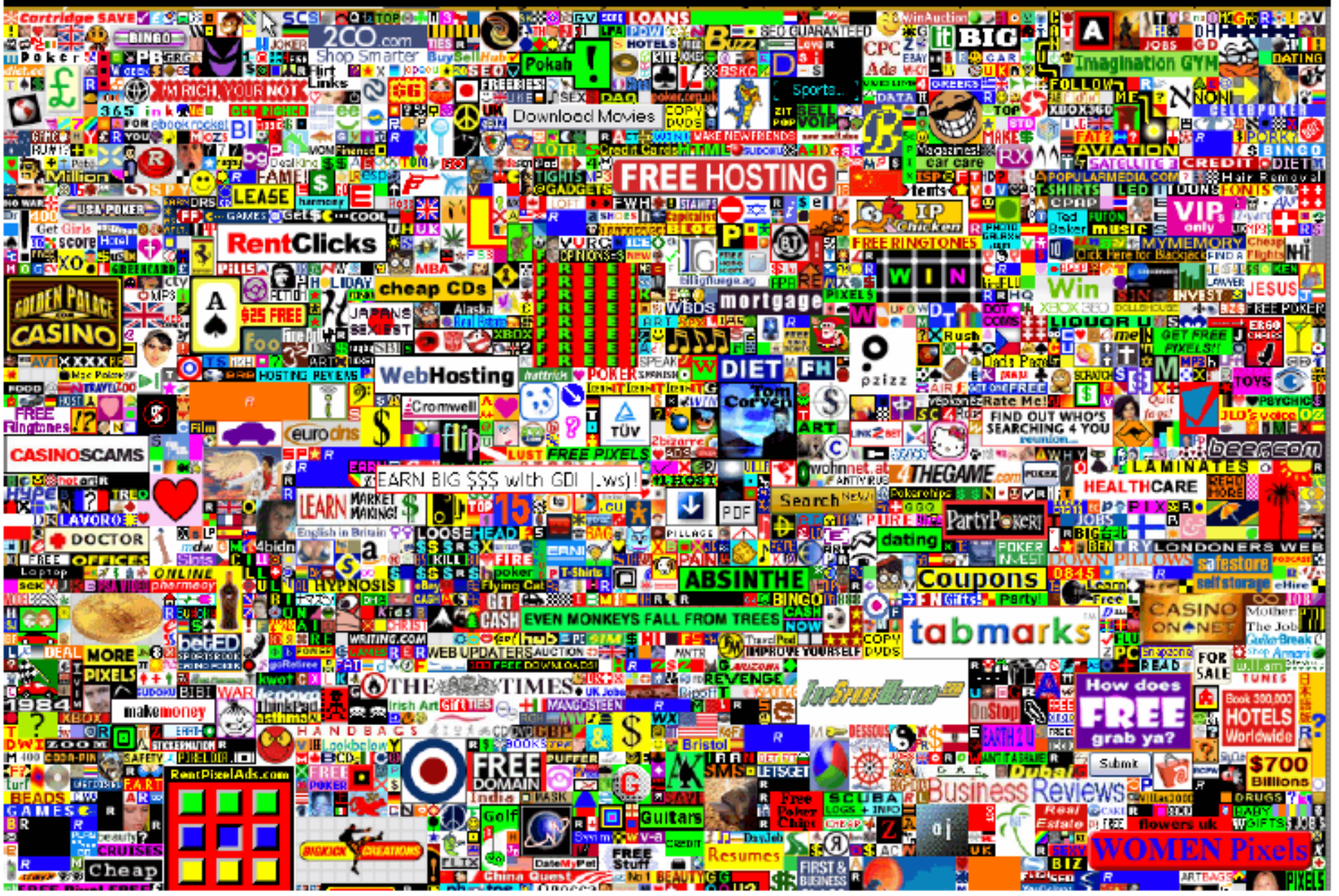
Recommendation Systems

Esther Pacitti

Master 2 – Informatique

Partage de Données a Grande Echelle

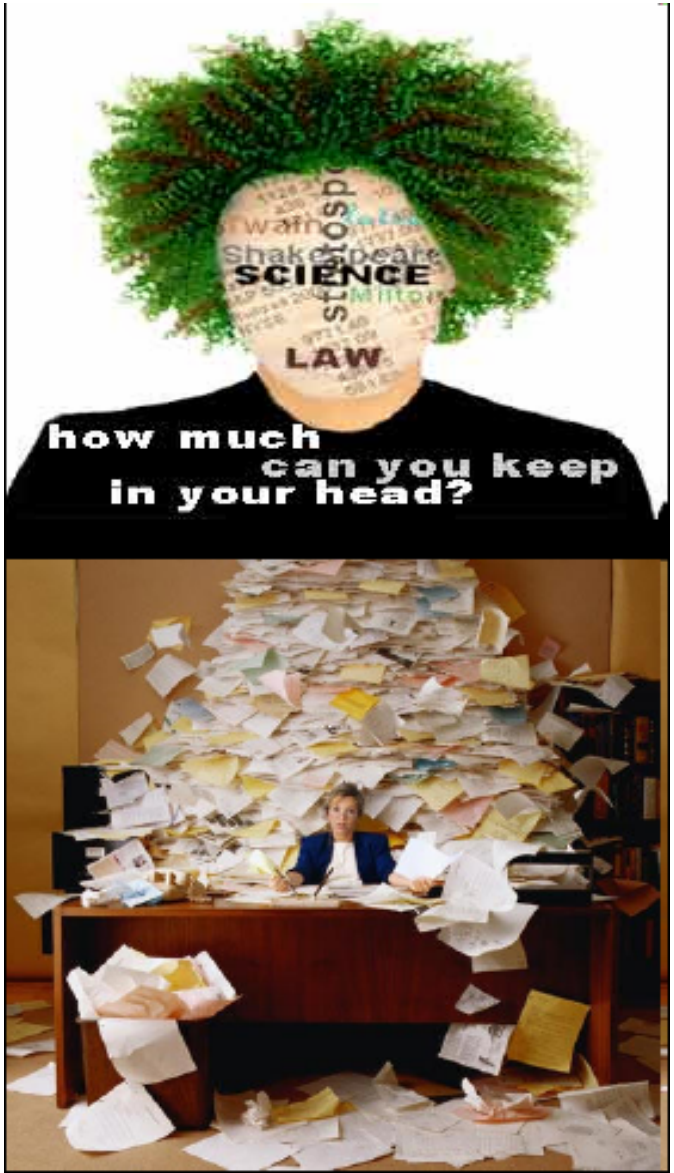
The world is an over-crowded place



They all want to get our attention

Information Deluge
We are OverLoaded !

- Thousands of news articlesand blog posts each day
- Millions of movies, books and music tracks online
- In Hanoi, > 50 TV channels, thousands of programs each day
- In New York, severalthousands of ad messages
- sent to us per day



But we really need and consume only a few of them!


Help me!

Can Google Help ?

- Yes, but only when we really know what we are looking for
- What if I just want some interesting music tracks?
- Btw, what does it mean by “interesting”?

Can Facebook Help ?

- Yes, I tend to find my friends' stuffs interesting
- What if I had only few friends, and what they like do not always attract me?

Can experts help?

- Yes, but it won't scale well
  - Everyone receives exactly the same advice!
- It is what they like, not me!
  - Like movies, what gets expert approval does not guarantee attention of the mass

Idea: Recommendation Systems

- To recommend to us something we may like
  - It may not be popular
  - The world is long-tailed
- How?

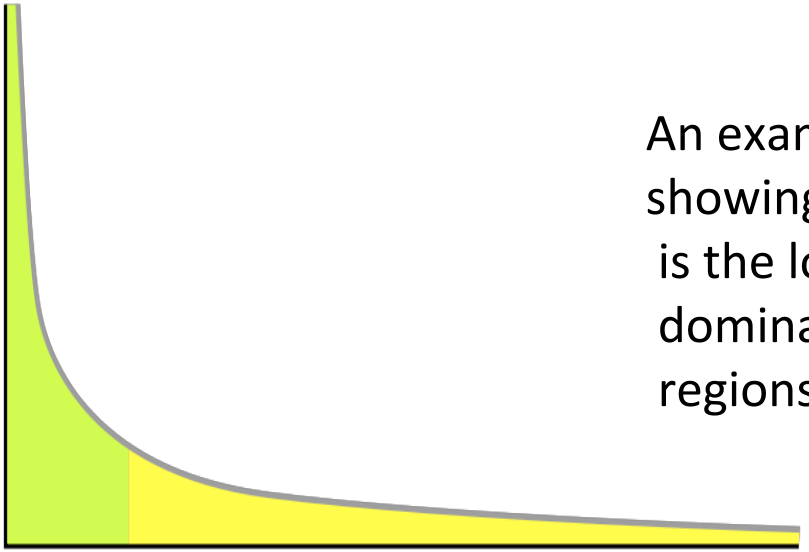
Based on our history of using services

Based on other people like us

Ever heard of “collective intelligence”?

Long Tail

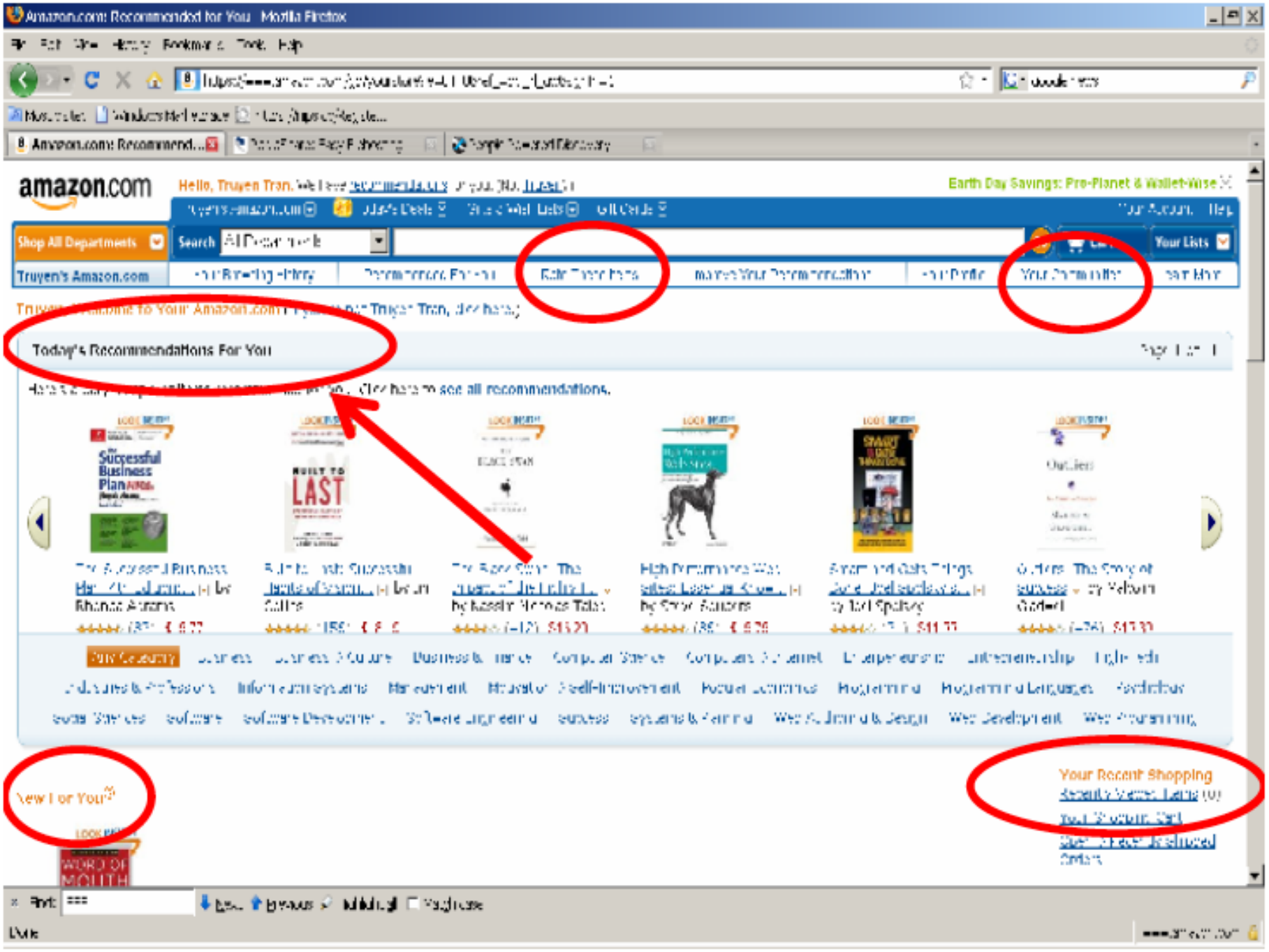
« In statistics, a long tail of some distributions of numbers is the portion of the distribution having a large number of occurrences far from the "head" or central part of the distribution.
The distribution could involve popularities, random numbers of occurrences of events with various probabilities, etc. »



An example of a power law graph showing popularity ranking. To the right (yellow) is the long tail; to the left (green) are the few that dominate. In this example, the areas of both regions are equal.

Ever heard of

- GroupLens?
- Amazon recommendation?
- Netflix Cinematch?
- Google News personalization?
- Netflix Prize $1mil challenge?
- Strands?
- TiVo?
- Findory?


amazon.com
Today's Recommendations For You
New For You

Some Intresting Facts

- Netflix:
  - 2/3 rented movies are from recommendation
- Google News
  - 38% more click-through are due to recommendation
- Amazon
  - 35% sales are from recommendation

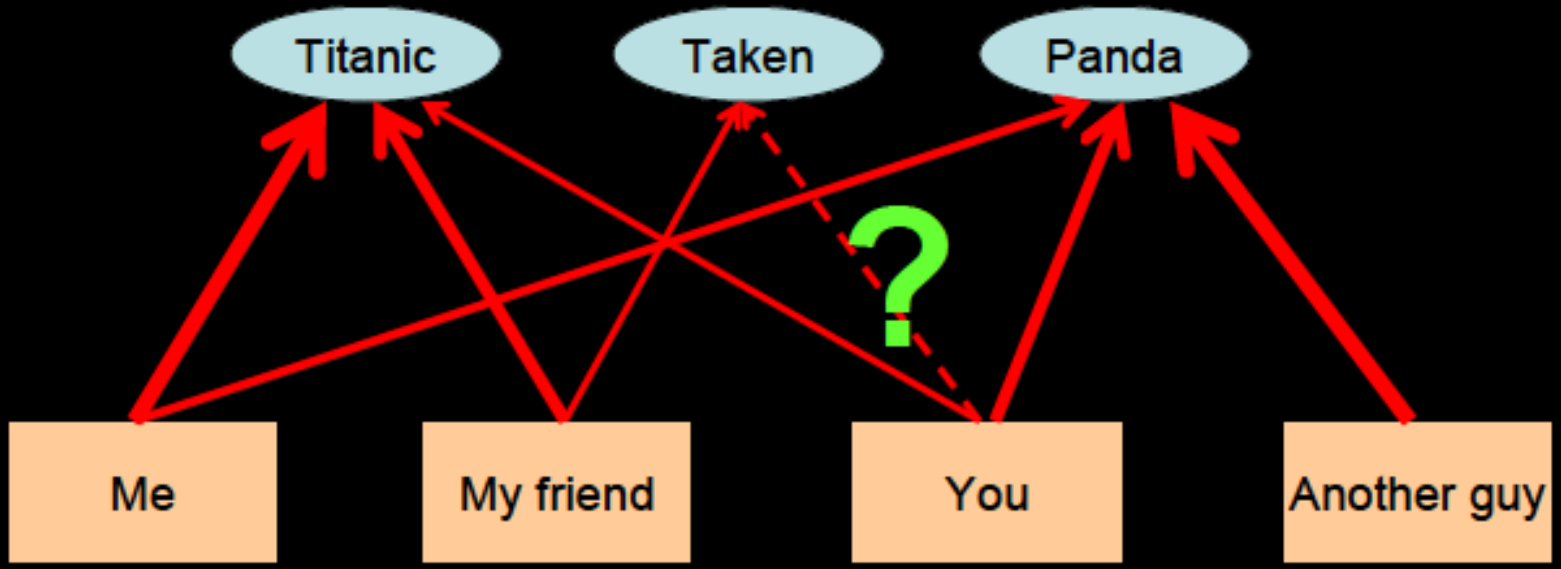
What can be Recommended ?

- Advertising messages
- Investment choices
- Restaurants
- Cafes
- Music tracks
- Movies
- TV programs
- Books
- Cloths
- Supermarket goods
- Tags
- News articles
- Online mates (Dating services)
- Future friends (Social network sites)
- Courses in e-learning
- Drug components
- Research papers
- Citations
- Code modules
- Programmers

What do Recommender Systems ?

1. Predict how much you may like a certain product/service
2. Compose a list of **N** best items for you
3. Compose a list of **N** best users for a certain product/ service
4. Explain to you why these items are recommended to you
5. Adjust the prediction and recommendation based on your feedback and other people

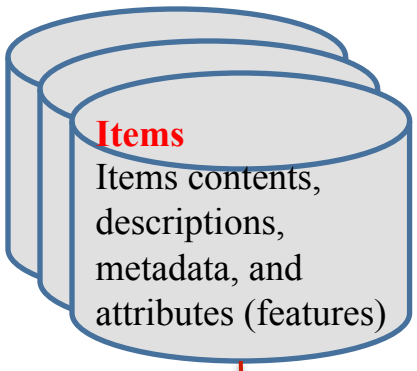
Graph representation

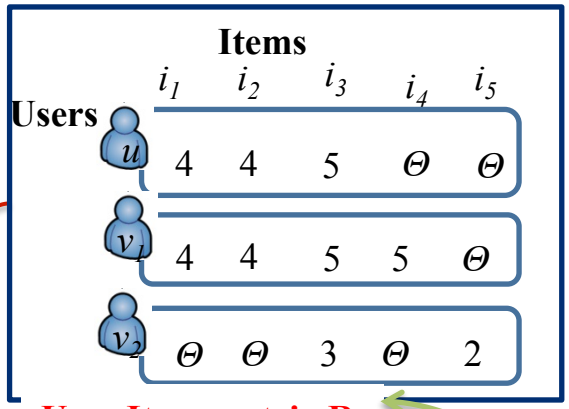


Recommendation Systems Definition

- **Recommender systems** or **recommendation systems** (sometimes replacing "system" with a synonym such as platform or engine) are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that user would give to an item (such as music, books, or movies) or social element (e.g. people or groups) they had not yet considered, using a model built from the characteristics of an item (content-based approaches) or the user's social environment (collaborative filtering approaches). (from Wikipedia)

Recommendation Systems





User-Item matrix R

	u	v_1	v_2
u	5	4	?
v_1	4	5	3.5
v_2	?	3.5	5

Trust matrix

Recommender

- Collects *u's* profile
- Find other users with similar patterns or items
- Use the data from those similar users to generate advice for *u*

Users' patterns
Info. about users such as name, age, rated items, items feature, friends, trust, etc.

Aggregate user's patterns either explicitly or implicitly

Recommendations

potential user *u*

Preference Predictions

- Collaborative filtering: Suggests to the user contents that have been seen and rated by users which have similar affinity with the user
  - User-based method
  - Matrix Factorization
  - etc…
- Content-based filtering: Suggest to the user contents that are similar to contents that it has seen or rated
- Social-based filtering: Suggest to the user contents that have been rated by its friends, or trustful users

Collborative Filtering

- User-based method (1994, GroupLens)
  - Many people liked “Kungfu Panda”
  - Can you tell how much I like it?
  - The idea is to pick about **20-50** people who share **similar** **taste** with me, then how much I like depend on how muchTHEY liked.
- – In short: you **may** like it
- because your “friends” liked it.

user \ item	1	2	3	4	5	6	7	8
1	5	4	5		3			4
2		3	5			4		5
3		4		5	4			
4	5		4	5		3	5	
5	4				3	3		4
6	5	2			3	5		
7			1	4	2			
8				5			4	3

Content-Based Filtering

- Suggest to the user contents that are similar to contents that it has seen or rated
- Users Profile
  - Items he has seen
- Method:
  - Compute similarity (e.g. cosinus) between items
  - Query items that are similar to a given item
  - Match item's content and user's profile

Problems with Collaborative Filtering

- Scale
  – Netflix (2007): 5M users, 50K movies, 1.4B rating

- Sparse data
  – I have rated only one book at Amazon!
- Cold-Start
  – New users and items do not have history
- Popularity bias
  – Everyone reads “Harry Potter”

Top-N recommendation

1) Find similar users, collect the items they like

2) Filter out those items the user u has rated

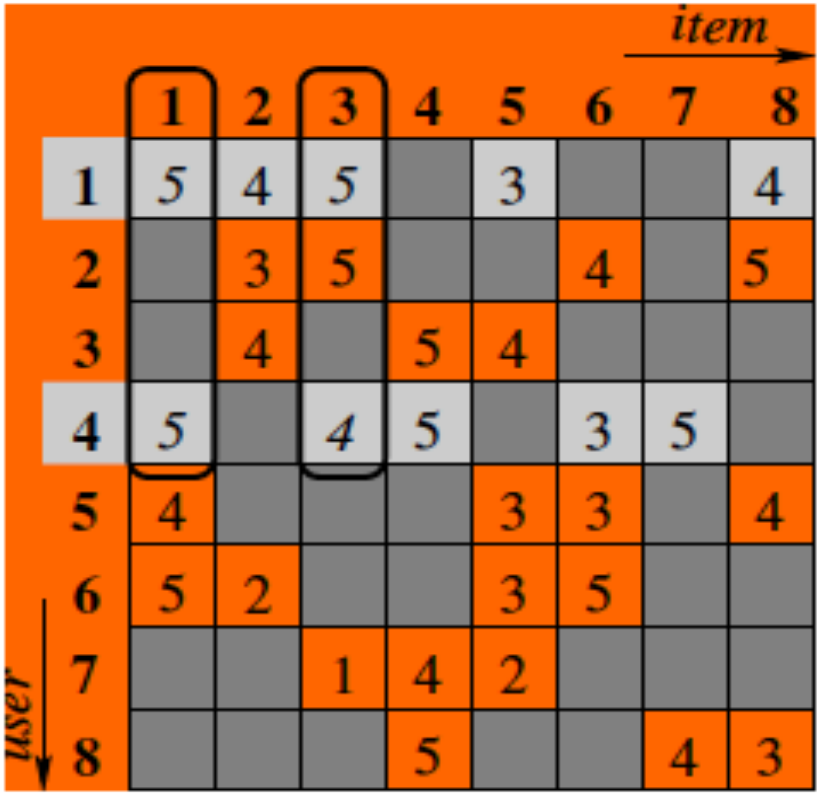
3) Rank the remaining items by considering

-The number of times each item is liked by those similar users

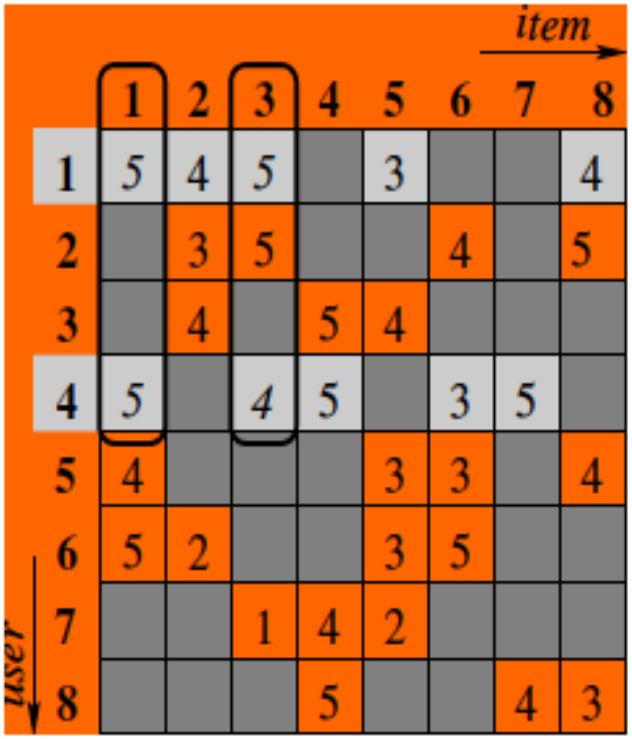
- The popularity of the item

- The associated ratings (how many users ranked in the whole matrix)

- Switching the role of item to user, we may have top-N user list



Example



Suppose we want to recommend items to user 4

1) Similar users = 1
   Items they like: 1, 2, 3, 5, 8 (all rated items)

2) Find (filter) the items not rated by user 4:

   1, 2, 3, 5, 8 - 1 3 4 6 7 = 2, 5, 8

3)

Items	Like (among similars)	No. Of associated Rate (in the corpus)	Popularity (good rates)	Final Score
2	1	4	2	2,2
5				
8				

Example

4) Define a weight of each parameter

Scoring Function = 0.4*Like + 0.3*rate + 0.3*Pop = 2,2

5) Compute the score of each item

6) Rank the items

7) Chose the Top-n

Exercice: Complete de table

Rank the items

Explanation

- This is a current hit …
- More on this artist …
- Try something from similar artists …
- Someone similar to you also like this …
- As you listened to that, you may want this …
- These two go together …
- This is most popular in your group …
- This is highly rated …
- Try something new …

Explanation

- Examples from Strands.com
  - Welcom back (recently viewed)
  - For you today
  - New for you
  - Hot / Most popular of this type
  - Other people also do this …
  - Similar or related products
  - Complementary accessories
  - This goes with this …
  - Gift idea
  - Shopping assisant

Task 5: Online updating

- New items and users come each hour or minute
- The two worlds:
  - Most songs and books are still interesting for a long time (the tail is really long)
  - Most news articles are read on the day and forgotten next day

    But tracking back is useful to follow an event or scandal
- Consider this effect in the recommendation process.