ie business school

RECOMMENDATION ENGINES

MASTER IN BUSINESS ANALYTICS AND BIG DATA

ABOUT THIS COURSE… (I/II)

- Session 1, Introduction to Recommendation Engines
- Session 2, Recommendation Methods

- Session 3, Collaborative Filtering
- *Session 4, **Recommendation Engine Labs (Part 1)***

- Session 5, Content-based Filtering and Hybrid Approaches
- Session 6, **Recommendation Engine Labs (Part 2)**

- Session 7, Building a Recommendation Engine in the Real World
- Session 8, **Recommendation Engine Labs (Part 3)**

- Session 9 & 10, **Final Project Evaluation**

TODAY'S AGENDA

- **Introductions**
- **Recapitulation Session 2…**
- **Session 3, Collaborative Filtering**
  - User-based Collaborative Filtering
  - Item-based Collaborative Filtering
  - Cold-start Problem
- **Session 4, Recommendation Engine Labs (Part 1)**
  - The Data for the labs…
  - Build a Non-Personalised Recommendation Engine
  - Build a User-based Collaborative Filtering

RECAPITULATION...

RECAP...

Long Tail

Paradox of Choice

Personalisation

Filter Bubble

RECAP…

Search Engines

Non-personalized

Personalized

Recommendation Engines

RECAP...

Recommendation Algorithms

Collaborative Filtering (user-based vs. item-based)

Content-based Filtering

WHAT IS A RECOMMENDER SYSTEM? (CONCEPT)

WHAT IS A RECOMMENDER SYSTEM? (ENGINEERING)

A Recommender processes information and transforms it into actionable knowledge

Recommender Components

- Knowledge Base
- Knowledge Processing Application
- Business Control & Analytics
- User Interface

It has a certain level of **autonomy** presenting recommendations to the end user

WHAT IS A RECOMMENDER SYSTEM? (ENGINEERING)

A Recommender processes information and transforms it into actionable knowledge

Recommender Components

- Knowledge Base
- Knowledge Processing Application
- Business Control & Analytics
- User Interface

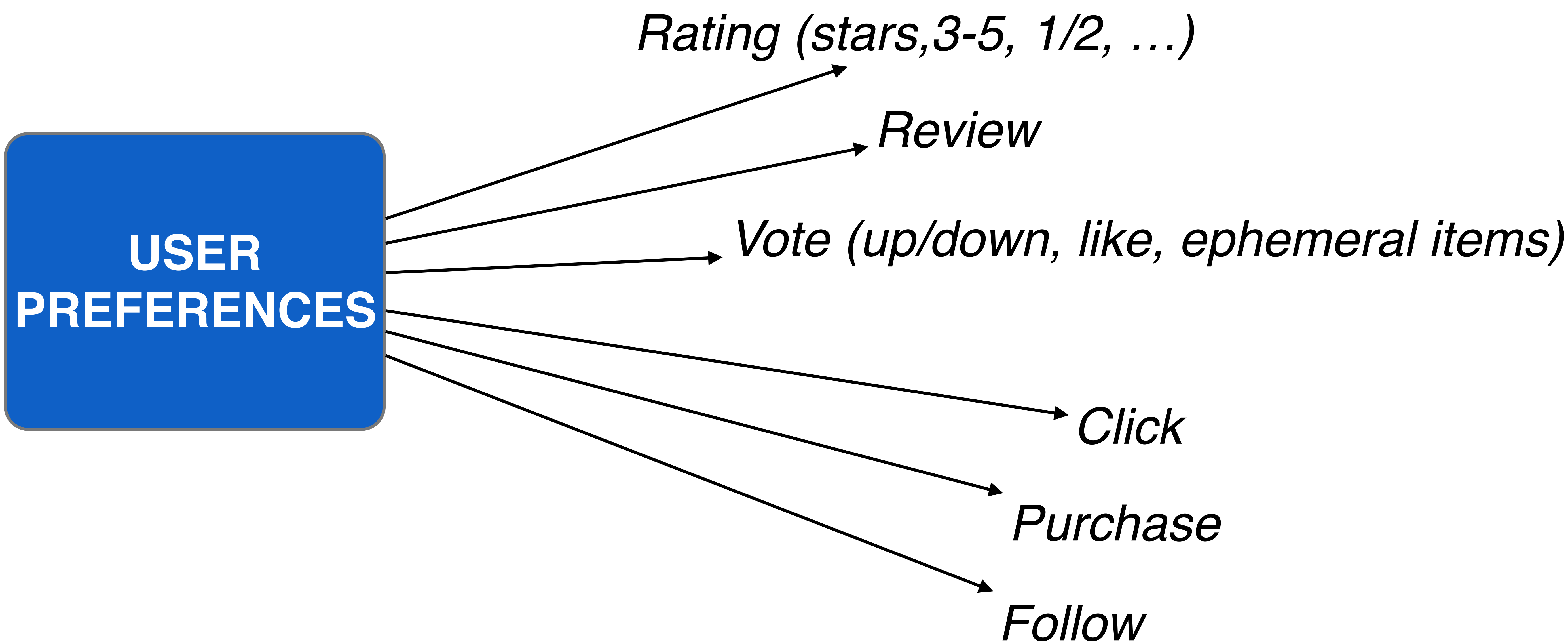
It has a certain level of **autonomy** presenting recommendations to the end user

THE KNOWLEDGE BASE

COLLECTING DATA

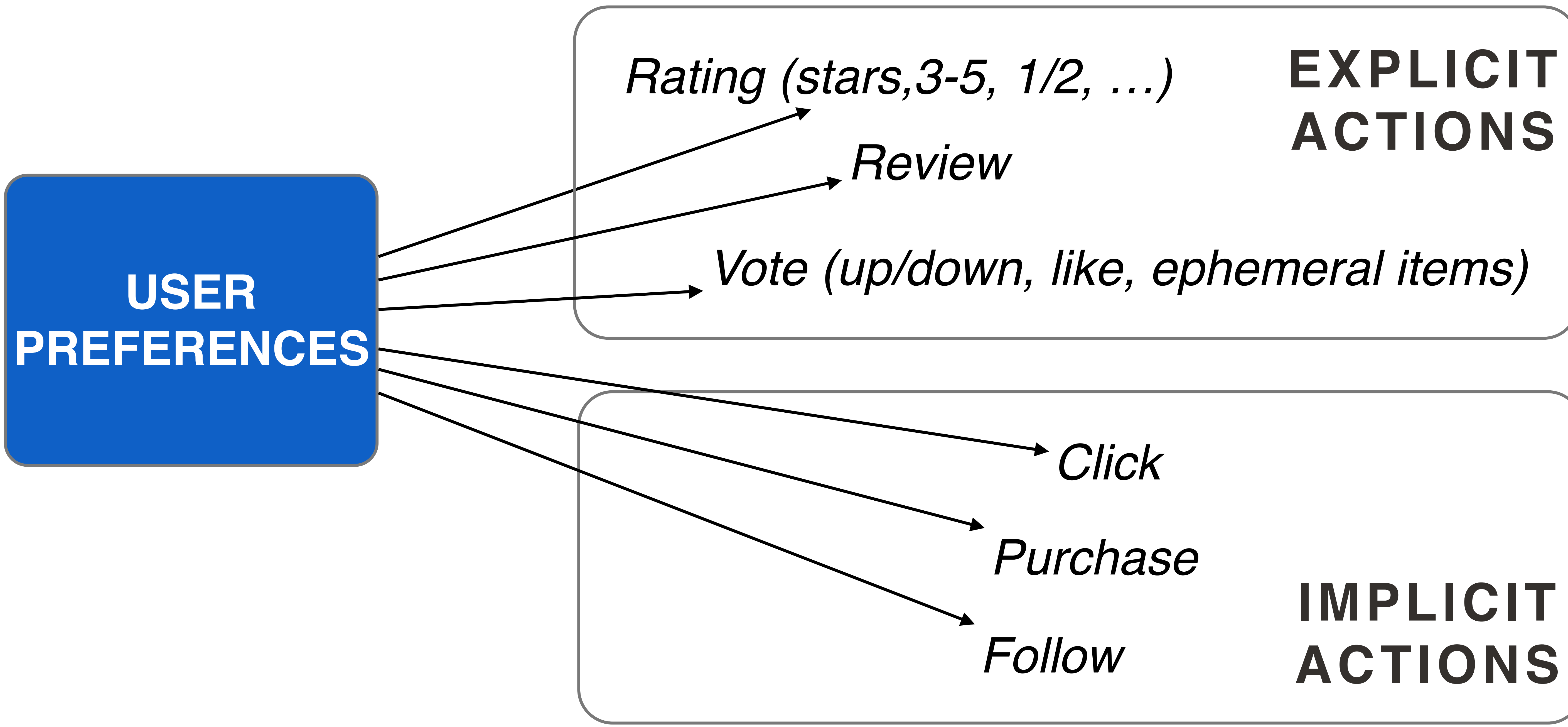
COLLECTING DATA

What the users do that might tell us something about their preferences?



COLLECTING DATA

What the users do that might tell us something about their preferences?



THE KNOWLEDGE PROCESSING APPLICATION

Recommendation methods

NON-PERSONALISED

PERSONALISED
- **COLLABORATIVE FILTERING**
- **CONTENT-BASED FILTERING**

There is NOT a BEST method… it all depends on the domain, goal, data, purpose, …

THE KNOWLEDGE PROCESSING APPLICATION

Recommendation methods

NON-PERSONALISED

VS

PERSONALISED

- COLLABORATIVE FILTERING
- CONTENT-BASED FILTERING

There is NOT a BEST method… it all depends on the domain, goal, data, purpose, …

NON-PERSONALISED RECOMMENDATIONS

if we do **NOT KNOW** the user,
the best we can do is to think he is like the majority of the users…

NON-PERSONALISED RECOMMENDATIONS

reminder: *a*ll users get the same recommendations…

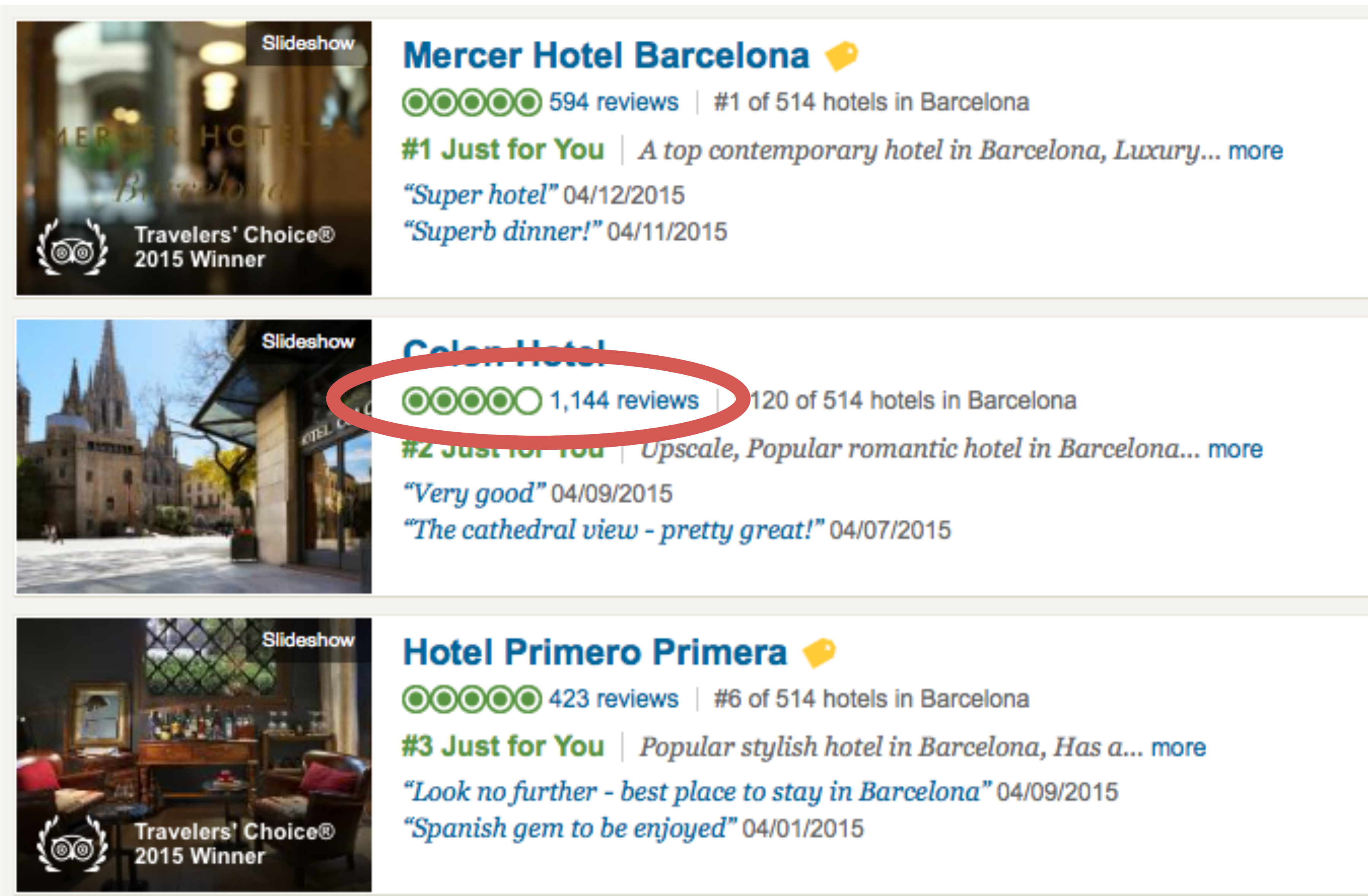
MIGHT BE SIMPLE, BUT MULTIPLE OPTIONS:

- top sold items
- most liked items
- top listened songs
- newest hits
- trending topics
- etc…

SOLVES THE COLD-START PROBLEM

NON-PERSONALISED RECOMMENDATIONS

Using *explicit actions*



user-item matrix:

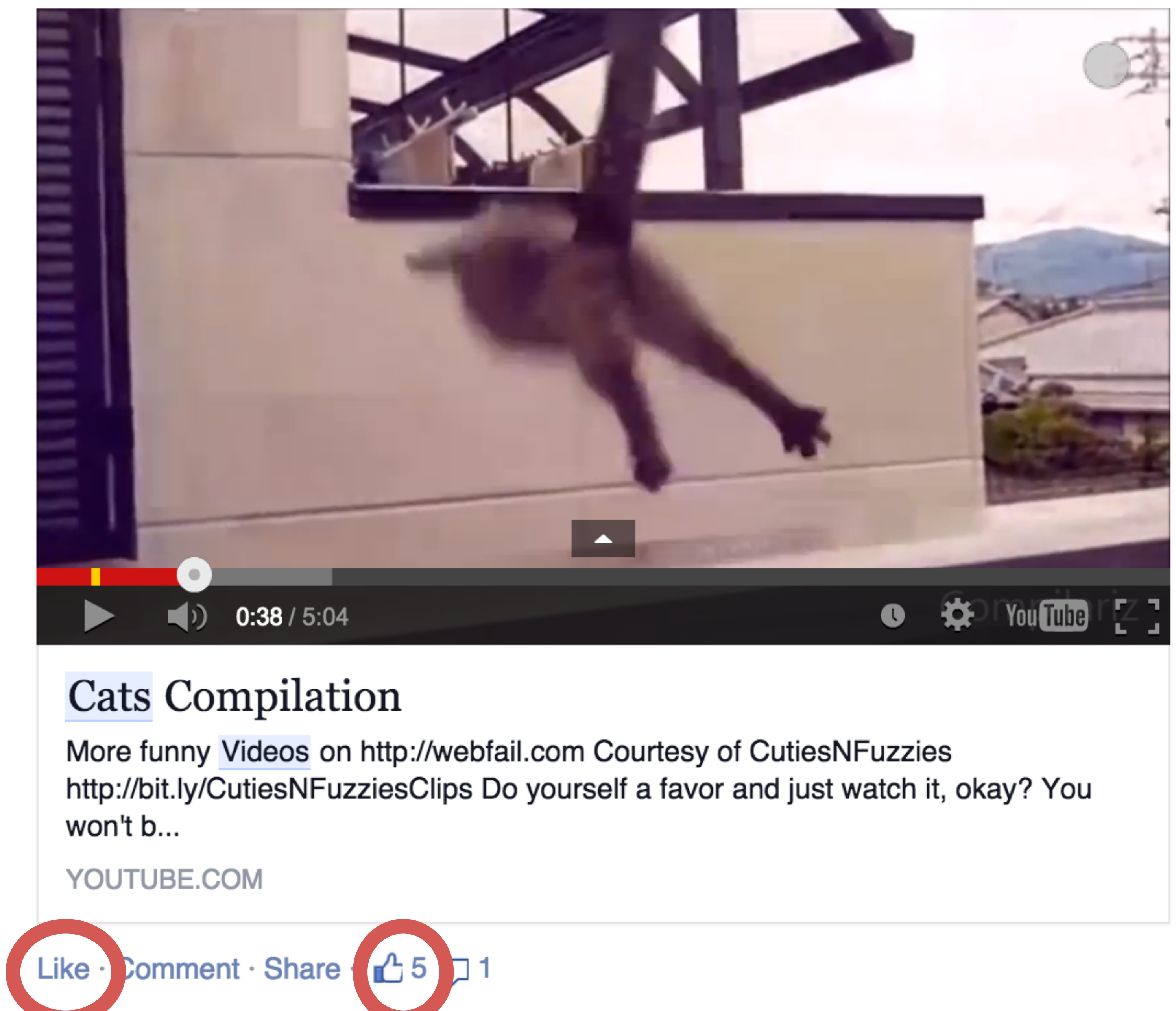
hotels

users					j					
					3					
					5					
					4					
i					4					
					3					
					4					
					3					
					5					
					5					
					4					

hotels (j) = avg(j) = 4

NON-PERSONALISED RECOMMENDATIONS

Using *explicit actions*



user-item matrix:

posts

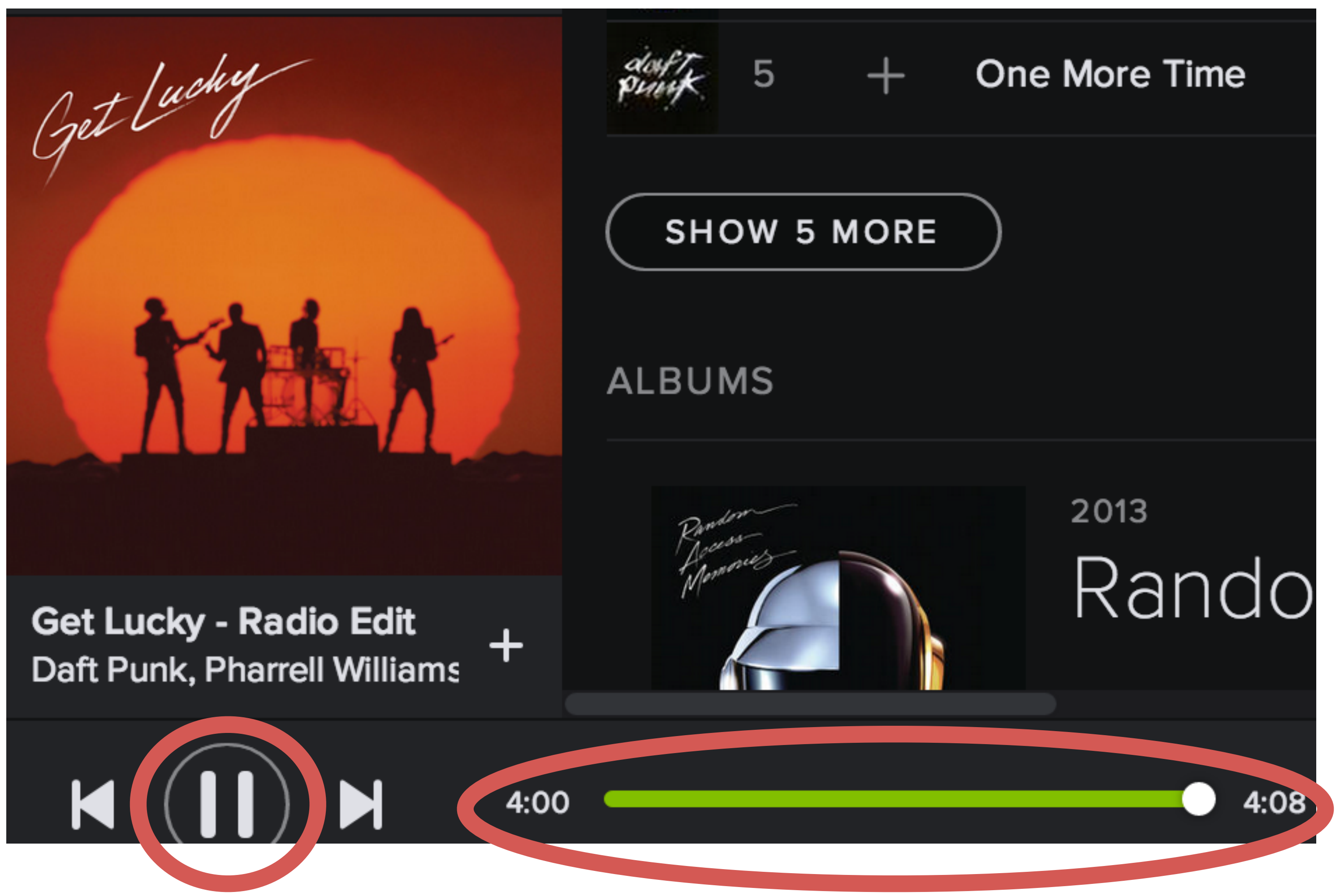
users

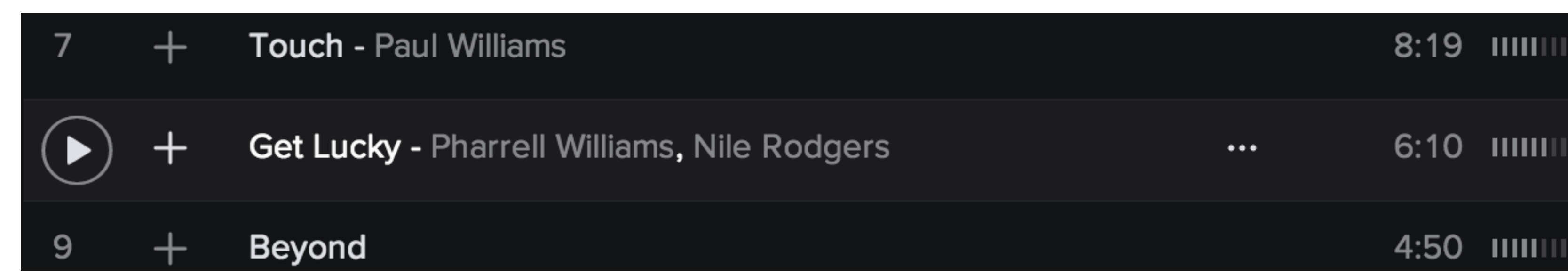
					j					
					1					
					1					
					1					
i					1					
					0					
					1					
					1					
					0					
					1					
					7					

posts (j) = sum(j) = 7

NON-PERSONALISED RECOMMENDATIONS

Using *implicit actions*





(1) the total number of plays compared to other tracks and (2) how recent those plays are.

the more a song is played the higher its popularity

ie business school

NON-PERSONALISED RECOMMENDATIONS

Using *external sources of data*

APIs !

NON-PERSONALISED RECOMMENDATIONS

Helping on **new user** *cold-start problem*

Your Watchlist is empty

Use Your Watchlist to track movies and TV shows that interest you.

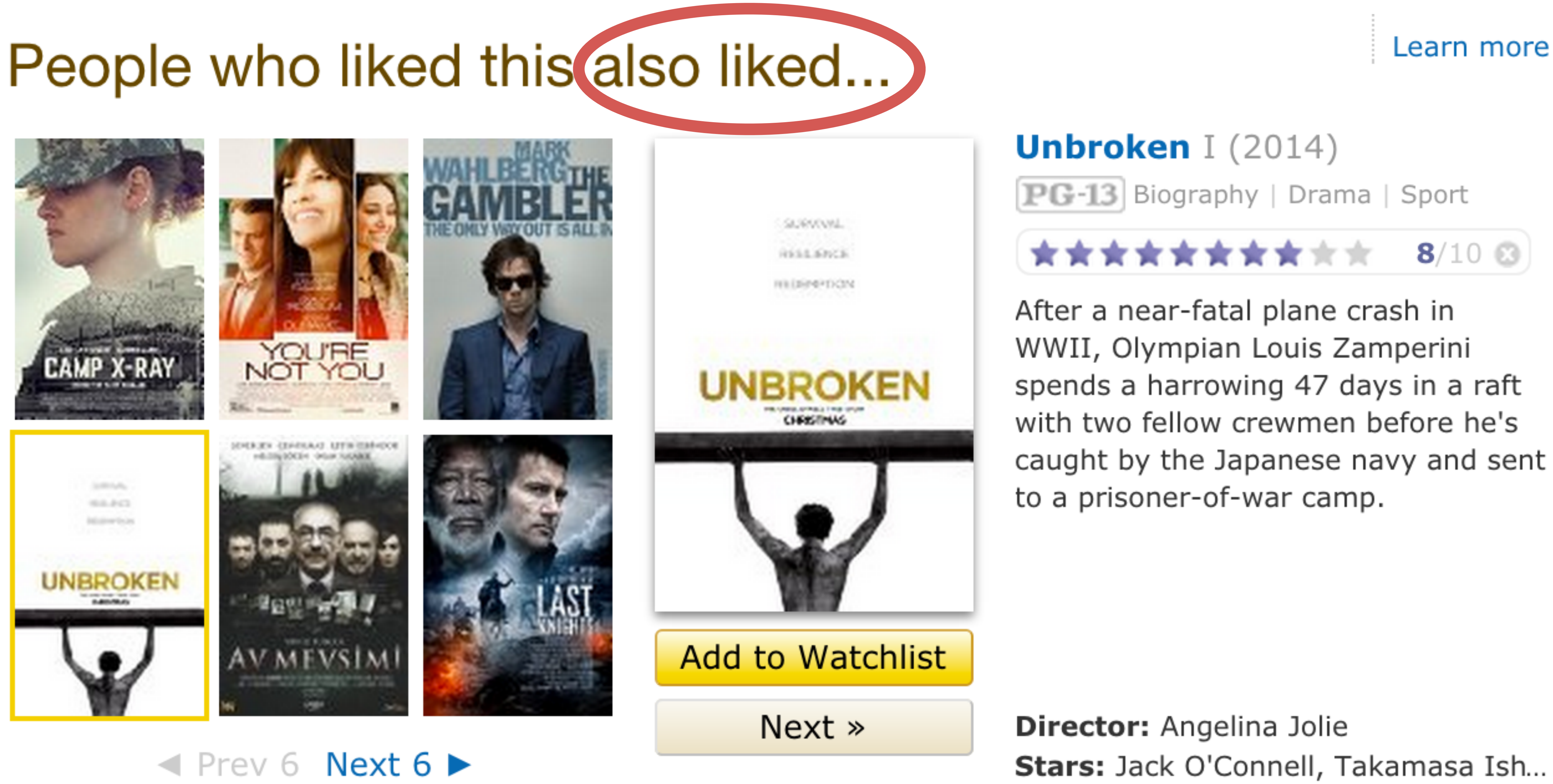
When you see a title you would like to add, click the ribbon on the poster OR click the "Add to Watchlist" button.

Explore these great titles to add to your list

IMDb Top 250 »
Top Movies by Genres »
Best Picture Winners »
Popular TV Series »

NON-PERSONALISED RECOMMENDATIONS

reminder : All users still get the same recommendations… product association



NON-PERSONALISED RECOMMENDATIONS

Product association

People who bought/liked **X** also bought/liked **Y** ...

(X and Y) / X 90/100

90% who bought tomatoes also bought cucumber

NON-PERSONALISED RECOMMENDATIONS

Product association and the best seller problem

But what if **Y** is a common purchase?

(X and Y) / X 90/100

90% who bought tomatoes, or cucumber, or garlics, or any other product also bought bananas…

NON-PERSONALISED RECOMMENDATIONS

Product association and the best seller problem

A better approach **…**

How many times Y appears with X

$$\frac{\textbf{(X and Y) / X}}{\textbf{(!X and Y) / !X}}$$

How many times Y appears with other products

if Y is very popular, denominator is high & decreases score

People who bought/liked **X** also bought/liked **Y …**

NON-PERSONALISED RECOMMENDATIONS

Product association and the best seller problem

A better approach **...**

$$\frac{(X \text{ and } Y)/X}{(!X \text{ and } Y)/!X}$$

tomatoes -> cucumber?

$$\frac{90/100}{90/500} = \frac{0{,}9}{0{,}18} = 5$$

STRONGER ASSOCIATION

tomatoes -> banana?

$$\frac{90/100}{400/500} = \frac{0{,}9}{0{,}8} = 1{,}125$$

People who bought/liked **X** also bought/liked **Y ...**

NON-PERSONALISED RECOMMENDATIONS

Product association and the best seller problem

another probabilistic approach (non-directional)

$$\frac{(X \text{ and } Y)\ /\ X}{(!X \text{ and } Y)\ /\ !X}$$

X -> Y

$$\frac{P(X \text{ AND } Y)}{P(X) * P(Y)}$$

X <-> Y

LAB 1!

LAB 1 - EVALUATION

EVALUATION			EXAMPLE
	Correct results	50 %	7
	Clean, documented and executable code	40 %	10
	Code efficiency	10 %	2
Homework Delivery		**FINAL SCORE**	7,7

Accepted programming languages: R
Accepted delivery: R Markdown file
Recommended IDE: R Studio
Individual work

LAB 1 - DATA

- **Category**: movies
- **Type**: explicit, multi-level ratings
- **Format**: csv
- **Size**: 20x20 matrix

User	260: Star War	1210: Star Wa	356: Forrest (	318: Shawsha	593: Silence c	3
755: John	1	5	2		4	
5277: Maria	5	3		2	4	
1577: Anton				5	2	
4388: Roger		3				
1202: Martina	4	3	4	1	4	
3823: Ana	2	4	4	4		
5448: Sergi			3	1	1	
5347: Marc	4				3	
4117: Jim	5	1		4	2	
2765: Chris	4	2		5	3	
5450: Bernarc	2	1	5			
139: Nuria	3	5	2		2	
1940: Nerea	2	3		5	4	
3118: Carles	3		3		2	
4656: Victoria	4	4			5	
4796: Ivan			1		3	
6037: Rachel					4	
3048: Nadia	4	5	1	5	1	
4790: Oriol	5	1				
4489: Valery	1	2	2	4	5	

LAB 1

PART 1!

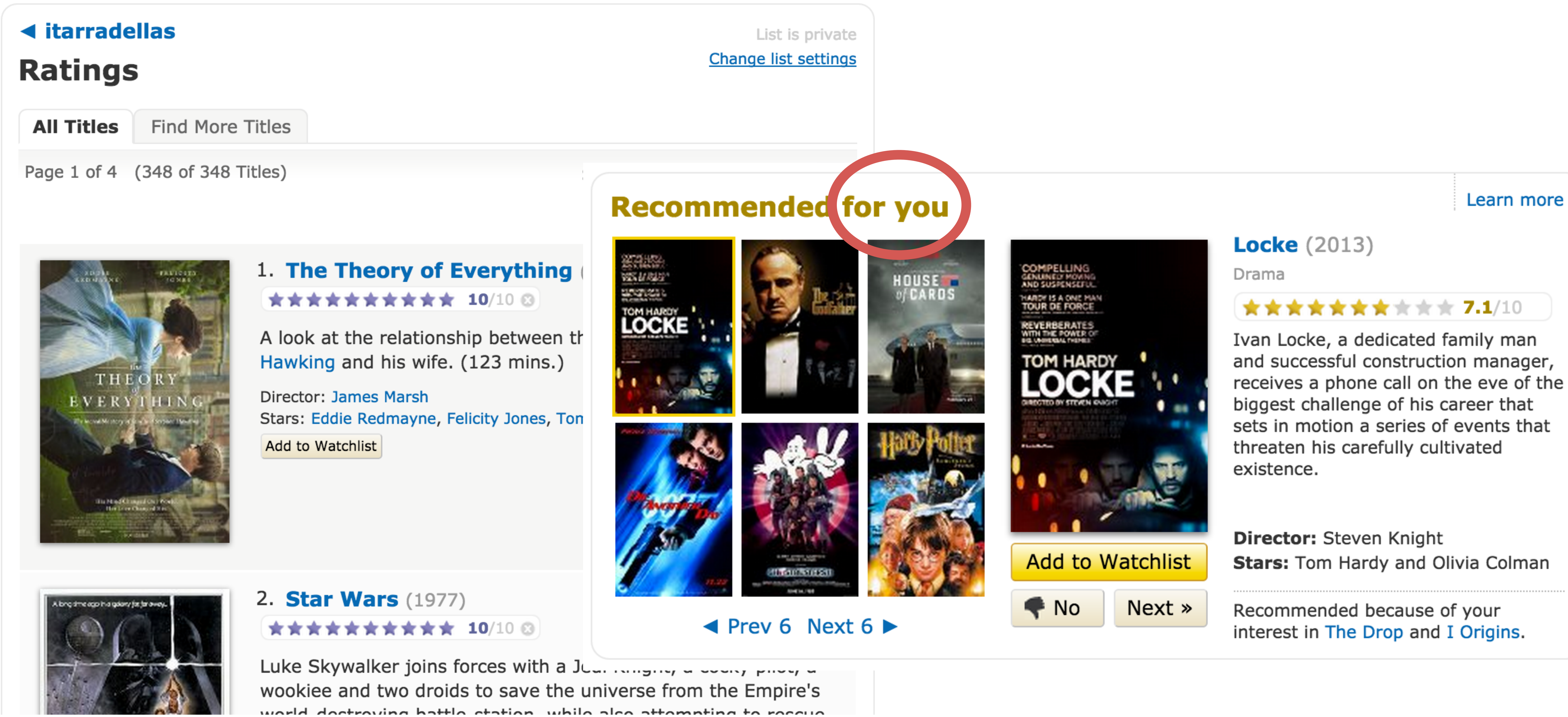
LAB 1.1 - NON PERSONALIZED

Build a Non Personalised Recommendation Engine

- **Mean rating**. Return the average rating per each movie, in descending order.
- **Top percentage**. Return the top percentage of movies equal or greater than a specific rating, in descending order.
- **Rating count**. Return the total number of ratings per each movie, in descending order.
- **People who watched x also watched y**. Return the top n movies that are related to a specific one. Calculate movies that most often occur with other movie.
- **People who liked x also liked y**. Return the top n movies that are related to a specific one. Calculate movies that are most often liked with other movie.

Homework

VS PERSONALISED RECOMMENDATIONS



SESSION 3,

COLLABORATIVE FILTERING

- User-based Collaborative Filtering
- Item-based Collaborative Filtering
- Cold-start Problem

COLLABORATIVE FILTERING

How do **YOU GET RECOMMENDATIONS** on movies (analogically)?

				j						
u1										
u2										
u3	5		5		2		4			5
u4										

COLLABORATIVE FILTERING

Collect **data** about the items rated from a set of users

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
u1	5	3	4	2			4	2		5
u2	2	3		3		2		1		1
u3	5		5		2		4			5
u4	2			5	5	4		3	3	

j1 **…**

j10

COLLABORATIVE FILTERING

Identify **target user** (you) and prediction (movie you are interested in).

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
u1	5	3	4	2			4	2		5
u2	2	3		3		2		1		1
u3	5		5	?			4			5
u4	2			5	5	4		3	3	

j1 ... **j4 ?** ... **j10**

COLLABORATIVE FILTERING

Identify **neighbours** to the target user (usually best friends have similar taste).

Need a similarity function!

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
u1	5	3	4	2			4	2		5
u2	2	3		3		2		1		1
u3	5		5	?			4			5
u4	2			5	5	4		3	3	

j1 ... j4 ? ... j10

COLLABORATIVE FILTERING

Identify **items** these neighbours have rated.

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
u1	5	3	4	2			4	2		5
u2	2	3		3		2		1		1
u3	5		5	?			4			5
u4	2			5	5	4		3	3	

j1 ... j4 ? ... j10

COLLABORATIVE FILTERING

Generate a **prediction** of those unknown items for the target user.

	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
u1	5	3	4	2			4	2		5
u2	2	3		3		2		1		1
u3	5	**3**	5	**2**			4	**2**		5
u4	2			5	5	4		3	3	

j1 ... j4 2 ... j10

COLLABORATIVE FILTERING

1. User-based CF
- Select neighbour users
- Use their ratings

2. Item-based CF
- Compute similarity among items based on ratings
- Use target user ratings to deduce recommendations

- Item attributes are irrelevant (**content agnostic**!)
- Can be used with **highly sparsity user-items matrices**

COLLABORATIVE FILTERING

User-based CF

Approaches:

1. Non-personalised
   - take average of ratings for ***j***

2. Personalised
   - take weighted average of ratings for ***j***
   - ***how to weight users?***
   - *more similar users get more weight!*
   - similarity function:
     - what neighbours to consider?
     - how much weight they get?

movies (m)

users (u)

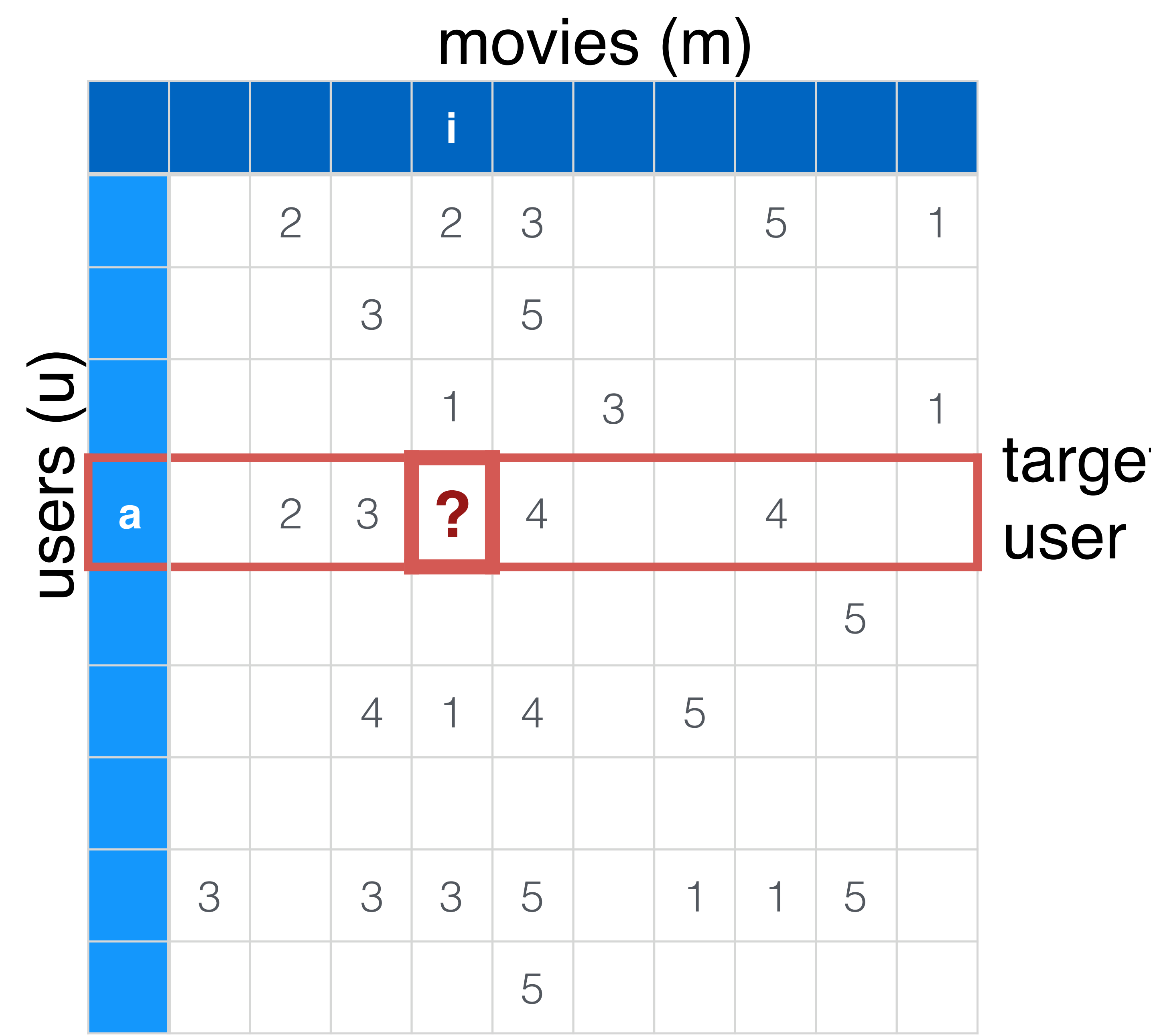
				j						
		2		2	3			5		1
			3		5					
				1		3				1
i		2	3	?	4			4		
									5	
			4	1	4		5			
	3		3	3	5		1	1	5	
					5					

target user

COLLABORATIVE FILTERING (USER-BASED)

Pros and Cons

1. Pros
   - No specific product or user features are used (content-agnostic)
2. Cons
   - Items need to be “standardised” in a catalog
   - Needs “lots” of reliable opinions/ratings



COLLABORATIVE FILTERING (USER-BASED)

NON-PERSONALISED

?

prediction

rating user, movie

$$Pa,i = \frac{\sum_{u=1}^{n} r_{u,i}}{n} \quad =9/4=2,25$$

user a

movie i

total users with ratings

movies (m)

users (u)

				i						
		2		2	3			5		1
			3		5					
				1		3				1
a		2	3	?	4			4		
									5	
			4	1	4		5			
	5		5	5	5		5	5	4	
					5					

target user

the average of **ALL** the ratings for the item to predict

COLLABORATIVE FILTERING (USER-BASED)

NON-PERSONALISED (normalised)

What if ratings are not in the same **scale**?
Users rate things in different scales!

?

prediction

avg. rating user a

ratings user

mean ratings user

$$Pa, i = \bar{r_a} + \frac{\sum_{u=1}^{n}(r_{u,i} - \bar{r_u})}{n}$$

user a

movie i

=3,25+(-0,89)=2,36

<u>Consider deviation of user ratings</u>. How much does a user likes an item compared to how much normally likes it.

movies (m)

users (u)

				i							avg(ru)
		-0,6		-0,6	0,4			2,4		-1,6	2,6
			-1		1						4
				-0,67		1,33				-0,67	1,67
a		2	3	?	4			4			3,25
									0		5
			0,5	-2,5	0,5		1,5				3,5
	0,2		0,2	0,2	0,2		0,2	0,2	-0,8		4,8
					0						5

COLLABORATIVE FILTERING (USER-BASED)

PERSONALISED (better approach)

the **weighted average** of **SIMILAR USERS** ratings for the item to predict. Reflects the agreement between users.

movies (m)

users (u)

				i							avg(ru)	w(a,u)
		-0,6		-0,6	0,4			2,4		-1,6	2,6	0.7
			-1		1						4	0,9
				-0,67		1,33				-0,67	1,67	0
a		2	3	?	4			4			3,25	1
									0		5	0
			0,5	-2,5	0,5		1,5				3,5	0.8
												0
	0,2		0,2	0,2	0,2		0,2	0,2	-0,8		4,8	0.5
					0						5	0.7

personalised weight user a - user u rating the movie. 1 is perfect match

scaled ratings

prediction

$$Pa, i = \bar{r_a} + \frac{\sum_{u=1}^{n} (r_{u,i} - \bar{r_u}) \cdot w_{a,u}}{\sum_{u=1}^{n} w_{a,u}}$$

user a

movie i

sum of the weights,
to still taking the average

COLLABORATIVE FILTERING (USER-BASED)

FINDING SIMILAR USERS

How to determine the similarity metric (w) to use?

$$Pa, i = \bar{r_a} + \frac{\sum_{u=1}^{n}(r_{u,i} - \bar{r_u}) \cdot w_{a,u}}{\sum_{u=1}^{n} w_{a,u}}$$

Pearson Correlation

Euclidean Distance Score

Vector Correlation

Jaccard coefficient

Manhattan distance

Multiple ways, **depending of problem** to solve and data collected
Take baseline metric (w), and **challenge** against baseline **iteratively**.

COLLABORATIVE FILTERING (USER-BASED)

FINDING SIMILAR USERS

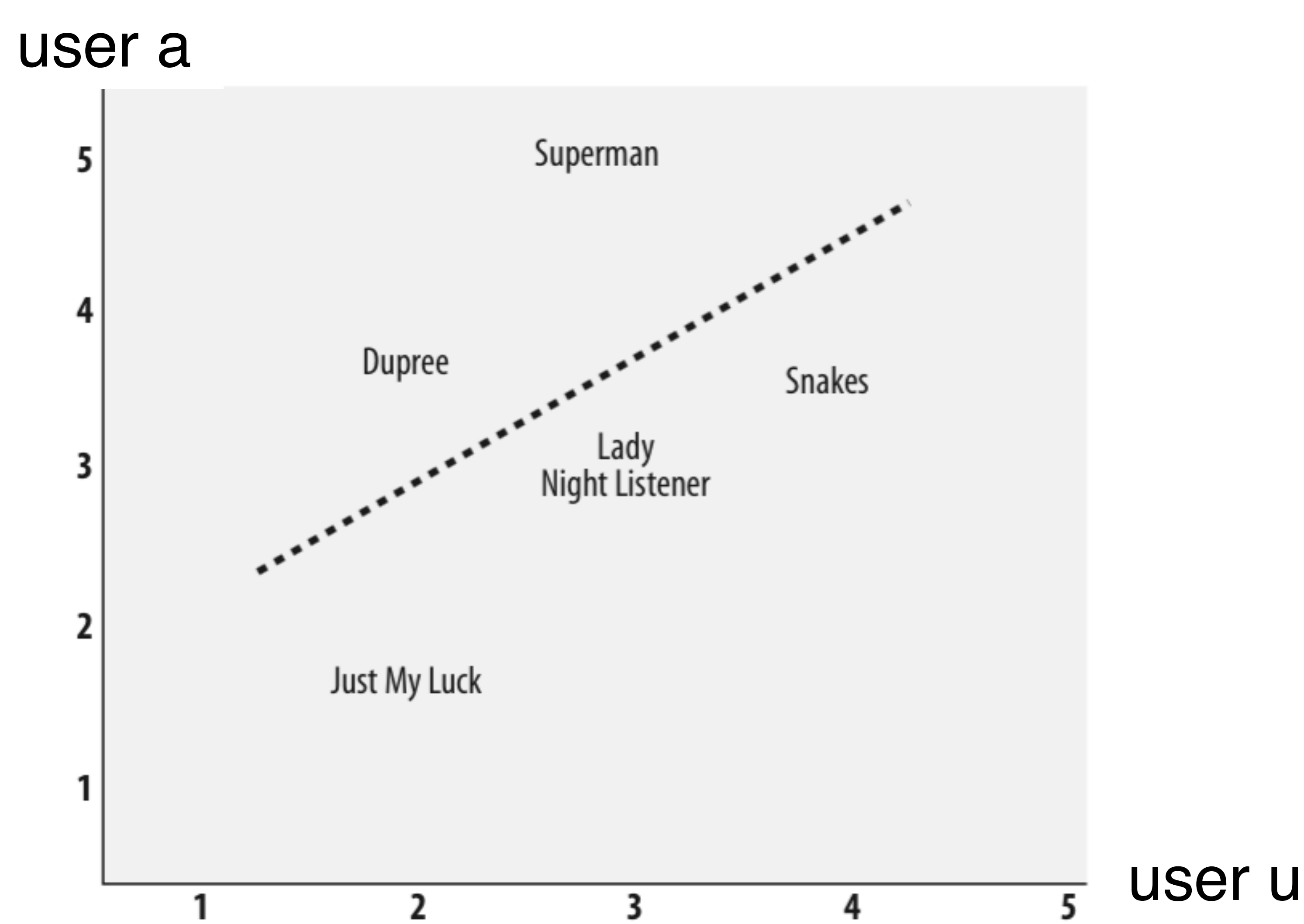
Good similarity metric (w) for movie ratings?

Pearson Correlation	• Items in common (m) big • Multi-level ratings

COLLABORATIVE FILTERING (USER-BASED)

PEARSON CORRELATION

How well two sets of data fit on a straight line



COLLABORATIVE FILTERING (USER-BASED)

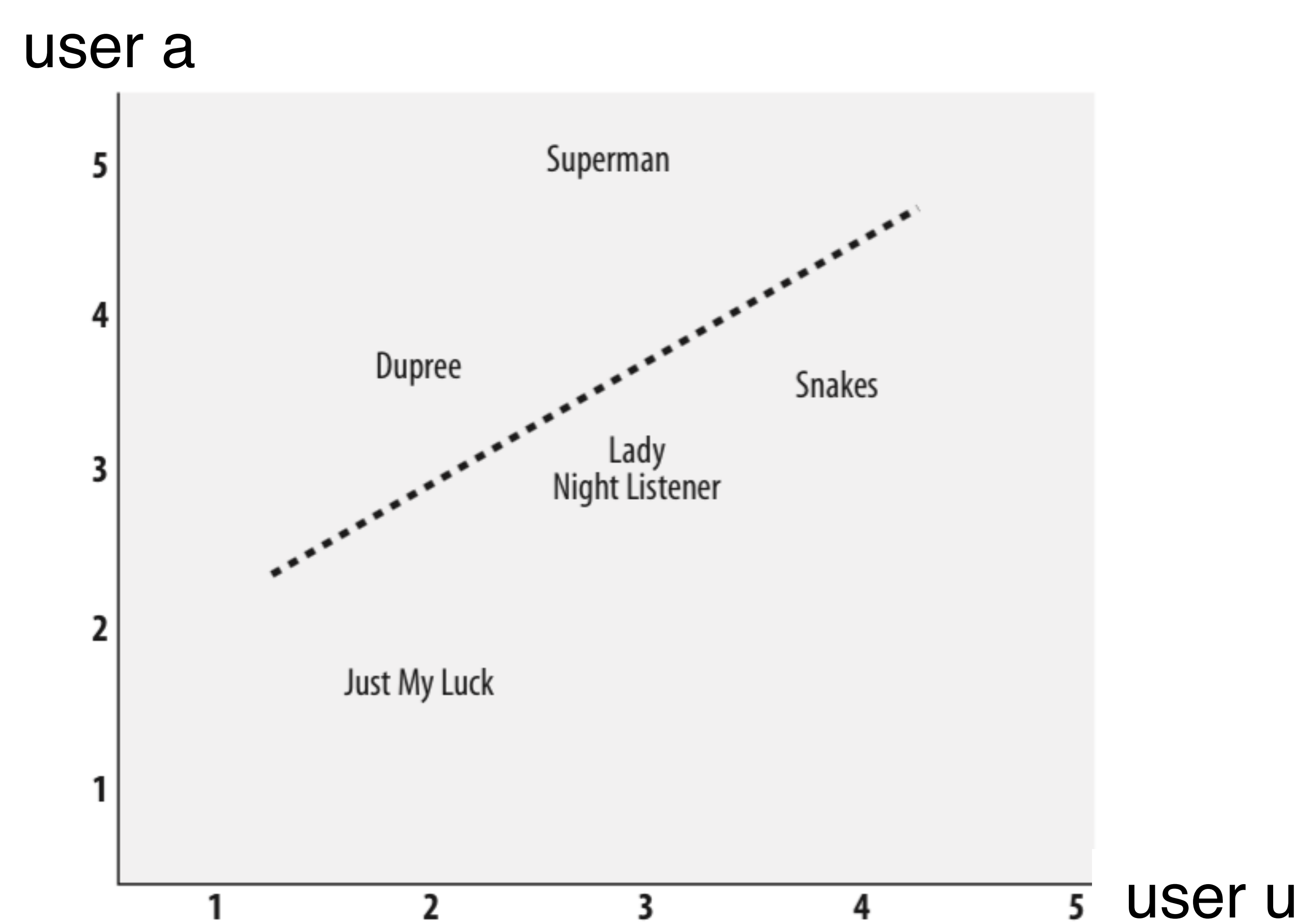
PEARSON CORRELATION

weight goes up when:

- items I (a) **love** you (u) love
- items I (a) **hate** you (u) hate

$$w_{a,u} = \frac{\sum_{i=1}^{m}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sigma_a \sigma_u}$$

higher weights when people **agree** a lot and specially when they agree on the **extremes**



COLLABORATIVE FILTERING (USER-BASED)

PEARSON CORRELATION

items
in **common**

scaled rating
user a item i

scaled rating
user u item i

$$w_{a,u} = \frac{\sum_{i=1}^{m}(r_{a,i} - \bar{r_a})(r_{u,i} - \bar{r_u})}{\sigma_a \sigma_u}$$

weight /
similarity
user a &
user u

std dev
user a

std dev
user u

$$\rightarrow \sqrt{\sum_{i=1}^{m}(r_{a,i} - \bar{r_a})^2} \cdot \sqrt{\sum_{i=1}^{m}(r_{u,i} - \bar{r_u})^2}$$

COLLABORATIVE FILTERING (USER-BASED)

Question!

What’s the key element in CF User-Based formula?

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SELECTION STRATEGY (KEY)

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SELECTION STRATEGY (KEY)

ALL (Computational expensive if U is large, caching mechanisms may be used, introduces noise)

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SELECTION STRATEGY (KEY)

Solving the scalability bottleneck

RANDOM (Can Introduce noise)

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SELECTION STRATEGY (KEY)

Solving the scalability bottleneck

THE MORE SIMILAR (Threshold reduces computational effort, i.e. Top 50 with similarity > 0.75)

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SELECTION STRATEGY (KEY)

Solving the scalability bottleneck

Combined (Be creative for the specific problem, i.e. 200 Random + top-50 more similar)

COLLABORATIVE FILTERING (USER-BASED)

Question!

$$Pa,i = \bar{r_a} + \frac{\sum_{u=1}^{n}(r_{u,i} - \bar{r_u}) \cdot w_{a,u}}{\sum_{u=1}^{n} w_{a,u}}$$

What happens with dissimilar neighbours?

COLLABORATIVE FILTERING (USER-BASED)

DISSIMILAR NEIGHBOURS MAY RECOMMEND YOU THINGS EVERYBODY HATES

As you usually like what they hate ;)

Two options to fix it : Exclude these neighbours, absolute denominator

COLLABORATIVE FILTERING (USER-BASED)

KEEPING NEGATIVE SIMILARITIES UNDER-CONTROL

SO, A BETTER FORMULA:

$$Pa, i = \bar{r_a} + \frac{\sum_{u=1}^{n} (r_{u,i} - \bar{r_u}) \cdot w_{a,u}}{\sum_{u=1}^{n} |w_{a,u}|}$$

COLLABORATIVE FILTERING (USER-BASED)

Question!

How many neighbours would you choose?

COLLABORATIVE FILTERING (USER-BASED)

NEIGHBOURHOOD SIZE SELECTION

The size has significant impact, adjust it to improve accuracy & scalability

↓

You have good data already, so …

↓

Experiment, use different sizes and measure results, then select the optimal size

*Between 25 and 100 is often used ;)

LAB 1

PART 2!

LAB 1.2 - USER-BASED CF

Build a Collaborative Filtering Engine

- **Pearson correlation coefficient**. Given two persons calculate the similarity distance using this weight formula.
- **Critics comparison.** Show the movie ratings for two persons on a scatter plot. How is this compared to other critics? What it means?
- **Movies recommendation**. Return all the recommendations for a given user, using the Pearson correlation.
- **Top n similar users.** Return the top n similar users to a given user, using the Pearson correlation.

Homework

COLLABORATIVE FILTERING (ITEM-BASED)

User-based CF. Great algorithm in the field but:

Sparsity Issues. Too often no rec. is possible

- large data sets
- small number of ratings

Computational costs* :

- with million of users
- when opinions change
- near real-time requirements

*Some costs may be reduced with caching strategies

				i						
		2						5		1
			3							
						3				1
a		2		?						
					4					
	3								5	
					5					

COLLABORATIVE FILTERING (ITEM-BASED)

Item-based CF. Solves the User-based CF problems when:

A lot more users than items. IUI >> III

- ie. 10 times more

Item-item similarities are more stable

- except time-bound items

Better computational performance

- with large data sets
- more pre computability options

items (i)

users (u)

	2		
5		1	3
		2	
	2		
1		4	4
		6	
		3	5
3		4	
	2		2
2		5	5
4		1	4

COLLABORATIVE FILTERING (ITEM-BASED)

1. **Compute similarity** among items based on ratings. Given a set of item rating vectors:

- Calculating the vectors Correlation (multi-level ratings)
- Calculating the vectors Cosine
- Using an Adjusted Cosine (normalisation)
- Using conditional probability (unary ratings)

items (i)

users (u)

	i1	i2	i3	
	1	2		
	5		1	3
			2	
	2	2		
	1		4	4
			6	
			3	5
	3		4	
	1	2		2
	2	2	5	5
	4	4	1	4

COLLABORATIVE FILTERING (ITEM-BASED)

Using items ratings vector correlation:

Items 1 & 2 have positive correlation

	i1	i2
	1	1
	2	2
	1	2
	2	2
	4	4

Items 1 & 3 have negative correlation

	i1	i3
	5	1
	2	5
	1	4
	2	5
	4	1

items (i)

users (u)

	i1	i2	i3	
	1	2		
	5		1	3
			2	
	2	2		
	1		4	4
			6	
			3	5
	3		4	
	1	2		2
	2	2	5	5
	4	4	1	4

COLLABORATIVE FILTERING (ITEM-BASED)

2. **Predict user-item score**, using the items similarity calculated

   - Use of weighted average
   - Use of linear regression

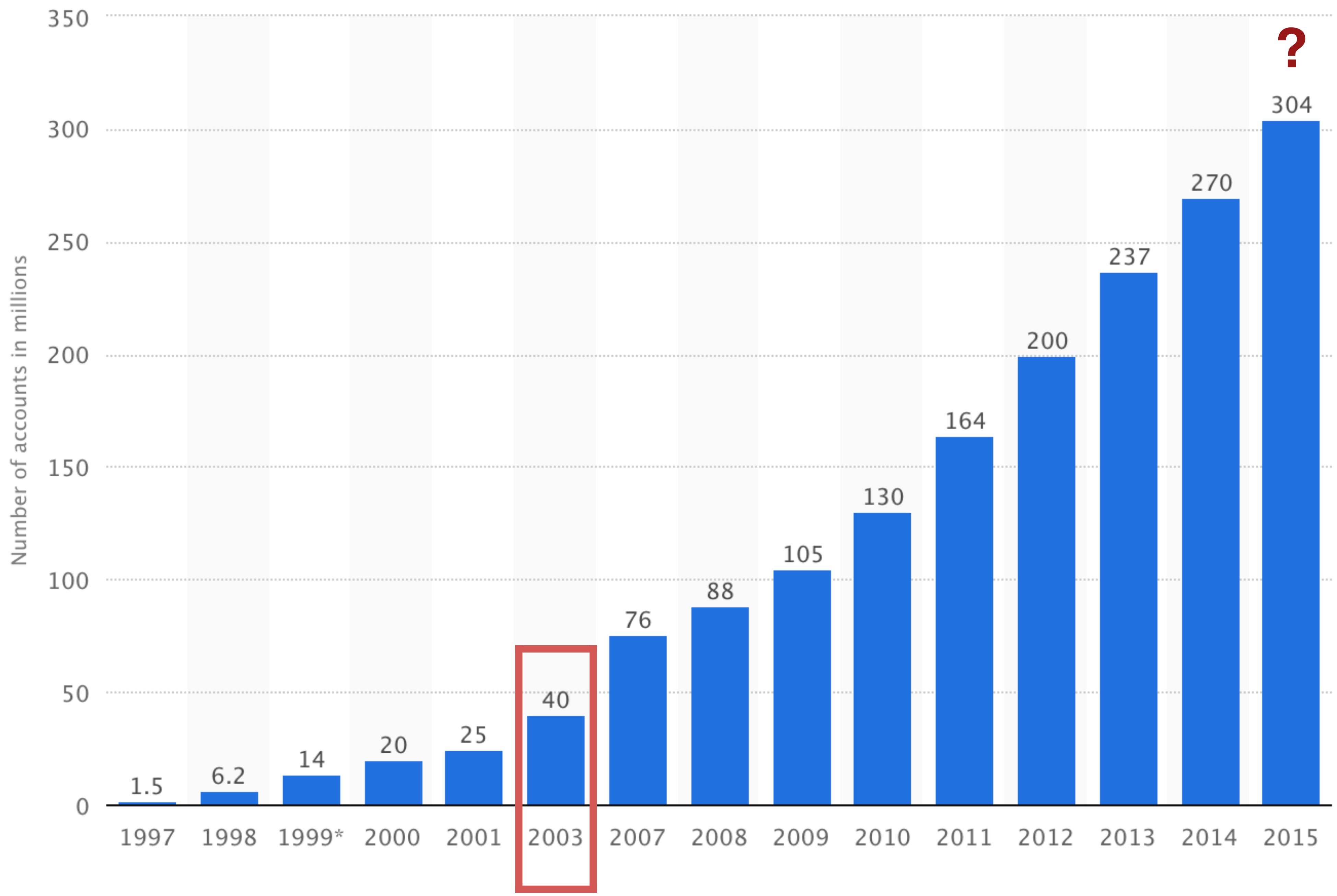
items (i)

users (u)

	i1	i2	i3	
	1	2		
	5		1	3
			2	
a	2	2	?	
	1		4	4
			6	
			3	5
	3		4	
	1	2		2
	2	2	5	5
	4	4	1	4

COLLABORATIVE FILTERING (ITEM-BASED)

ITEM-ITEM CF IN AMAZON



https://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf

COLLABORATIVE FILTERING (ITEM-BASED)

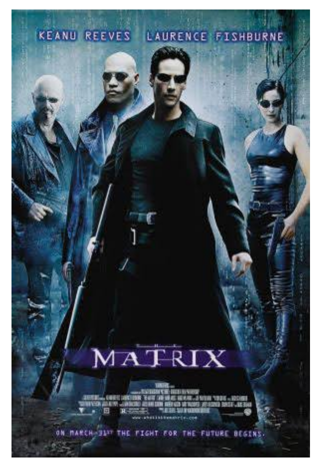
Question!

What could be the main downside of ITEM-ITEM RECs?

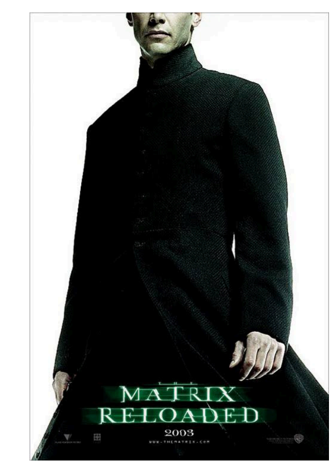
COLLABORATIVE FILTERING (ITEM-BASED)

Item-based CF downside: Too obvious recommendations.

If you liked :



And this item is similar (having similar ratings) to:



You might like:

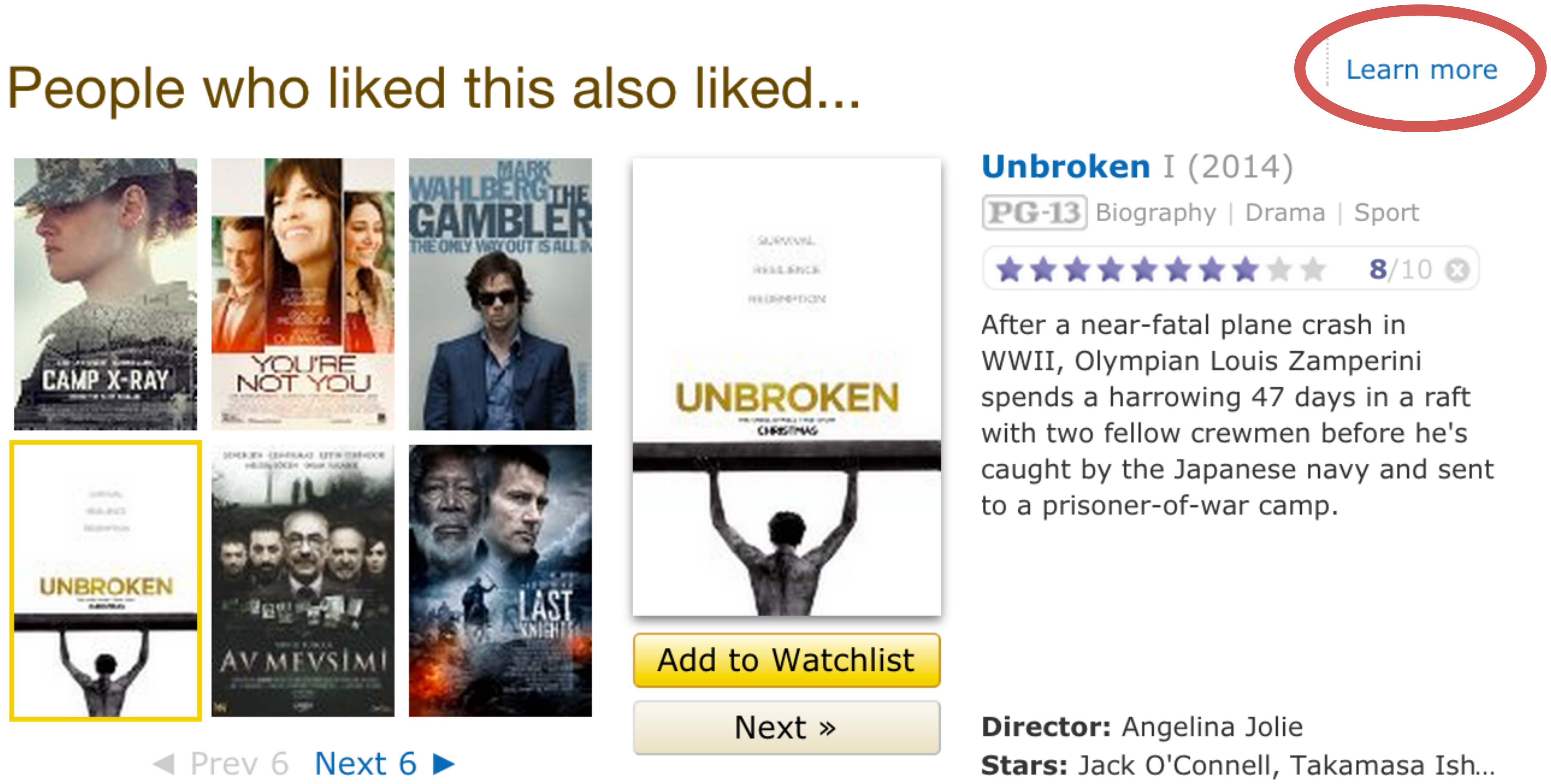
ie business school

BONUS: THE USER INTERFACE

Explaining Recommendations

EXPLAINING RECOMMENDATIONS

Building *Transparency and Trust giving options to understand the recommendations*



EXPLAINING RECOMMENDATIONS

Using non-statistical terms to explain the recommendation methods

IMDb | Find Movies, TV shows, Celebrities and more... | All | IMDbPro | IMDb Apps | Help

Movies, TV & Showtimes | Celebs, Events & Photos | News & Community | Watchlist | Ivan Tarradellas

Help | Search Help | Search | Current site issues

Personalized Recommendations Frequently Asked Questions

non-statistical terms

IMDb makes personalized recommendations to help you discover movies and TV shows that you will love.

Where can I find my Watchlist and my ratings?
Find your Watchlist using the 'Your Watchlist' link in the top right of the nav bar. Find your ratings by using the 'Your Ratings' menu item under the 'Community' menu in the nav bar.

How does IMDb choose personalized recommendations?
First, we take all of the movies and TV shows that you've either rated or added to your Watchlist. Then, we compare your data to ratings made by other users. We can then find movies and TV shows that people with similar tastes to you. For each recommendation, you can see a list of the movies or TV shows upon which the recommendation was based. You have either rated these titles highly, or added them to your Watchlist.

How does IMDb know what I "showed interest in"?
When you give a movie a positive rating or add a movie to your Watchlist, we track that as a movie that you are interested in.

EXPLAINING RECOMMENDATIONS

Using non-statistical terms to explain the recommendation methods

how final rating is made

EXPLAINING RECOMMENDATIONS

Calibration. Helping users understand what it means ie 5,4…1 starts rating

EXPLAINING RECOMMENDATIONS

Allowing manual improvements from the users to fine-tune the engine

Your Watchlist is empty

Use Your Watchlist to track movies and TV shows that interest you.

When you see a title you would like to add, click the ribbon on the poster OR click the "Add to Watchlist" button.

Explore these great titles to add to your list

IMDb Top 250 »
Best Picture Winners »
Top Movies by Genres »
Popular TV Series »

business school

EXPLAINING RECOMMENDATIONS

Manual user identification to set profiles with more precision

THANKS!