

COMP 345: Data Mining

More on Recommender Systems

Slides Adapted From: www.mmds.org (Mining Massive Datasets)

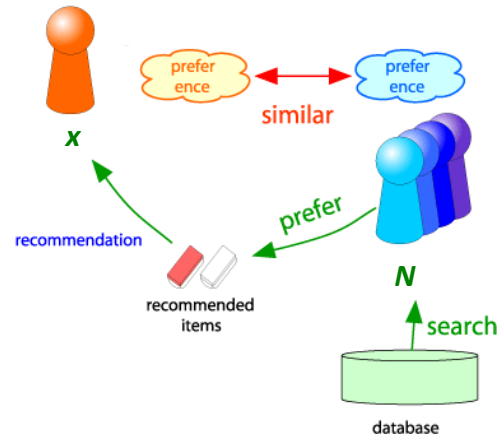


Collaborative Filtering

Harnessing quality judgments of other users

Collaborative Filtering

- Consider user x
- Find set N of other users whose ratings are “similar” to x ’s ratings
- Estimate x ’s ratings based on ratings of users in N



3

Similarity Metric

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- Consider users x and y with rating vectors r_x and r_y
- We need a similarity metric $\text{sim}(x, y)$
- Capture intuition that $\text{sim}(A, B) > \text{sim}(A, C)$

4

Option 1: Jaccard Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A, B) = |r_A \cap r_B| / |r_A \cup r_B|$
- $\text{sim}(A, B) = 1/5$; $\text{sim}(A, C) = 2/4$
– $\text{sim}(A, B) < \text{sim}(A, C)$
- Problem: Ignores ratings values!

5

Option 2: Cosine Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A, B) = \cos(r_A, r_B)$

6

Example: Cosine Similarity

- $\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| ||d_2||$,
where \bullet indicates vector dot product, $||d||$: the length of vector d
- Ex: Find the **similarity** between documents 1 and 2.

$$d_1 = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)$$

$$d_2 = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)$$

$$d_1 \bullet d_2 = 5*3 + 0*0 + 3*2 + 0*0 + 2*1 + 0*1 + 0*1 + 2*1 + 0*0 + 0*1 = 25$$

$$||d_1|| = (5*5 + 0*0 + 3*3 + 0*0 + 2*2 + 0*0 + 0*0 + 2*2 + 0*0 + 0*0)^{0.5} = (42)^{0.5} = 6.481$$

$$||d_2|| = (3*3 + 0*0 + 2*2 + 0*0 + 1*1 + 1*1 + 0*0 + 1*1 + 0*0 + 1*1)^{0.5} = (17)^{0.5} = 4.12$$

$$\cos(d_1, d_2) = 0.94$$

7

Option 2: Cosine Similarity

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

- $\text{sim}(A, B) = \cos(r_A, r_B)$
- $\text{sim}(A, B) = 0.38$; $\text{sim}(A, C) = 0.32$
– $\text{sim}(A, B) > \text{sim}(A, C)$, but not by much
- Problem: treats missing ratings as negative

8

Option 3: Centered Cosine

Normalize ratings by subtracting row mean

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	2/3			5/3	-7/3		
B	1/3	1/3	-2/3				
C				-5/3	1/3	4/3	
D		0					0

Also known as
the Pearson
Correlation
Coefficient

$\text{sim}(A, B) = \cos(rA, rB) = 0.09$; $\text{sim}(A, C) = -0.44$

- $\text{sim}(A, B) > \text{sim}(A, C)$
- Captures intuition better
 - Missing ratings treated as “average”
 - Handles “tough raters” and “easy raters”

Rating Predictions

From similarity metric to recommendations:

- Let r_x be the vector of user x 's ratings
- Let N be the set of k users most similar to x who have rated item i
- **Prediction for item s of user x :**
 - $r_{xi} = \frac{1}{k} \sum_{y \in N} r_{yi}$
 - $r_{xi} = \frac{\sum_{y \in N} s_{xy} \cdot r_{yi}}{\sum_{y \in N} s_{xy}}$
 - Other options?
- **Many other tricks possible...**

Shorthand:
 $s_{xy} = \text{sim}(x, y)$

Item-Item Collaborative Filtering

- So far: **User-user collaborative filtering**
- **Another view: Item-item**
 - For item i , find other similar items
 - Estimate rating for item i based on ratings for similar items
 - Can use same similarity metrics and prediction functions as in user-user model

$$r_{xi} = \frac{\sum_{j \in N(i;x)} s_{ij} \cdot r_{xj}}{\sum_{j \in N(i;x)} s_{ij}}$$

s_{ij} ... similarity of items i and j
 r_{xj} ... rating of user u on item j
 $N(i;x)$... set items rated by x similar to i

11

Item-Item CF ($|N|=2$)

		users											
		1	2	3	4	5	6	7	8	9	10	11	12
movies	1	1		3			5			5		4	
	2			5	4			4			2	1	3
	3	2	4		1	2		3		4	3	5	
	4		2	4		5			4			2	
	5			4	3	4	2					2	5
	6	1		3		3			2			4	



- unknown rating



- rating between 1 to 5

12

Item-Item CF ($|N|=2$)

		users											
		1	2	3	4	5	6	7	8	9	10	11	12
movies	1	1		3		?	5			5		4	
	2			5	4			4			2	1	3
	3	2	4		1	2		3		4	3	5	
	4		2	4		5			4			2	
	5			4	3	4	2					2	5
	6	1		3		3			2			4	



- estimate rating of movie 1 by user 5

13

Item-Item CF ($|N|=2$)

		users												
		1	2	3	4	5	6	7	8	9	10	11	12	sim(1,m)
movies	1	1		3		?	5			5		4		1.00
	2			5	4			4			2	1	3	-0.18
	<u>3</u>	2	4		1	2		3		4	3	5		<u>0.41</u>
	4		2	4		5			4			2		-0.10
	5			4	3	4	2					2	5	-0.31
	<u>6</u>	1		3		3			2			4		<u>0.59</u>

Neighbor selection:
Identify movies similar to
movie 1, rated by user 5

Here we use Pearson correlation as similarity:

1) Subtract mean rating m_i from each movie i

$$m_i = (1+3+5+5+4)/5 = 3.6$$

row 1: [-2.6, 0, -0.6, 0, 0, 1.4, 0, 0, 1.4, 0, 0, 0.4, 0]

2) Compute cosine similarities between rows

Item-Item CF ($|N|=2$)

		users												
		1	2	3	4	5	6	7	8	9	10	11	12	
movies	1	1		3		?	5			5		4		$\text{sim}(1,m)$ 1.00
	2			5	4			4			2	1	3	-0.18
	<u>3</u>	2	4		1	2		3		4	3	5		<u>0.41</u>
	4		2	4		5			4			2		-0.10
	5			4	3	4	2					2	5	-0.31
	<u>6</u>	1		3		3			2			4		<u>0.59</u>

Compute similarity weights:

$$s_{1,3}=0.41, s_{1,6}=0.59$$

15

Item-Item CF ($|N|=2$)

		users												
		1	2	3	4	5	6	7	8	9	10	11	12	
movies	1	1		3		2.6	5			5		4		
	2			5	4			4			2	1	3	
	<u>3</u>	2	4		1	2		3		4	3	5		
	4		2	4		5			4			2		
	5			4	3	4	2					2	5	
	<u>6</u>	1		3		3			2			4		

Predict by taking weighted average:

$$r_{1,5} = (0.41 \cdot 2 + 0.59 \cdot 3) / (0.41 + 0.59) = 2.6$$

$$r_{ix} = \frac{\sum_{j \in N(i,x)} s_{ij} \cdot r_{jx}}{\sum s_{ij}}$$

16

CF: Common Practice

Before:

$$r_{xi} = \frac{\sum_{j \in N(i;x)} s_{ij} r_{xj}}{\sum_{j \in N(i;x)} s_{ij}}$$

- Define **similarity** s_{ij} of items i and j
- Select k nearest neighbors $N(i, x)$
 - Items most similar to i , that were rated by x
- Estimate rating r_{xi} as the weighted average:

$$r_{xi} = b_{xi} + \frac{\sum_{j \in N(i;x)} s_{ij} \cdot (r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s_{ij}}$$

baseline estimate for r_{xi}

$$b_{xi} = \mu + b_x + b_i$$

- μ = overall mean movie rating
- b_x = rating deviation of user x
= (avg. rating of user x) - μ
- b_i = rating deviation of movie i

17

Item-Item vs. User-User

- In theory, user-user and item-item are dual approaches.
- In practice, item-item outperforms user-user in many use cases.
- Items are “simpler” than users
 - Items belong to a small set of “genres”, users have varied tastes.
 - Item Similarity is more meaningful than User Similarity

18

Pros/Cons of Collaborative Filtering

- **+ Works for any kind of item**
 - No feature selection needed
- **- Cold Start:**
 - Need enough users in the system to find a match
- **- Sparsity:**
 - The user/ratings matrix is sparse
 - Hard to find users that have rated the same items
- **- First rater:**
 - Cannot recommend an item that has not been previously rated
 - New items, Esoteric items
- **- Popularity bias:**
 - Cannot recommend items to someone with unique taste
 - Tends to recommend popular items

19

Hybrid Methods

- **Implement two or more different recommenders and combine predictions**
 - Perhaps using a linear model
- **Add content-based methods to collaborative filtering**
 - Item profiles for new item problem
 - Demographics to deal with new user problem

20

Evaluation

movies

users

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

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmids.org>

21

Evaluation

movies

users

1	3	4			
	3	5			5
		4	5		5
		3			
		3			
2			?		?
				?	
	2	1			?
	3			?	
1					

Test Data Set

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmids.org>

22

Evaluating Predictions

- Compare predictions with known ratings (test set T)
 - **Root-mean-square error (RMSE)**

$$\sqrt{\frac{\sum_{(x,i) \in T} (r_{xi} - r_{xi}^*)^2}{N}}$$

- where $N = |T|$
- r_{xi} is predicted rating
- r_{xi}^* is the actual rating of x on i

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmids.org>

23

Problems with Error Measures

- **Narrow focus on accuracy sometimes misses the point**
 - Prediction Diversity
 - Prediction Context
 - Order of predictions
- **In practice, we care only to predict high ratings:**
 - RMSE might penalize a method that does well for high ratings and badly for others
 - Alternative: precision at top k
 - Percentage of predictions in the user's top k withheld ratings

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmids.org>

24