Interaction-Based Recommender Systems

Mining Massive Datasets

Prof. Carlos Castillo

Topic 17

Universitat Pompeu Fabra Barcelona

Sources

- Data Mining, The Textbook (2015) by Charu Aggarwal (Section 18.5) – slides by Lijun Zhang
- Mining of Massive Datasets 2nd edition (2014) by Leskovec et al. (Chapter 9) - slides A, B

Interaction-based recommendations

Missing-value estimation/completion

- The matrix is extremely large and sparse

$$M = \begin{bmatrix} \blacksquare & & \blacksquare & & & \blacksquare & \\ & \blacksquare & & \blacksquare & \blacksquare & & \blacksquare \\ \blacksquare & & & \blacksquare & & \blacksquare & \\ & & \blacksquare & & \blacksquare & & \\ & \blacksquare & & \blacksquare & & & \blacksquare \end{bmatrix} \in \mathbb{R}^{n \times d}$$

Only black squares have non-zero values.

Types of algorithms

- Neighborhood-Based Methods
  - User-Based or Item-Based Similarity with Ratings
- Graph-Based Methods
- Clustering Methods
  - Adapting k-Means Clustering or Adapting Co-Clustering
- Latent Factor Models
  - Matrix Factorization, e.g., Singular Value Decomposition

User-based similarity with ratings

- Let $I_{u,v}$ be common ratings between two users
- Similarity: Pearson correlation coefficient

$$\mathrm{sim}(u,v) = \frac{\sum_{i \in I_{u,v}} (u_i - \hat{u}) \cdot (v_i - \hat{v})}{\sqrt{\sum_{i \in I_{u,v}} (u_i - \hat{u})^2 \cdot \sum_{i \in I_{u,v}} (v_i - \hat{v})^2}}$$

$$\hat{u} = \frac{1}{|u|} \sum_{i=1}^{|u|} u_i \qquad \hat{v} = \frac{1}{|v|} \sum_{i=1}^{|v|} v_i$$

Note: averages are taken over all elements, not only ones in common

User-based similarity with ratings (cont.)

$$\mathrm{sim}(u,v) = \frac{\sum_{i \in I_{u,v}} (u_i - \hat{u}) \cdot (v_i - \hat{v})}{\sqrt{\sum_{i \in I_{u,v}} (u_i - \hat{u})^2 \cdot \sum_{i \in I_{u,v}} (v_i - \hat{v})^2}}$$

- Score of recommendation

$$\mathrm{score}(u,i) = \hat{u} + \frac{\sum_{v : v_i \neq \mathrm{NULL}} \mathrm{sim}(v,u) \cdot (v_i - \hat{v})}{\sum_{v : I_{u,v} \neq \emptyset} |\,\mathrm{sim}(v,u)|}$$

Note: for efficiency one can take only the most similar users

Exercise

u

2			4	5	
5		4			1
		5		2	
	1		5		4
		4			2
4	5		1		

Answer in
Google Spreadsheet

In the spreadsheet, complete all yellow cells:

1. The computation of sim(u,v)
2. The rating of all the movies that user u has not seen yet

Which movie is recommended?

$$\mathrm{sim}(u,v) = \frac{\sum_{i \in I_{u,v}} (u_i - \hat{u}) \cdot (v_i - \hat{v})}{\sqrt{\sum_{i \in I_{u,v}} (u_i - \hat{u})^2 \cdot \sum_{i \in I_{u,v}} (v_i - \hat{v})^2}}$$

$$\mathrm{score}(u,i) = \hat{u} + \frac{\sum_{v: v_i \neq \mathrm{NULL}} \mathrm{sim}(v,u) \cdot (v_i - \hat{v})}{\sum_{v: I_{u,v} \neq \emptyset} |\mathrm{sim}(v,u)|}$$

You can do the same with items!

- Item-based similarities with ratings

$$\mathrm{sim}(i,j) = \frac{\sum_{u \in I_{i,j}} (u_i - \hat{i}) \cdot (u_j - \hat{j})}{\sqrt{\sum_{u \in I_{i,j}} (u_i - \hat{i})^2 \cdot \sum_{u \in I_{i,j}} (u_j - \hat{j})^2}}$$

- Item-based recommendations

$$\mathrm{score}(u,i) = \hat{i} + \frac{\sum_{j: u_j \neq \mathrm{NULL}} \mathrm{sim}(i,j) \cdot (u_j - \hat{j})}{\sum_{j: I_{i,j} \neq \emptyset} |\mathrm{sim}(i,j)|}$$

(Do it at home)

2			4	5	
5		4			1
		5		2	
	1		5		4
		4			2
4	5		1		

1. Compute avg(j) for all items
2. Compute sim(i,j) for all items for which there is some intersection with i
3. Compute score(u,i) for all users who have not seen i yet

$$\mathrm{sim}(i,j) = \frac{\sum_{u \in I_{i,j}} (u_i - \hat{i}) \cdot (u_j - \hat{j})}{\sqrt{\sum_{u \in I_{i,j}} (u_i - \hat{i})^2 \cdot \sum_{u \in I_{i,j}} (u_j - \hat{j})^2}}$$

$$\mathrm{score}(u,i) = \hat{i} + \frac{\sum_{j : u_j \neq \mathrm{NULL}} \mathrm{sim}(i,j) \cdot (u_j - \hat{j})}{\sum_{j : I_{i,j} \neq \emptyset} |\mathrm{sim}(i,j)|}$$

Exercise from ML for Recommender Systems tutorial by Alex Karatzoglou, 2015.

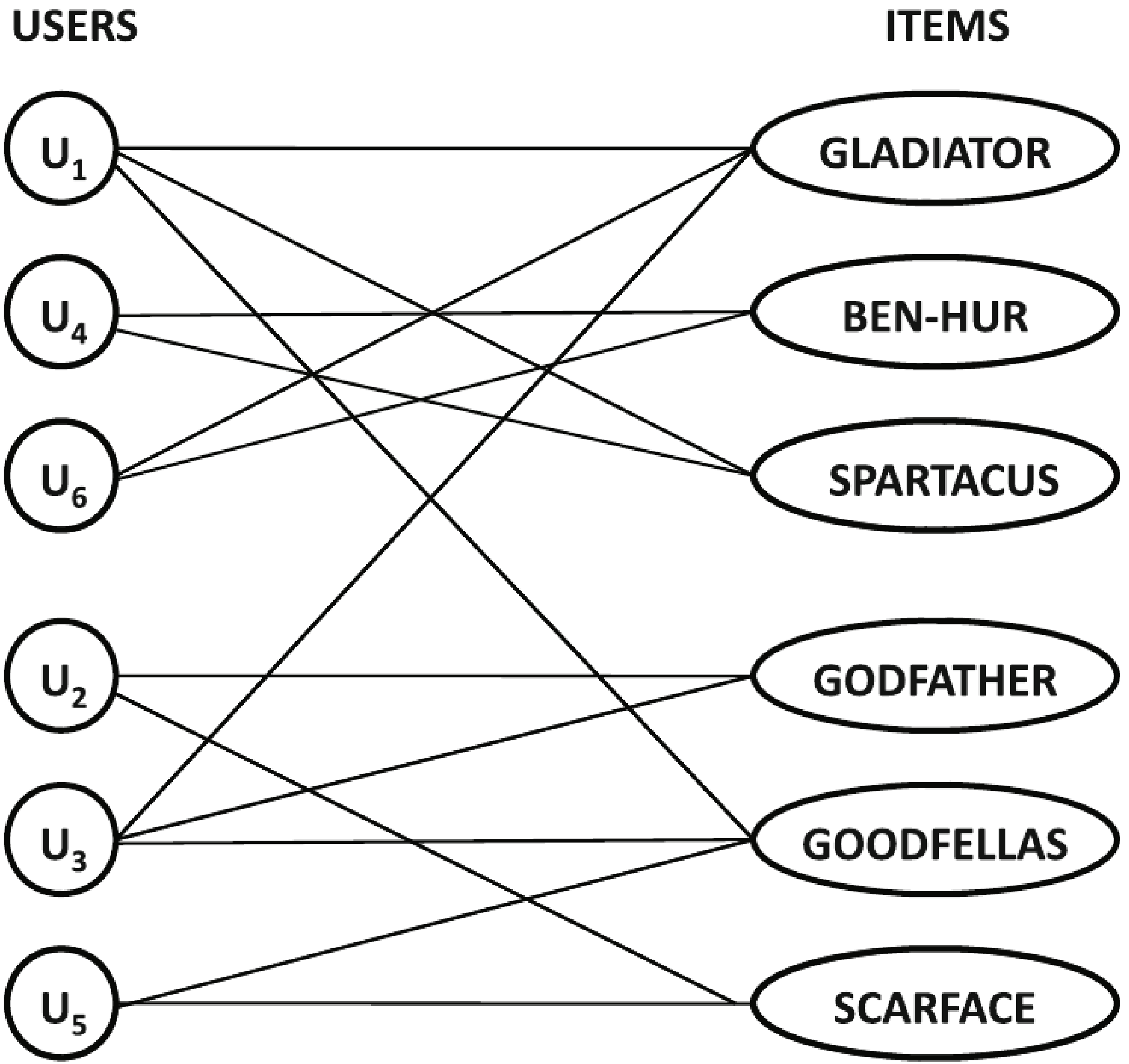
Note

- There are many ways of computing user-based similarity and item-based similarity
- There are many ways of using these to generate recommendations
- The method we have described is aware of the **bias of users**, in the sense of some users being more positive/negative than others in general

Graph- and clustering-based methods

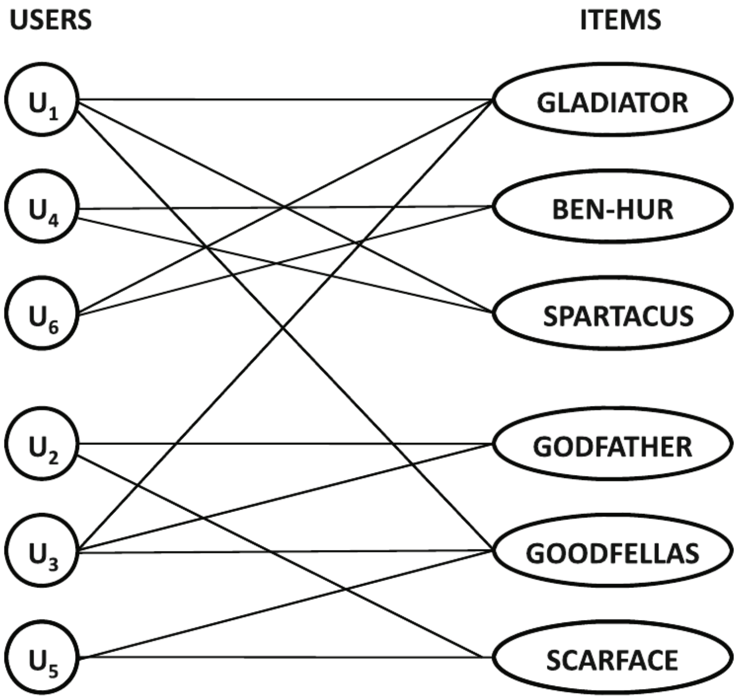
Graph-based methods

- Bipartite user-item graph with nodes N_u U N_i
- N_u users
- N_u items
- Non-zero utility ⇒ edge



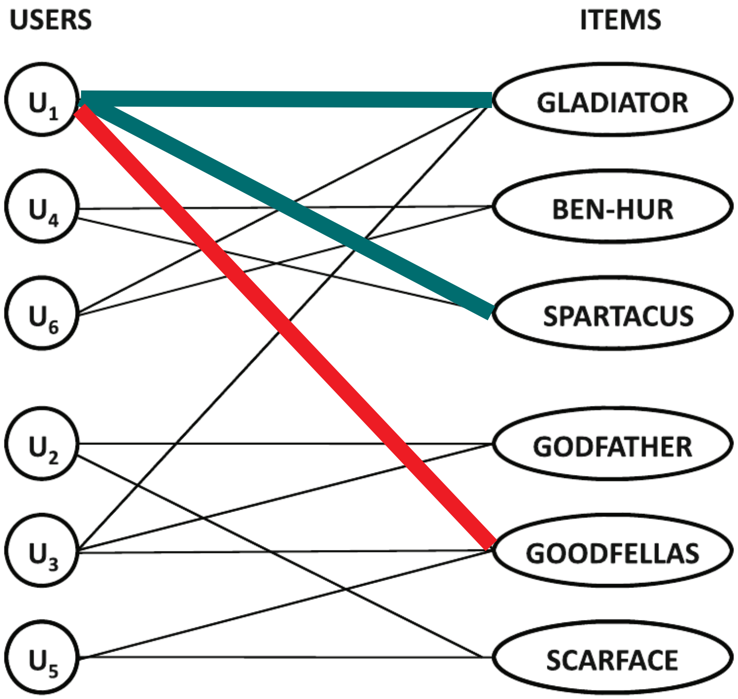
Graph-based methods (cont.)

- Use graph-based methods
  - Random walk with restart to a user or item
  - SimRank
- Low “random jump” probability might favor popular items



Graph-based methods (cont.)

- Signed networks can be used
  - Remember we must interpret ratings with respect to user and item averages
  - Below average rating ⇒ -
  - Above average rating ⇒ +
- Positive link prediction problem



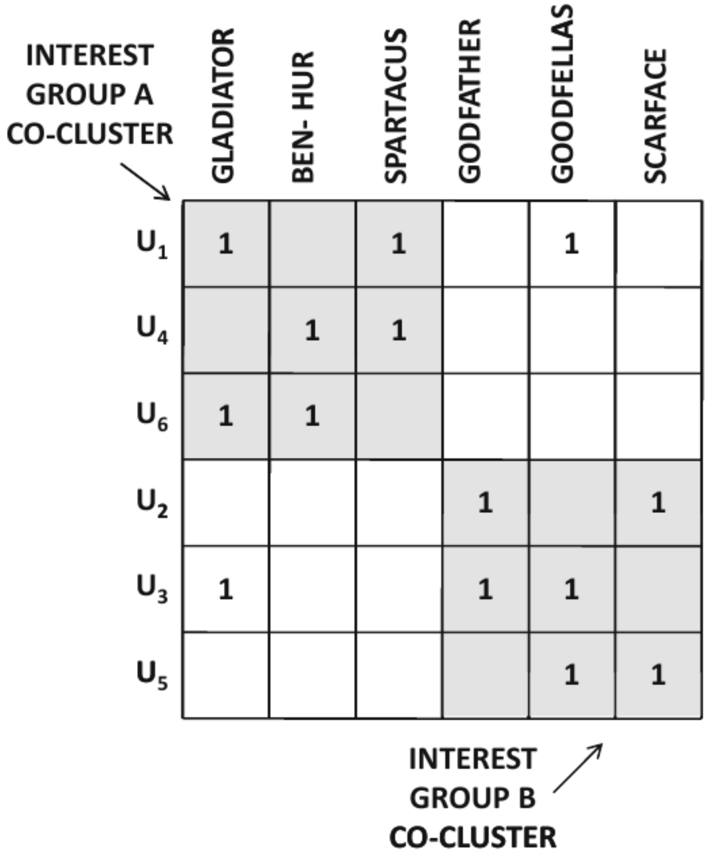
Clustering methods

- Motivations
  - Reduce computational cost
  - To some extent address data sparsity
- Results of clustering
  - Clusters of users for user-user similarity recs.
  - Clusters of items for item-item similarity recs.

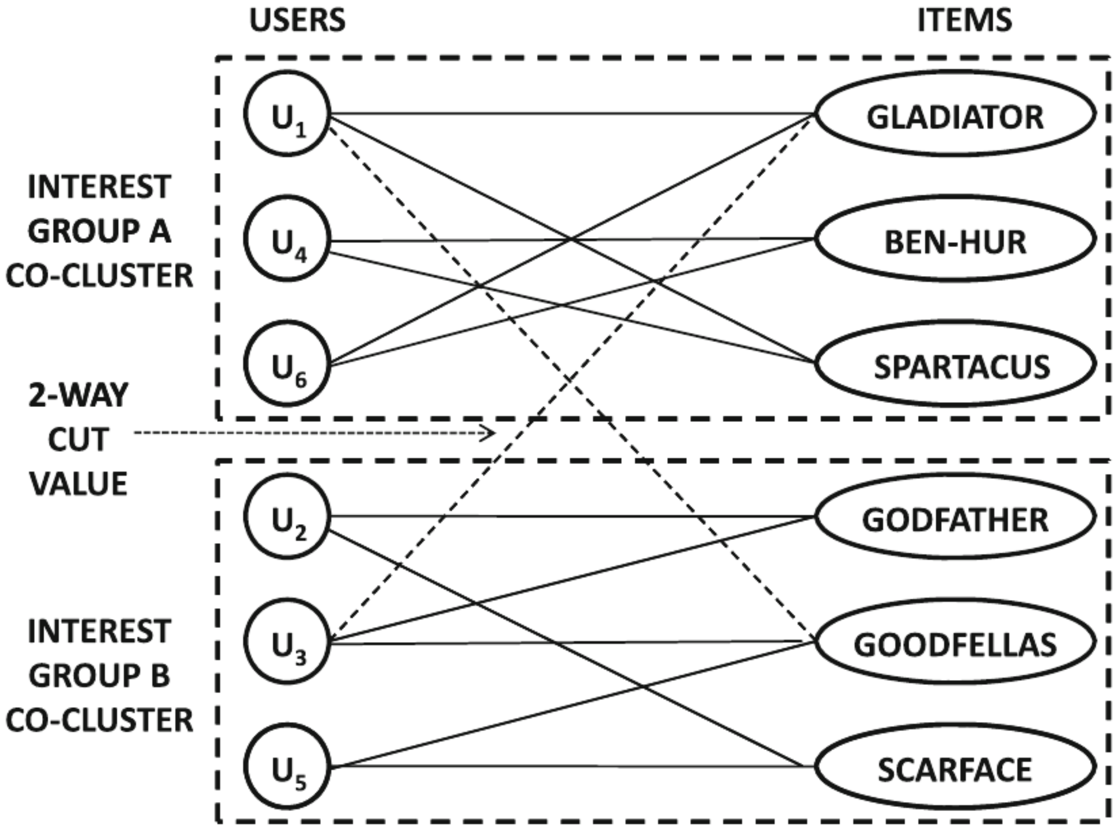
Clustering methods (cont.)

- User-user recommendation approach
  - Cluster users into groups
  - For any user u, compute average normalized rating for each item i the user has not seen
  - Report these ratings for (u,i)
- Same with item-item recommendations
- Neighborhoods will be smaller

Co-Clustering Approach



(a) Co-cluster



(b) User-item graph

Summary

Things to remember

- Interaction-based recommendations
  - User-based
  - Item-based

Exercises for TT16-TT18

- Mining of Massive Datasets 2nd edition (2014) by Leskovec et al. Note that some exercises cover advanced concepts:
  - Exercises 9.2.8
  - Exercises 9.3.4
  - Exercises 9.4.6