Lecture 21:
CS 5306 / INFO 5306:
Crowdsourcing and
Human Computation

Course Projects: Paying for Amazon Mechanical Turk

- Put $20 on account
- Get reimbursed by Information Science

Types of Crowdsourcing

- Overt
  - Collecting (Amazon Reviews)
  - Labor Markets (Amazon Mechanical Turk)
  - Collaborative Decisions (Prediction Markets)
  - Collaborative Creation (Wikipedia)
  - Smartest in the Crowd (Contests)
  - Games with a Purpose
- Covert / Crowd Mining
  - Web page linkage, search logs, social media, collaborative filtering
- Dark side of crowdsourcing and human intelligence
- Collective intelligence in animals

Types of Crowdsourcing

- Overt
  - Collecting (Amazon Reviews)
  - Labor Markets (Amazon Mechanical Turk)
  - Collaborative Decisions (Prediction Markets)
  - Collaborative Creation (Wikipedia)
  - Smartest in the Crowd (Contests)
  - Games with a Purpose
- Covert / Crowd Mining
  - Web page linkage, search logs, social media, collaborative filtering
- Dark side of crowdsourcing and human intelligence
- Collective intelligence in animals

Upcoming Lectures

- Tuesday, April 26: Brian McInnis
  - Worker experience on Amazon Mechanical Turk and what makes good vs bad requesters

- Thursday, April 28:
  - Read "Mammon and the Archer" from *The Four Million* by O. Henry
  - http://americanenglish.state.gov/files/ae/resource_files/mammon-and-the-archer.pdf

Information Filtering

- We face more information online than we can process
- Information filtering:
  - Select and prioritize among all the information of possible relevance to you
  - *Predict* what a person would want to see

“Recommender Systems”

Information Filtering: Prehistory

- Generic recommendations:
  - Bestseller lists
  - “Recent returns” at the library
  - Well-used paths through the wood
- Personalized recommendations:
  - Word of mouth
  - Marketing

Information Filtering Approaches

- Content-based filtering
- Collaborative/”social” filtering
- Hybrid approaches

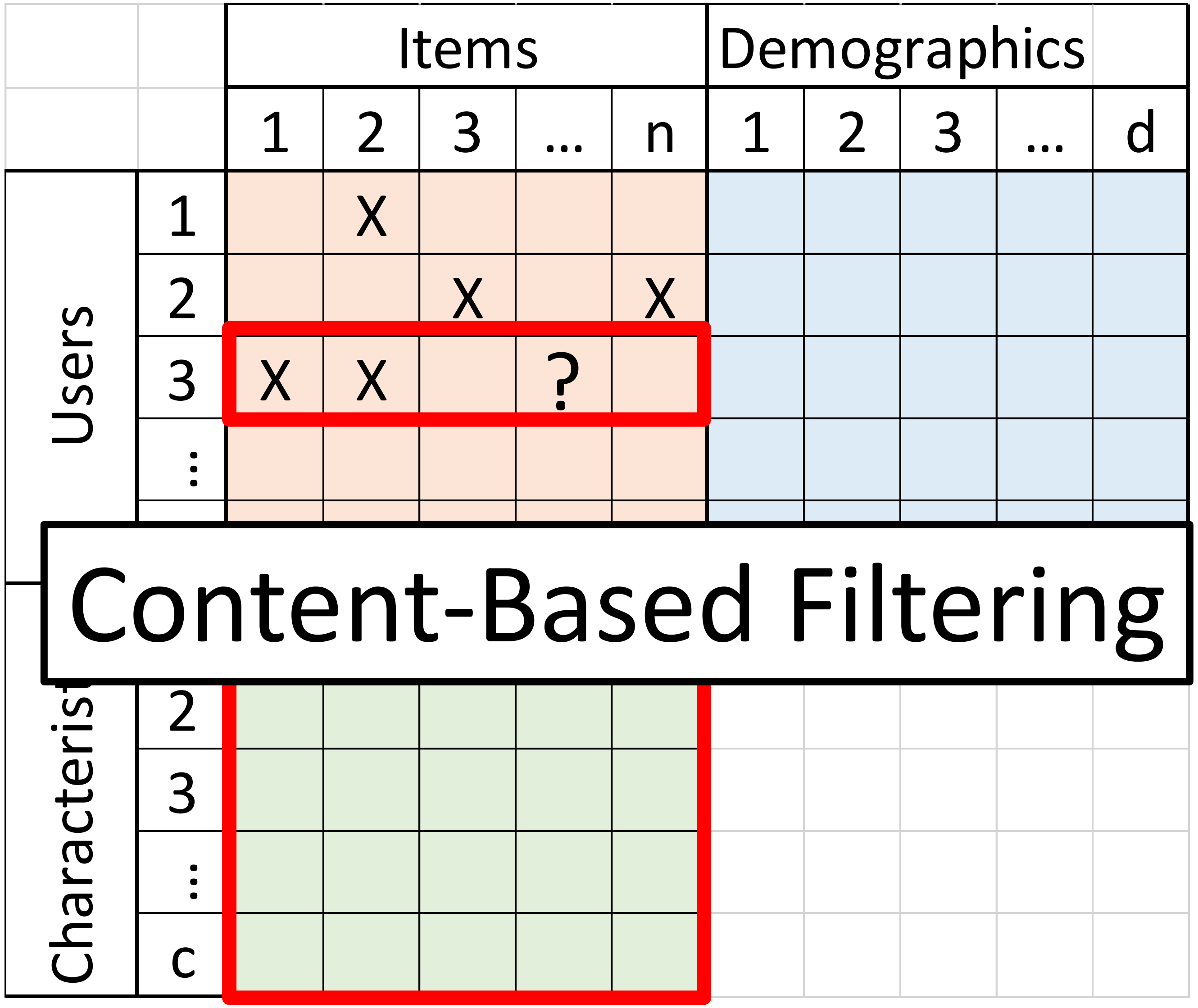
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X								
	…										
	m					X					
Characteristics	1										
	2										
	3										
	…										
	c										

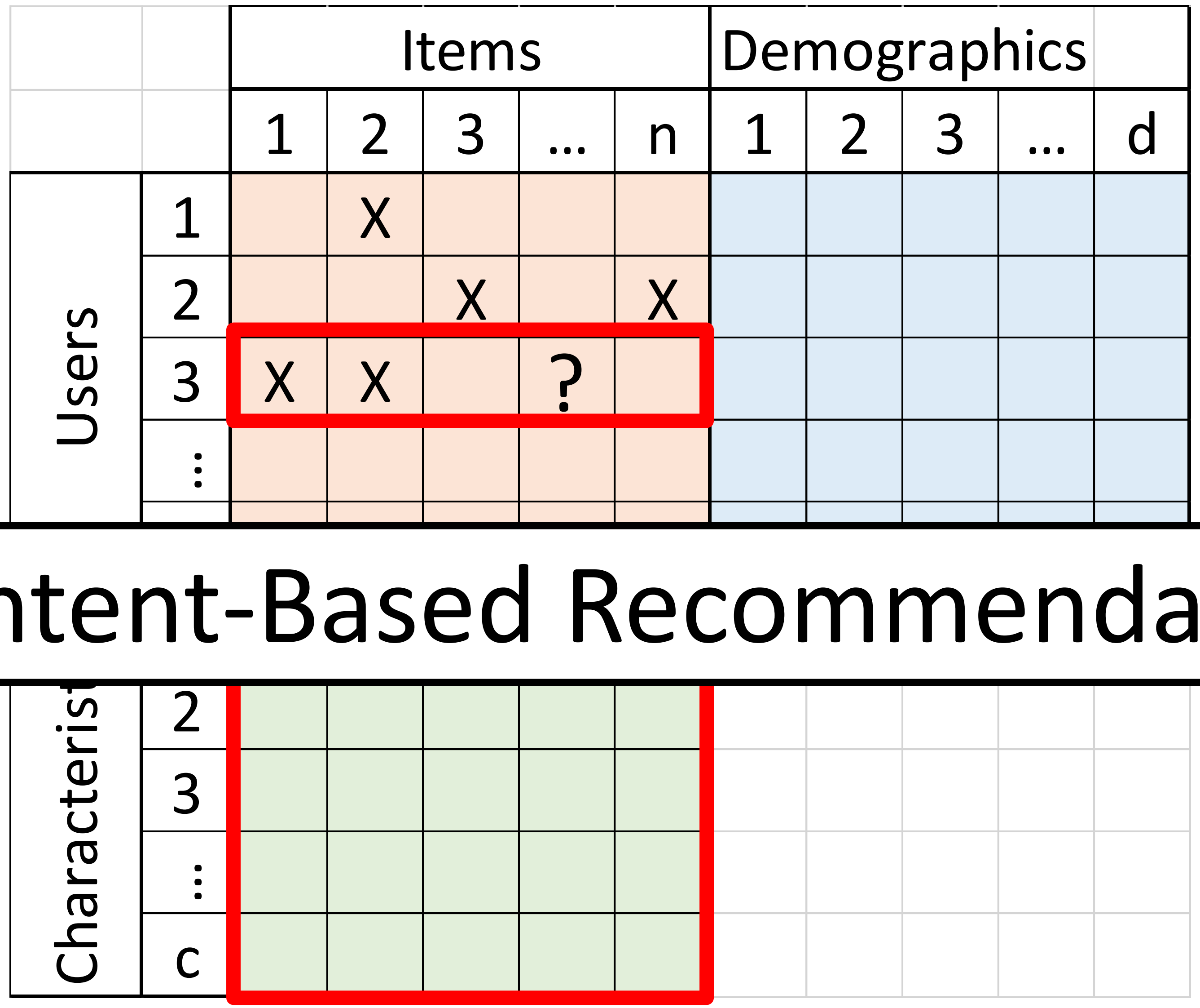
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X								
	⋮										
	m					X					
Characteristics	1										
	2										
	3										
	⋮										
	c										

Information Filtering Framework



Information Filtering Framework



Content-Based Recommendation

Content-based Filtering/Recommendation

- Intuition:
  The things you like share characteristics
- Approach:
  - Inspect the items you like and don’t like
    - Can explicitly ask for such information, or infer it by observation
  - Figure out what’s in the things you like that aren’t in the things you don’t like
  - Recommend other items with those same characteristics
- Example: Spam detection

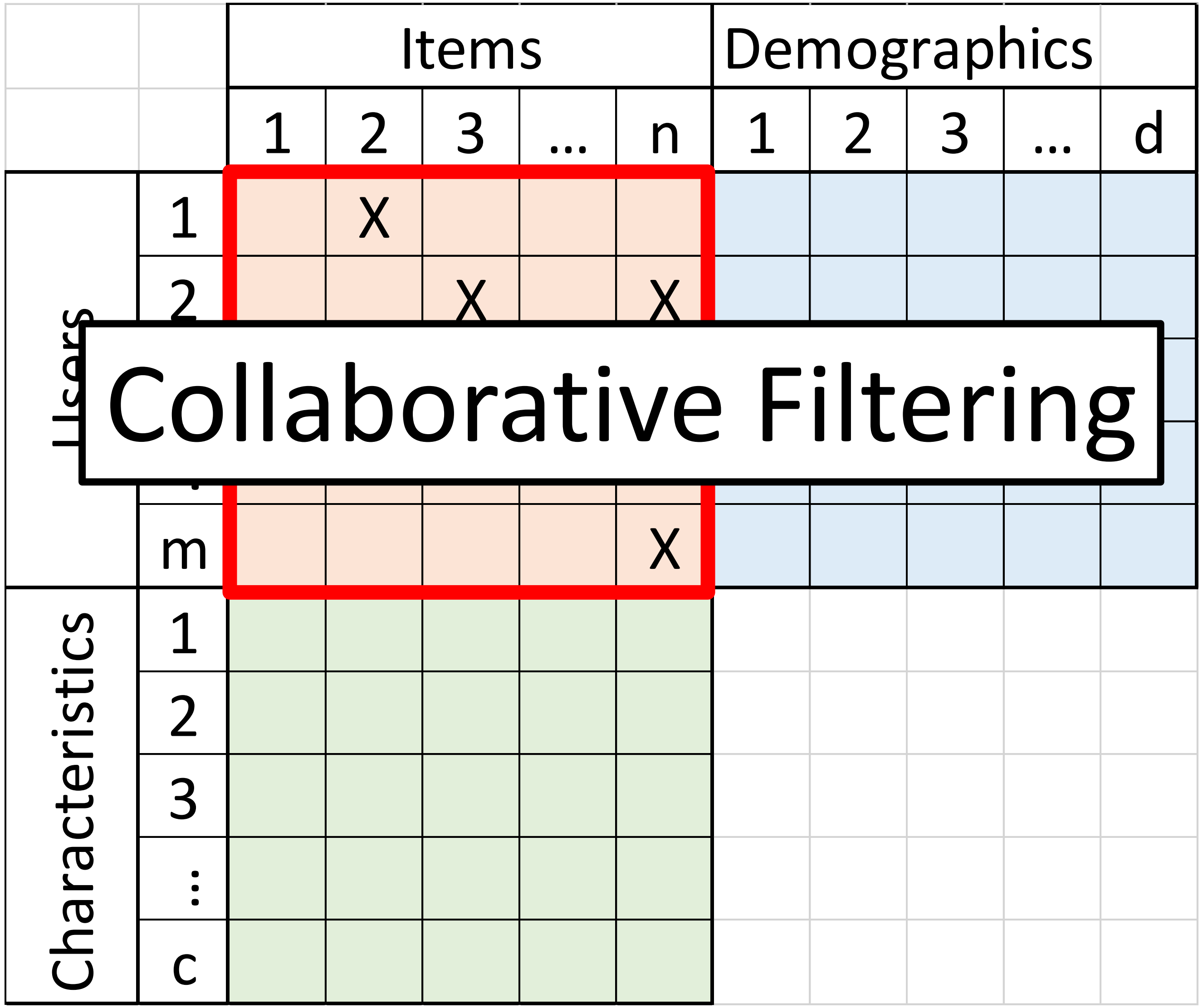
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X								
	…										
	m					X					
Characteristics	1										
	2										
	3										
	…										
	c										

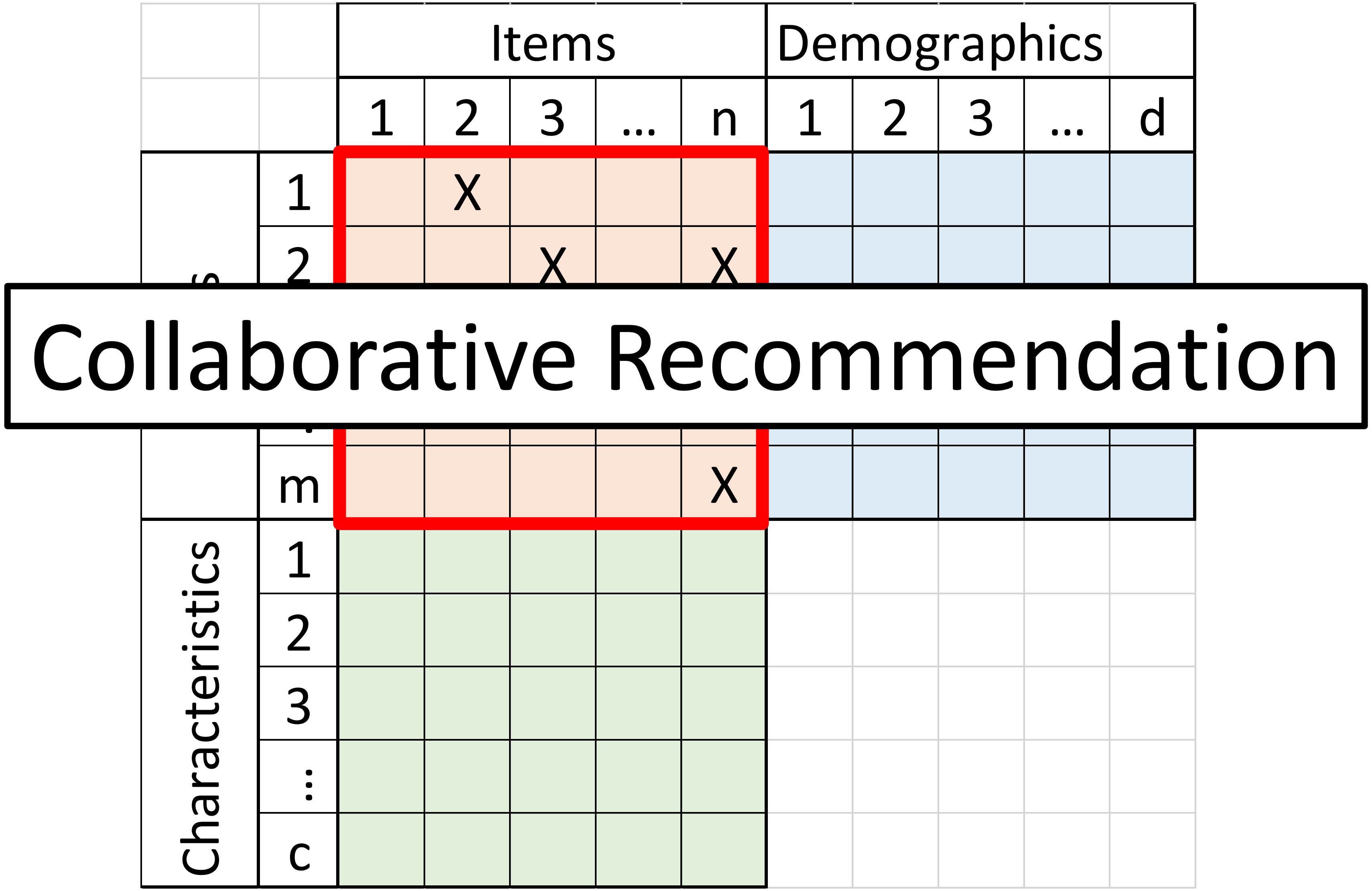
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X		?						
	⋮										
	m					X					
Characteristics	1										
	2										
	3										
	⋮										
	c										

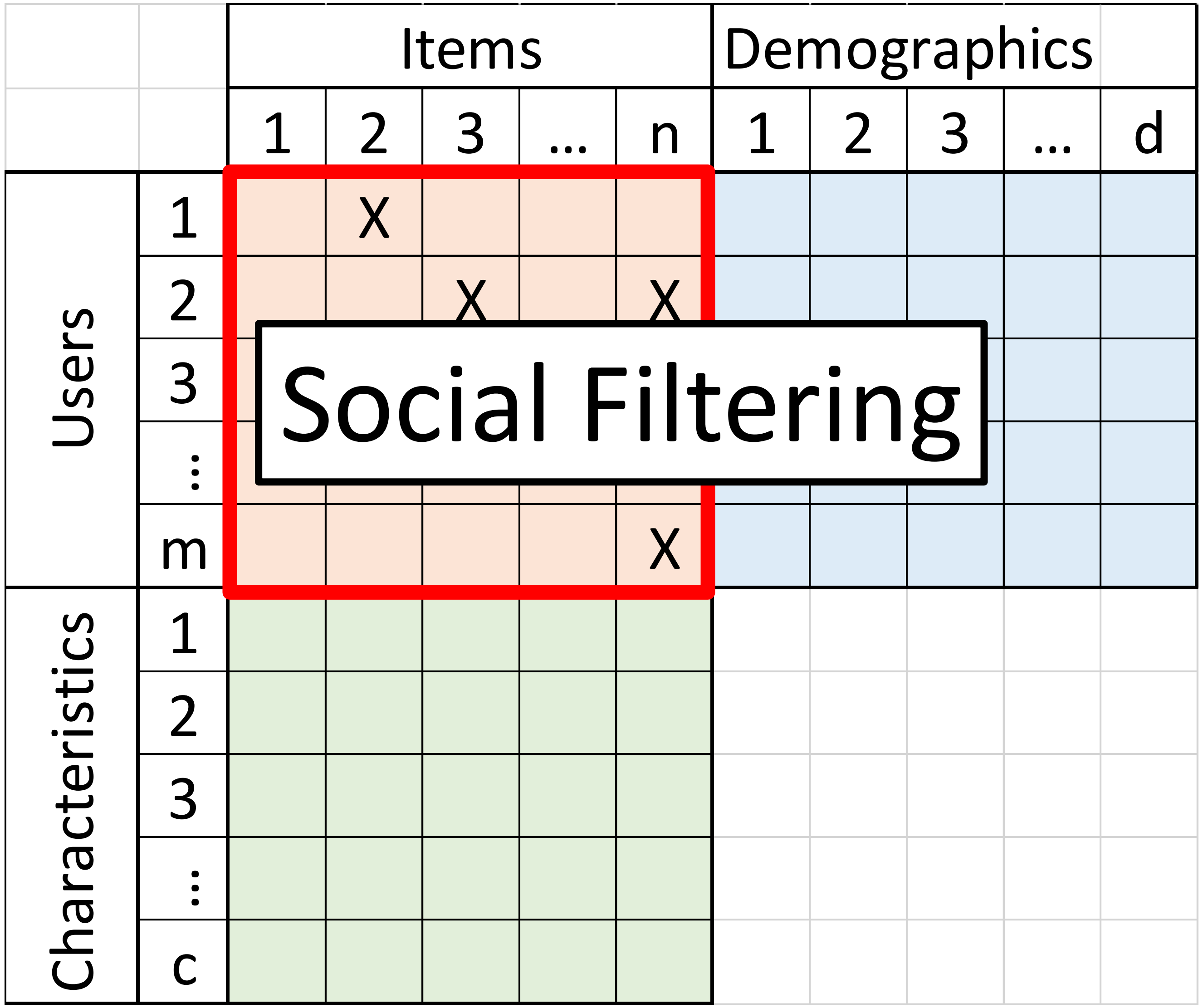
Information Filtering Framework



Information Filtering Framework



Information Filtering Framework



Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X		?						
	⋮										
	m					X					
Characteristics	1										
	2										
	3										
	⋮										
	c										

Collaborative/Social Filtering/Recommendation

- Intuition:
  If Al and Bob like a lot of the same things, and Al likes something Bob hasn’t seen, then Bob is more likely to like it too
- Approach:
  - Inspect the items you like and don’t like
    - Can explicitly ask for such information, or infer it by observation
  - Find other people with similar profiles of likes and dislikes
  - Recommend other items that those people like
- Example: Amazon recommendations

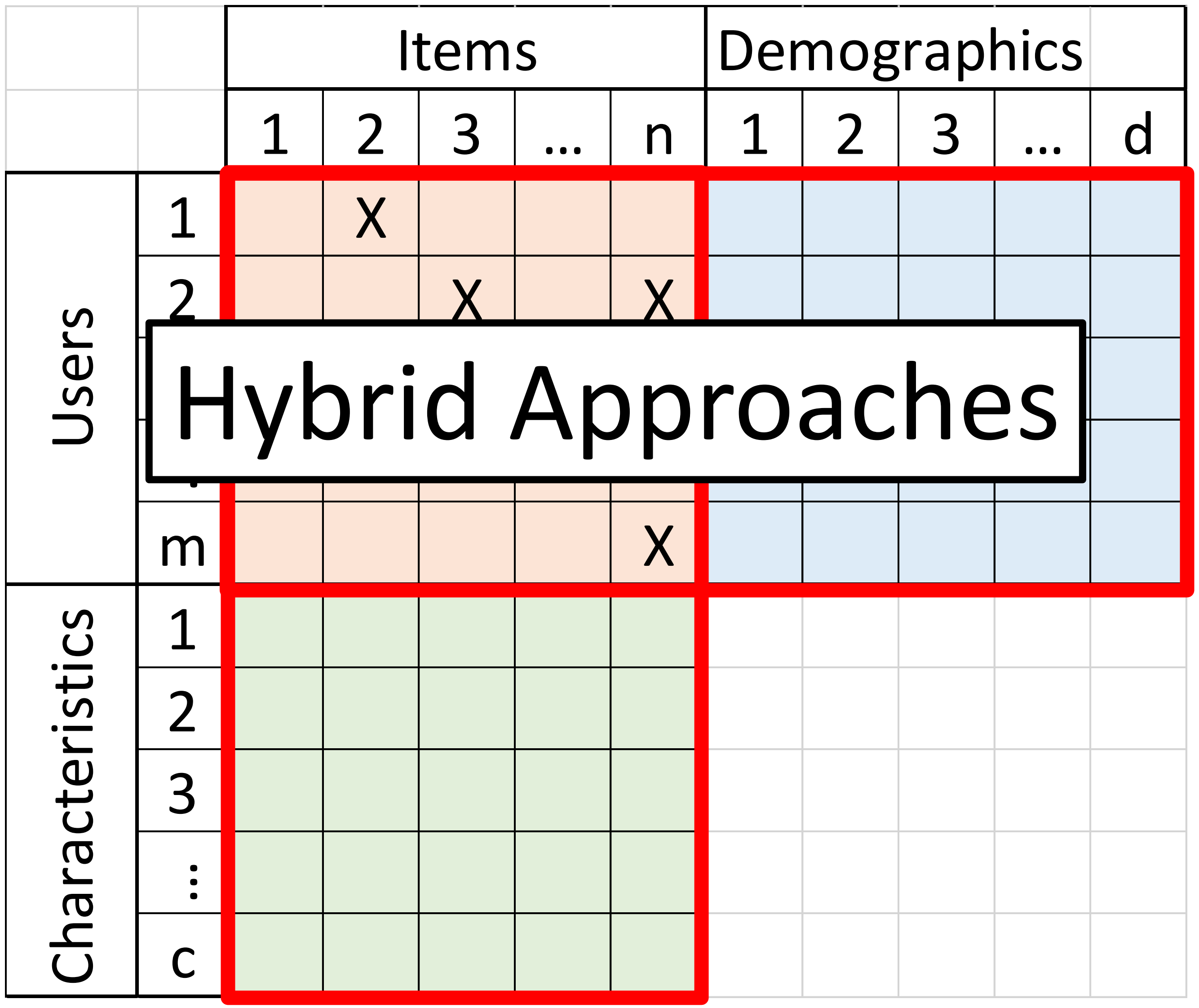
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X								
	⋮										
	m					X					
Characteristics	1										
	2										
	3										
	⋮										
	c										

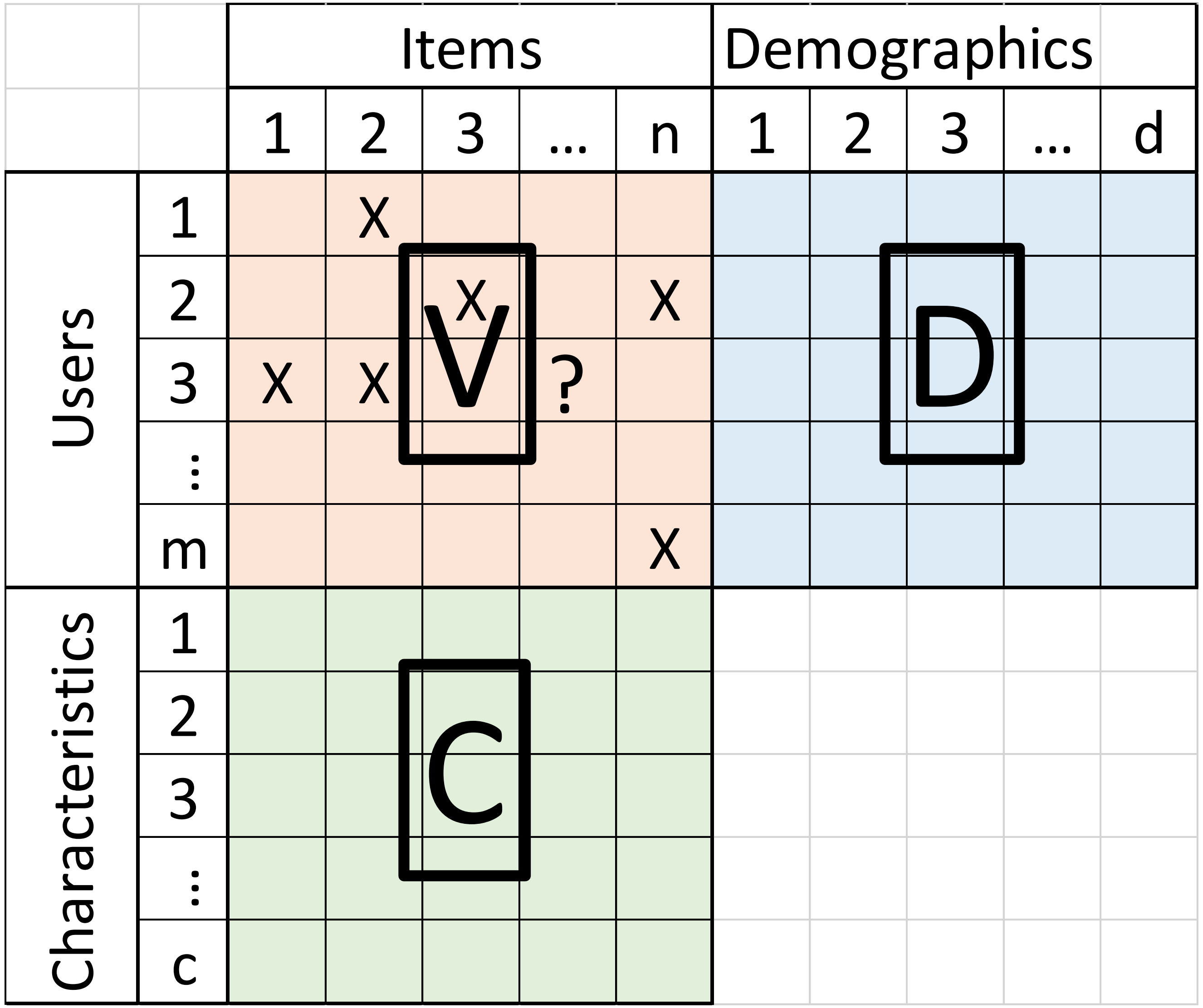
Information Filtering Framework

		Items					Demographics				
		1	2	3	…	n	1	2	3	…	d
Users	1		X								
	2			X		X					
	3	X	X		?						
	⋮										
	m					X					
Characteristics	1										
	2										
	3										
	⋮										
	c										

Information Filtering Framework



Information Filtering Framework



Information Filtering Framework

- Given:
  - Information about items C
  - Information about users D
  - User ratings of items V
- Predict:
  - $V_{i,j}$

Collaborative Filtering

- Widely used in e-commerce
- Often called “recommender systems”

- Let the “crowd” recommend things to you

“Memory-Based” Approach

- $v_{i,j}$= vote of user *i* on item *j*
- I_i = items for which user *i* has voted
- Mean vote for *i* is

$$\overline{v}_i = \frac{1}{|I_i|} \sum_{j \in I_i} v_{i,j}$$

- Predicted vote for “active user” *a* is weighted sum of *n* “nearest” users

$$p_{a,j} = \overline{v}_a + \kappa \sum_{i=1}^{n} \underbrace{w(a,i)}_{\text{weights of } n \text{ similar users}} (v_{i,j} - \overline{v}_i)$$

normalizer

"Memory-Based" Approach

- K-nearest neighbor

$$w(a,i) = \begin{cases} 1 & \text{if } i \in \text{neighbors}(a) \\ 0 & \text{else} \end{cases}$$

- Pearson correlation coefficient (Resnick '94, Grouplens):

$$w(a,i) = \frac{\sum_j (v_{a,j} - \overline{v}_a)(v_{i,j} - \overline{v}_i)}{\sqrt{\sum_j (v_{a,j} - \overline{v}_a)^2 \sum_j (v_{i,j} - \overline{v}_i)^2}}$$

- Cosine distance (from IR)

$$w(a,i) = \sum_j \frac{v_{a,j}}{\sqrt{\sum_{k \in I_a} v_{a,k}^2}} \frac{v_{i,j}}{\sqrt{\sum_{k \in I_i} v_{i,k}^2}}$$

“Memory-Based” Approach

- Cosine with “inverse user frequency” $f_i = log(n/n_j)$, where n is number of users, n_j is number of users voting for item j

$$w(a,i) = \frac{\sum_j f_j \sum_j f_j v_{a,j} v_{i,j} - (\sum_j f_j v_{a,j})(\sum_j f_j v_{i,j})}{\sqrt{UV}}$$

where

$$U = \sum_j f_j (\sum_j f_j v_{a,j}^2 - (\sum_j f_j v_{a,j})^2)$$

$$V = \sum_j f_j (\sum_j f_j v_{i,j}^2 - (\sum_j f_j v_{i,j})^2)$$

"Item-Based" Approach

- For each item find other items with similar profiles of ratings
- Recommend to me items with similar profiles to the ones I like

Collaborative Filtering Challenges

- Data sparsity:
  - Early stages of a system when there are few ratings
  - “Cold start” problem: New user with no ratings
  - New items with no ratings
- “Shilling” attacks:
  - Fake ratings that make an item look good
  - Related to Sybil attacks
- Recommending items in the “long tail”:
  - Can wind up only recommending popular items

Hybrid Approaches

View as machine learning

- Given:
  - Training data V(<C(i)>,<D(j)>)
- Predict:
  - New item V(<C(a)>,<D(b)>)