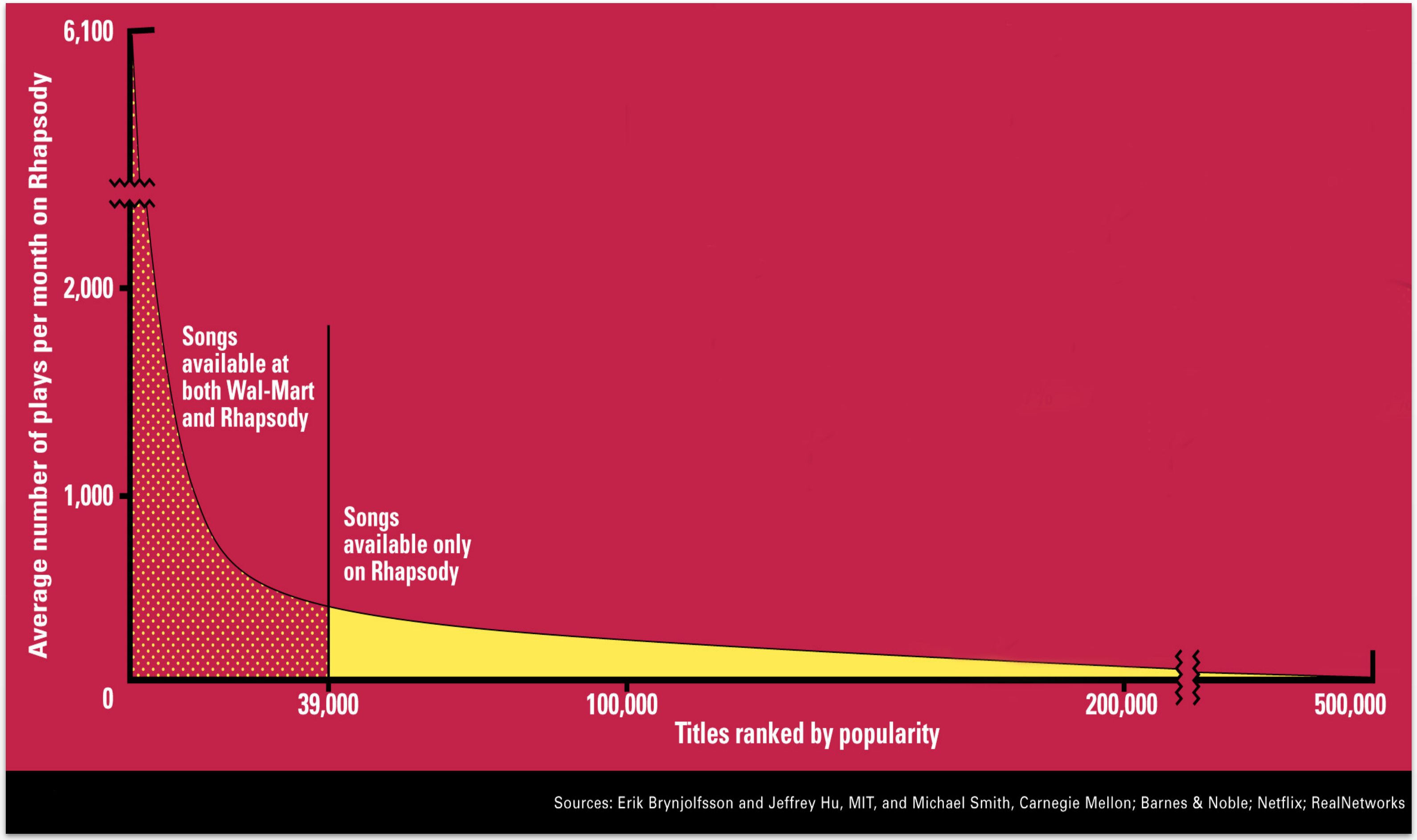
Recommender Systems

Shantanu Jain

Recommender Systems

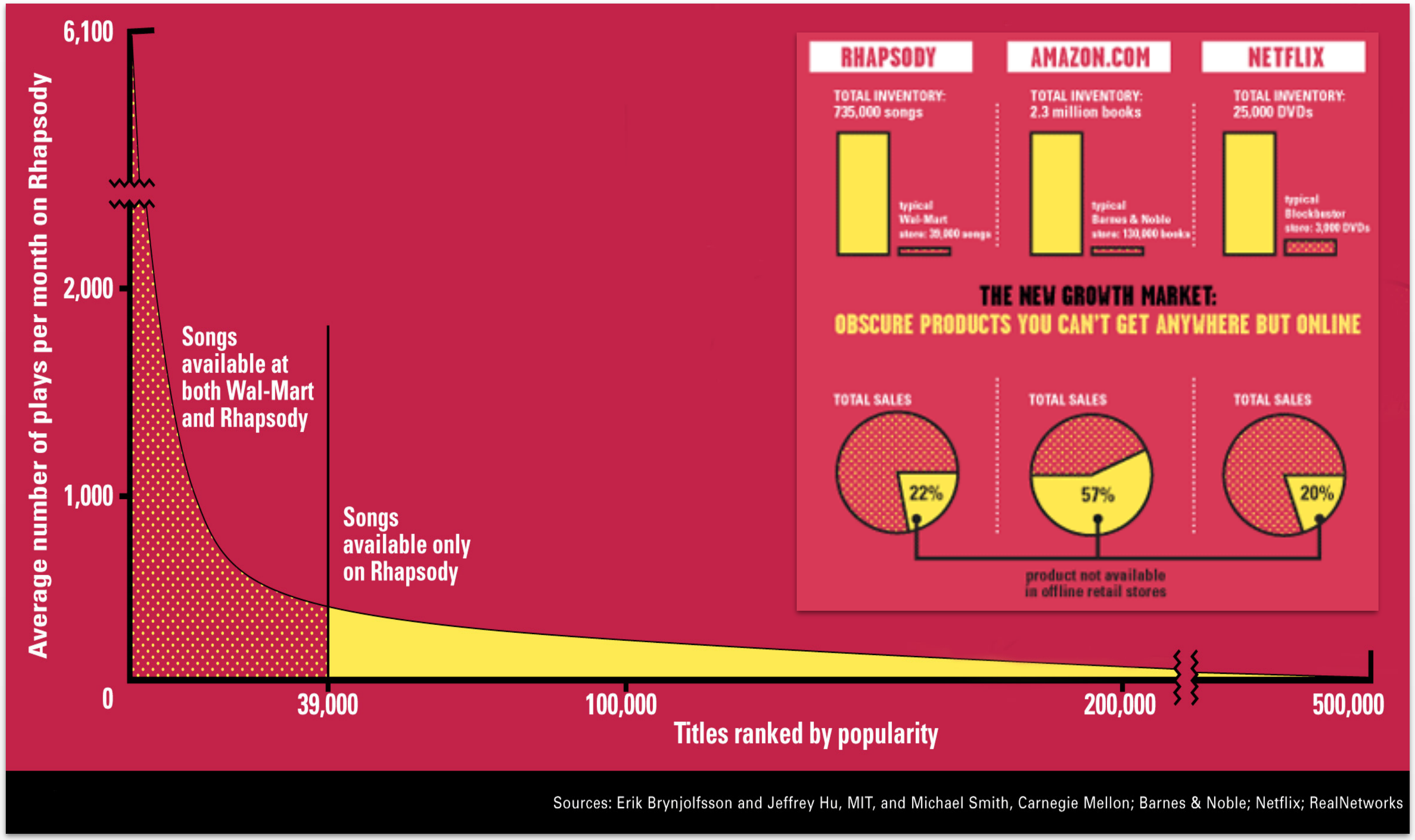
Reasoning about the Long Tail

The Long Tail



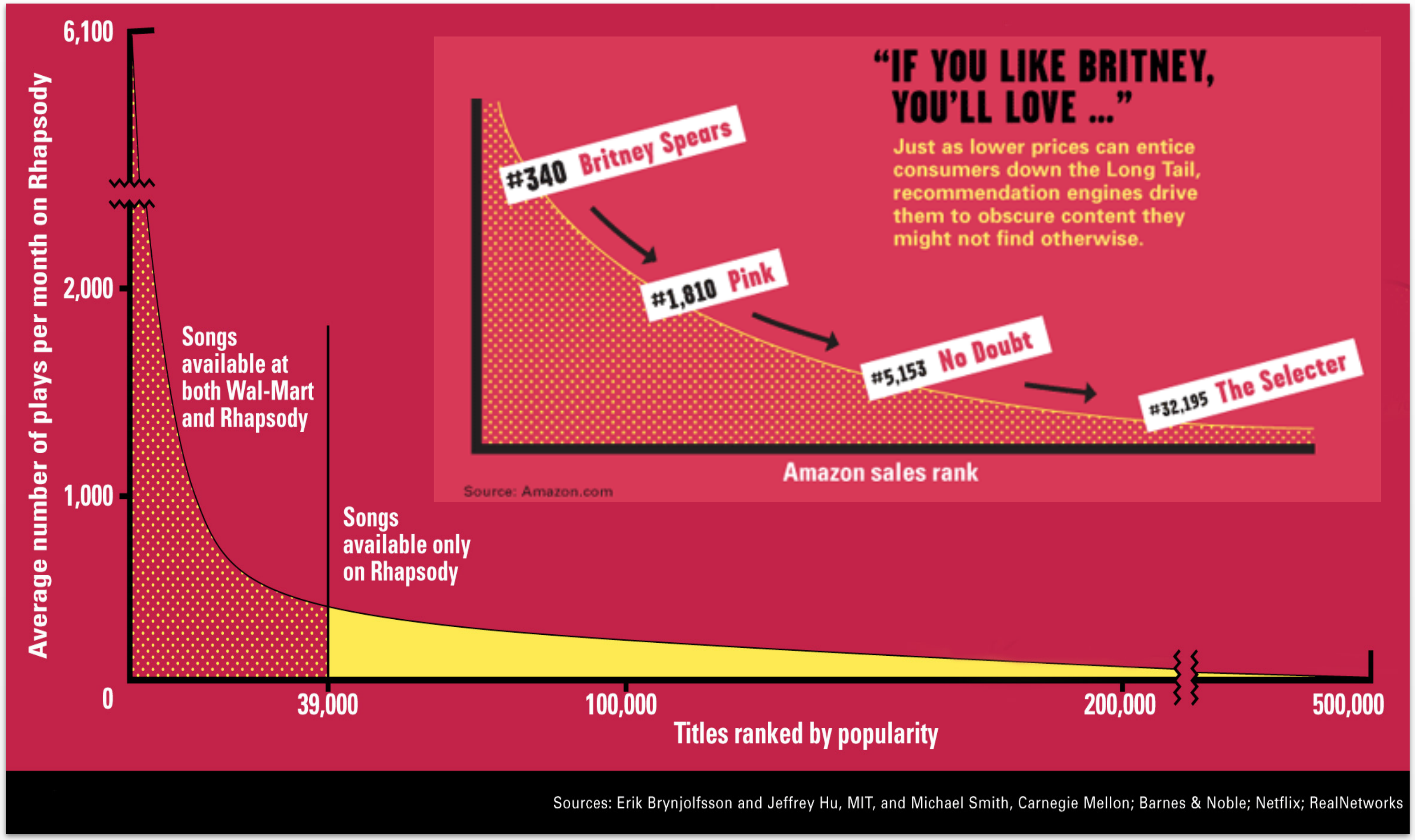
(from: https://www.wired.com/2004/10/tail/)

The Long Tail



(from: https://www.wired.com/2004/10/tail/)

The Long Tail



(from: https://www.wired.com/2004/10/tail/)

Applications of Recommender Systems

- Movie recommendation (Netflix)
- Related product recommendation (Amazon)
- Web page ranking (Google)
- Social recommendation (Facebook)
- Priority inbox & spam filtering (Google)
- Online dating (OK Cupid)
- Computational Advertising (Everyone)

Problem Setting

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

Problem Setting

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

Problem Setting

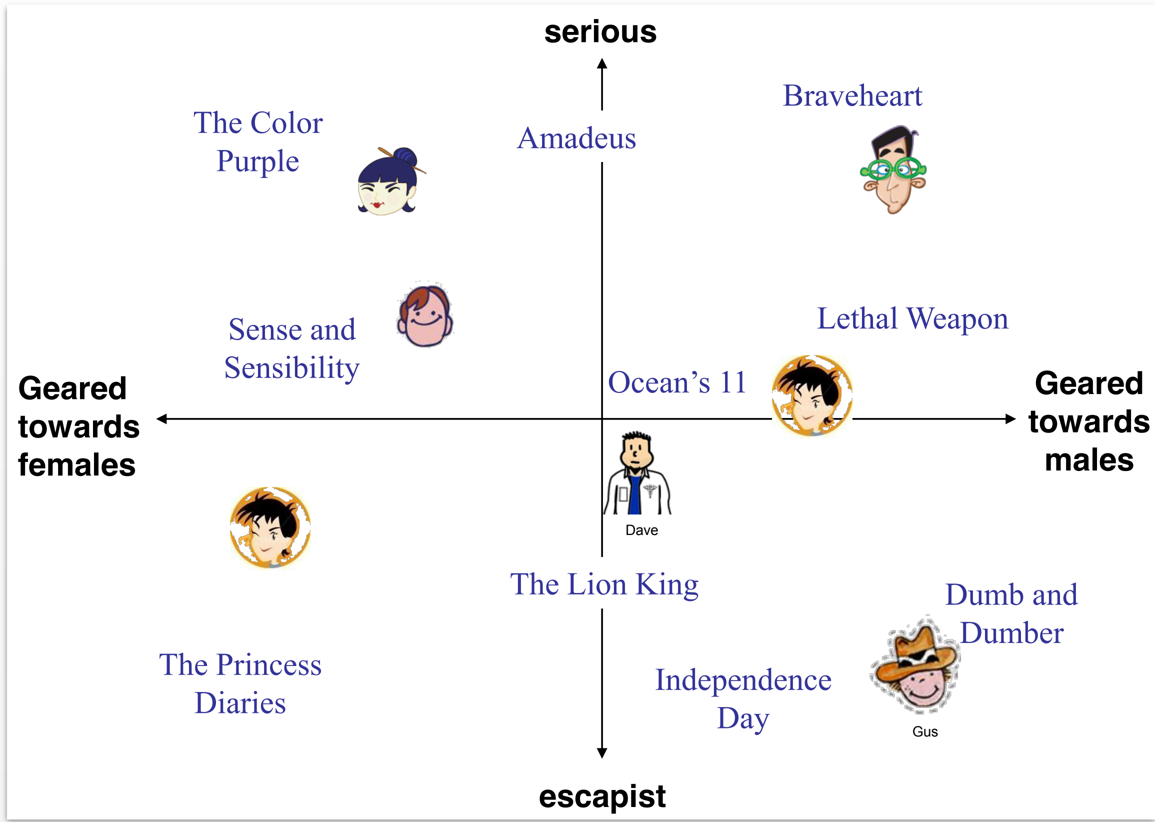
Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

Problem Setting

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

- *Task*: Predict user preferences for unseen items

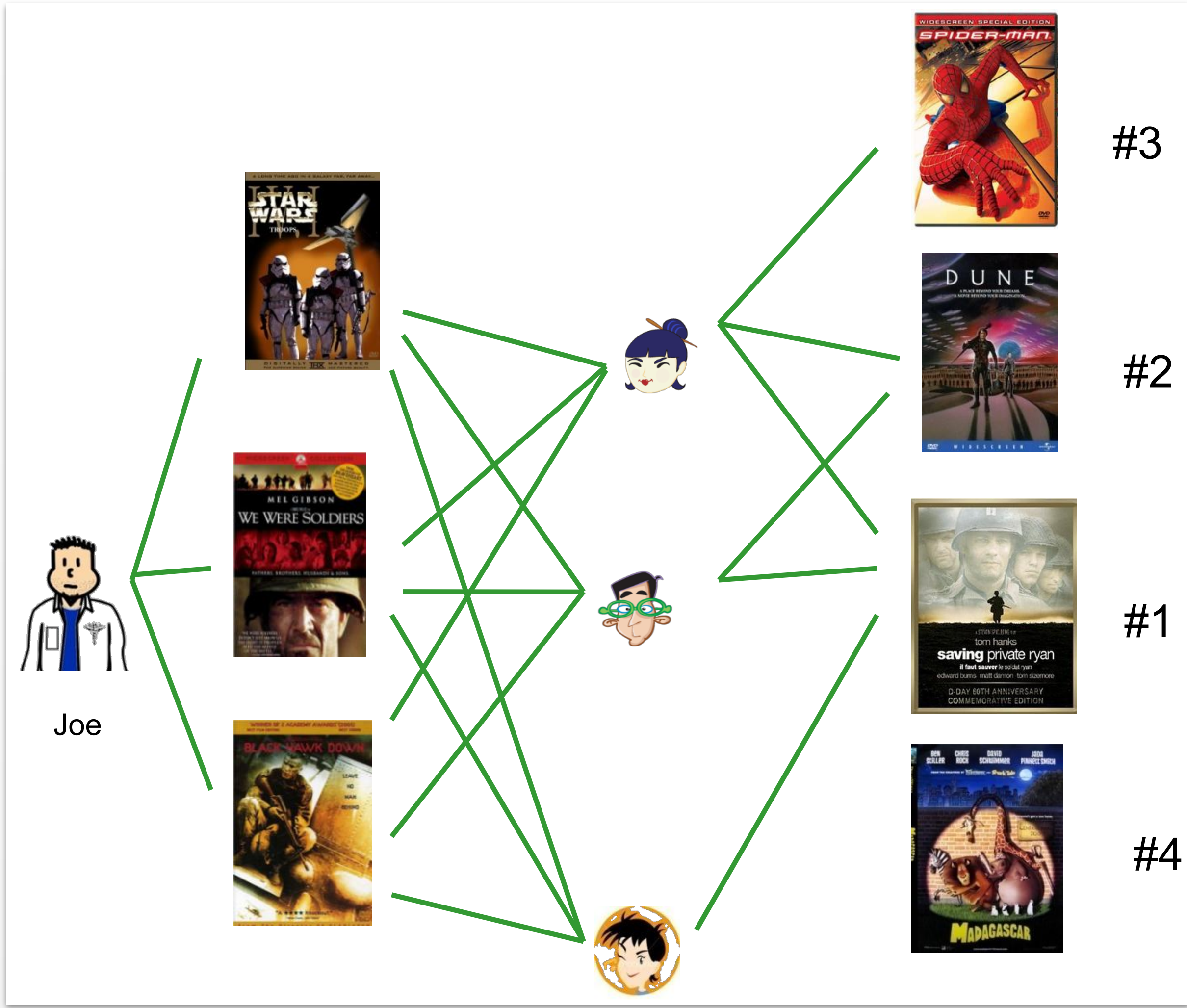
Content-based Filtering



Two Approaches:

1. Predict rating using ***item*** features on a ***per-user*** basis
2. Predict rating using ***user*** features on a ***per-item*** basis

Collaborative Filtering



Idea: Predict rating based on similarity to other users

Problem Setting

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

- *Task*: Predict user preferences for unseen items
- *Content-based filtering*: Model user/item features
- *Collaborative filtering*: Implicit similarity of users or items

Running Yardstick: RMSE

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

$$\mathrm{rmse}(S) = \sqrt{|S|^{-1} \sum_{(i,u) \in S} (\hat{r}_{ui} - r_{ui})^2}$$

S contains user-item pairs for which ratings are observed

Recommender Systems

Shantanu Jain

Content-based Filtering

Feature-based recommendation

Item-based Features

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?

Item-based Features

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9

Item-based Features

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9

Per-user Regression

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9

Learn a set of regression coefficients for each user

$$\boldsymbol{w}_u = \underset{\boldsymbol{w}}{\operatorname{argmin}} |\boldsymbol{r}_u - \boldsymbol{X}\boldsymbol{w}|^2 \qquad w_u = (X^T X)^{-1} X^T r_u$$

Each row of X contains an encoding for an item.

Bias

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9

Bias

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9
Moonrise Kingdom	4	5	4	4	0.3	0.2

Bias

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	5	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	4	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9
Moonrise Kingdom	4	5	4	4	0.3	0.2

Problem: Some movies are universally loved / hated

Bias

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	3	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	3	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9
Moonrise Kingdom	4	3	4	4	0.3	0.2

Problem: Some movies are universally loved / hated
some users are more picky than others

Bias

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)	x_1 (romance)	x_2 (action)
Love at last	5	3	0	0	0.9	0
Romance forever	5	?	?	0	1.0	0.01
Cute puppies of love	?	3	0	?	0.99	0
Nonstop car chases	0	0	5	4	0.1	1.0
Swords vs. karate	0	0	5	?	0	0.9
Moonrise Kingdom	4	3	4	4	0.3	0.2

Problem: Some movies are universally loved / hated
some users are more picky than others

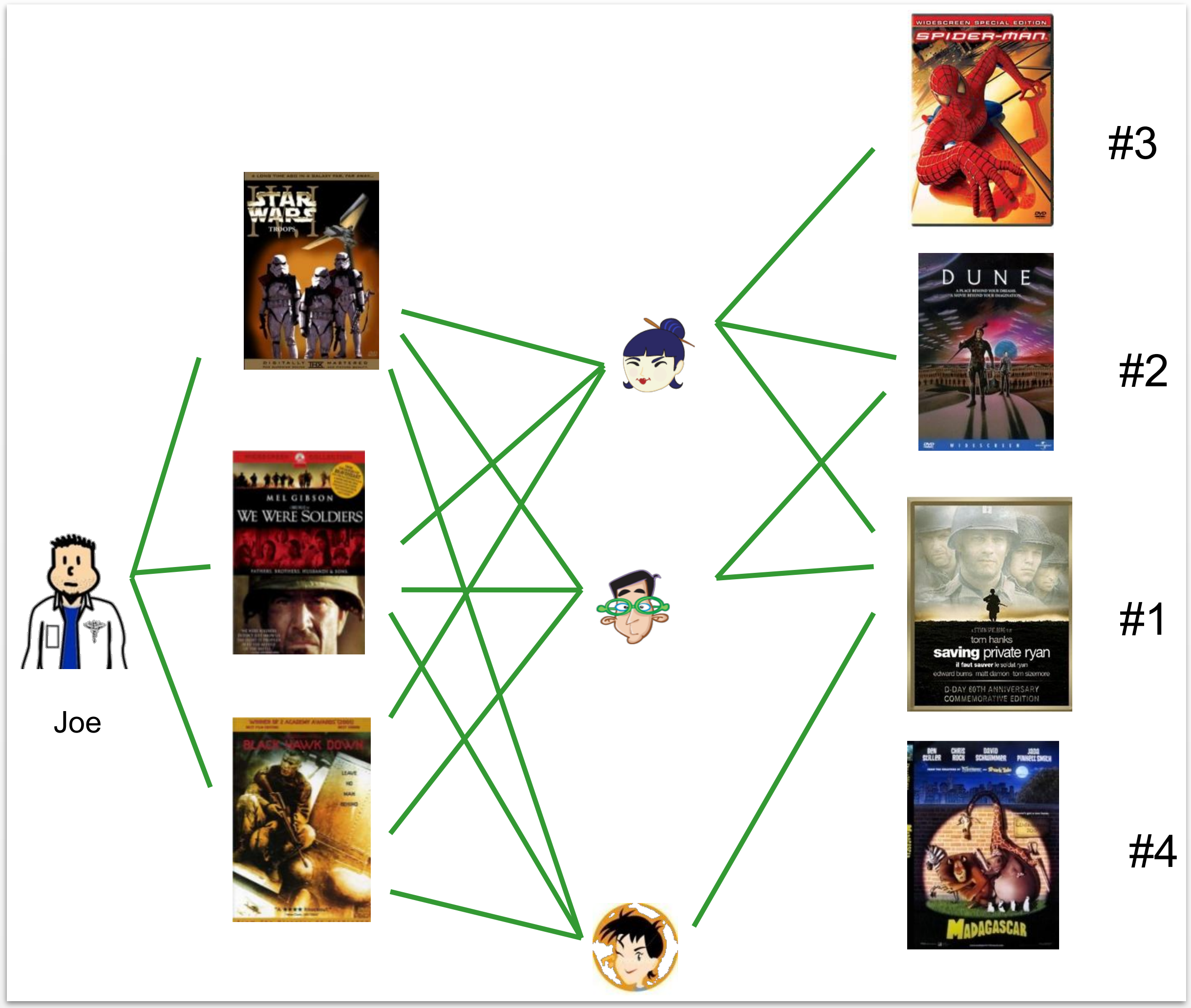
Solution: Introduce a per-movie and per-user bias

$$\hat{r}_{ui} = \mu + b_u + b_i + \boldsymbol{x}_i^\top \boldsymbol{w}_u$$

Collaborative Filtering

Connectivity-based recommendation

Neighborhood Based Methods



Users and items form a bipartite graph (edges are ratings)

Neighborhood Based Methods

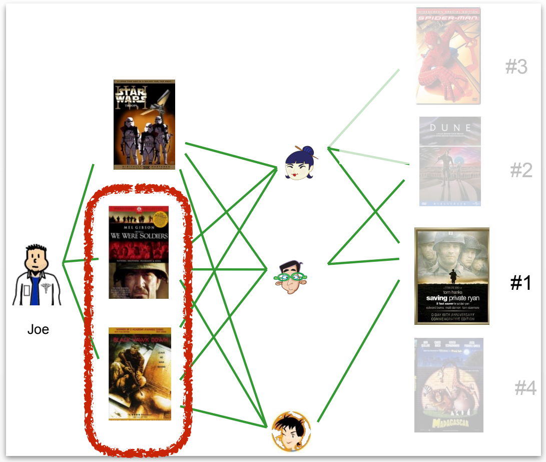
(user, user) similarity

- predict rating based on average from k-nearest users
- good if item base changes rapidly

(item,item) similarity

- predict rating based on average from k-nearest items
- good if user base changes rapidly

Parzen-Window Style CF



$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in \epsilon_k(i,u)} s_{ij}(r_{uj} - b_{uj})}{\sum_{j \in \epsilon_k(i,u)} |s_{ij}|}$$

$$b_{ui} = \mu + b_u + b_i$$

- Define a similarity s_{ij} between items
- Find set $\varepsilon_k(i,u)$ of k-nearest neighbors to *i* that were rated by user *u*
- Predict rating using weighted average over set
- How should we define s_{ij}?

Pearson Correlation Coefficient

User ratings for item i:

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

User ratings for item j:

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

$$s_{ij} = \frac{\mathrm{cov}(r_{\cdot i}, r_{\cdot j})}{\mathrm{std}(r_{\cdot i}) \times \mathrm{std}(r_{\cdot j})}$$

(item,item) similarity

Empirical estimate of Pearson correlation coefficient

$$\hat{\rho}_{ij} = \frac{\sum_{u \in U(i,j)} (r_{ui} - b_{ui})(r_{uj} - b_{uj})}{\sqrt{\sum_{u \in U(i,j)} (r_{ui} - b_{ui})^2 \sum_{u \in U(i,j)} (r_{uj} - b_{uj})^2}}$$

U(i, j): set of users who have rated both *i* and *j*

Regularize towards 0 for small support

$$s_{ij} = \frac{|U(i,j)| - 1}{|U(i,j)| - 1 + \lambda} \hat{\rho}_{ij}$$

Regularize towards baseline for small neighborhood

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in \epsilon_k(i,u)} s_{ij}(r_{uj} - b_{uj})}{\lambda + \sum_{j \in \epsilon_k(i,u)} |s_{ij}|}$$

Similarity for binary labels

Pearson correlation not meaningful for binary labels
(e.g. Views, Purchases, Clicks)

Jaccard similarity

$$s_{ij} = \frac{m_{ij}}{\alpha + m_i + m_j - m_{ij}}$$

Observed / Expected ratio

$$s_{ij} = \frac{\text{observed}}{\text{expected}} \approx \frac{m_{ij}}{\alpha + m_i m_j / m}$$

m_i users acting on i

m_{ij} users acting on both i and j

m total number of users

Recommender Systems

Shantanu Jain

Matrix Factorization Methods

Learning user and item features

Matrix Factorization

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?
Moonrise Kingdom	4	5	4	4

$$\hat{r}_{ui} = x_i^T w_u$$

Idea: pose as matrix factorization problem

$$\hat{R} = XW^T$$

Matrix Factorization

users

items

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

~

~ items

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

●

users

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

A rank-3 SVD approximation

Prediction

users

items

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

~

~

items

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

●

users

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

A rank-3 SVD approximation

Prediction

users

items												
	1		3			5			5		4	
			5		2.4		4			2	1	3
	2	4		1	2		3		4	3	5	
		2	4		5			4			2	
			4	3	4	2					2	5
	1		3		3			2			4	

~

~ items

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

●

users

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

A rank-3 SVD approximation

SVD with missing values

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

~

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

Pose as regression problem

$$\underset{X,W}{\operatorname{argmin}} \sum_{(u,i)\in S} (r_{ui} - \boldsymbol{w}_u^\top \boldsymbol{x}_i)^2 + \lambda \left(||\boldsymbol{X}||_{\mathrm{F}}^2 + ||\boldsymbol{W}||_{\mathrm{F}}^2 \right)$$

Regularize using Frobenius norm

$$||\boldsymbol{A}||_{\mathrm{F}}^2 = \sum_{ij} |A_{ij}|^2$$

Alternating Least Squares

$\boldsymbol{R}$

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

~

$\boldsymbol{X}$

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

$\boldsymbol{W}^\top$

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

$$\boldsymbol{w}_u \leftarrow \left[\lambda \boldsymbol{I} + \sum_{i:(u,i)\in S} \boldsymbol{x}_i \boldsymbol{x}_i^\top\right]^{-1} \sum_{i:(u,i)\in S} \boldsymbol{x}_i r_{ui}$$

(regress w_u given X)

Alternating Least Squares

$\boldsymbol{R}$

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

~

$\boldsymbol{X}$

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

$\boldsymbol{W}^\top$

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

$$\boldsymbol{w}_u \leftarrow \left[\lambda \boldsymbol{I} + \sum_{i:(u,i)\in S} \boldsymbol{x}_i \boldsymbol{x}_i^\top\right]^{-1} \sum_{i:(u,i)\in S} \boldsymbol{x}_i r_{ui}$$

(regress w_u given X)

*L*2: closed form solution

$$\mathbf{w} = (\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda \mathbf{I})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$

(known as Ridge Regression)

Alternating Least Squares

$\boldsymbol{R}$

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

~

$\boldsymbol{X}$

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

$\boldsymbol{W}^\top$

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

$$\boldsymbol{w}_u \leftarrow \left[\lambda \boldsymbol{I} + \sum_{i:(u,i)\in S} \boldsymbol{x}_i \boldsymbol{x}_i^\top\right]^{-1} \sum_{i:(u,i)\in S} \boldsymbol{x}_i r_{ui}$$

(regress w_u given X)

$$\boldsymbol{x}_i \leftarrow \left[\lambda \boldsymbol{I} + \sum_{u:(u,i)\in S} \boldsymbol{w}_u \boldsymbol{w}_u^\top\right]^{-1} \sum_{u:(u,i)\in S} \boldsymbol{w}_u r_{ui}$$

(regress x_i given W)

Matrix Factorization

Movie	Alice (1)	Bob (2)	Carol (3)	Dave (4)
Love at last	5	5	0	0
Romance forever	5	?	?	0
Cute puppies of love	?	4	0	?
Nonstop car chases	0	0	5	4
Swords vs. karate	0	0	5	?
Moonrise Kingdom	4	5	4	4

$$\hat{r}_{ui} = \mu + b_u + b_i + \boldsymbol{x}_i^\top \boldsymbol{w}_u$$

Idea: matrix factorization problem with bias terms

$$\hat{\boldsymbol{R}} = \boldsymbol{B} + \boldsymbol{X}\boldsymbol{W}^\top$$