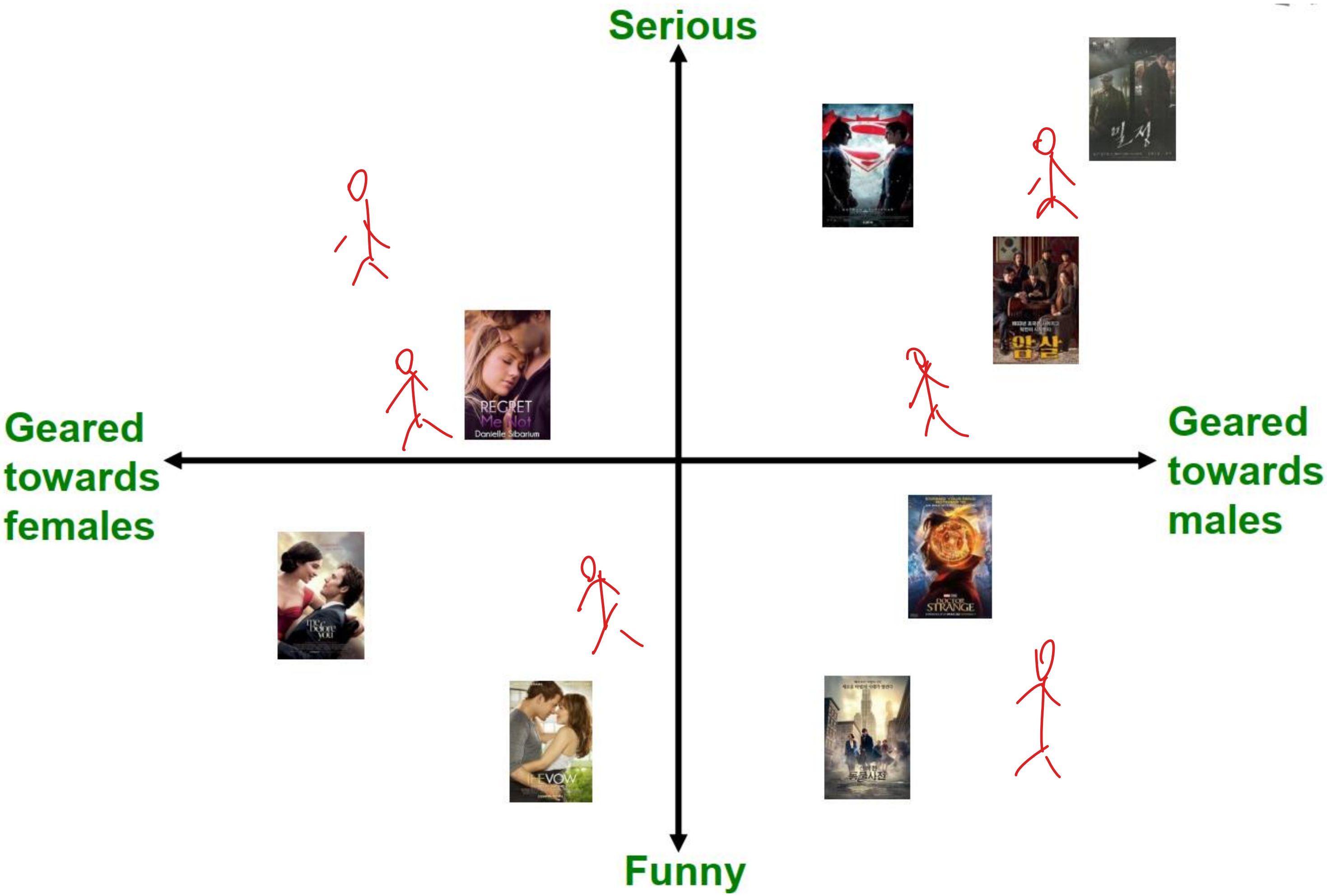



u
i
$z_1$ $z_2$
$-\infty \sim \infty$

# Latent Factor Models

## ❑ What is latent factor?

- ❑ A feature that describe characteristics of users and items **hidden** in data



# Thank You

Data
Intelligence
Lab