CMSC5733 Social Computing

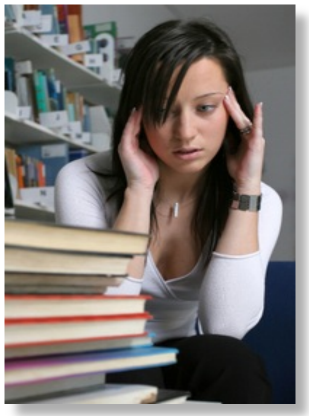
Irwin King

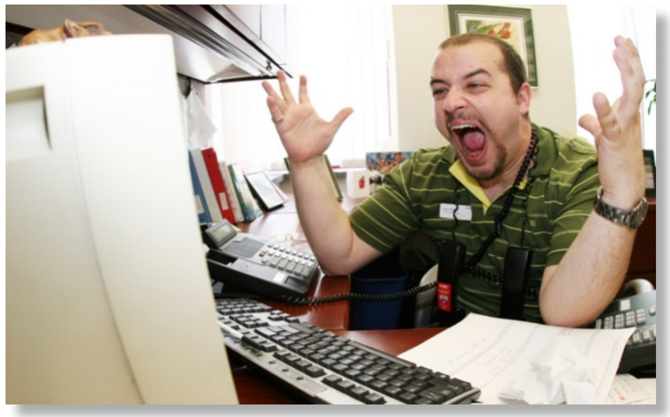
The Chinese University of Hong Kong

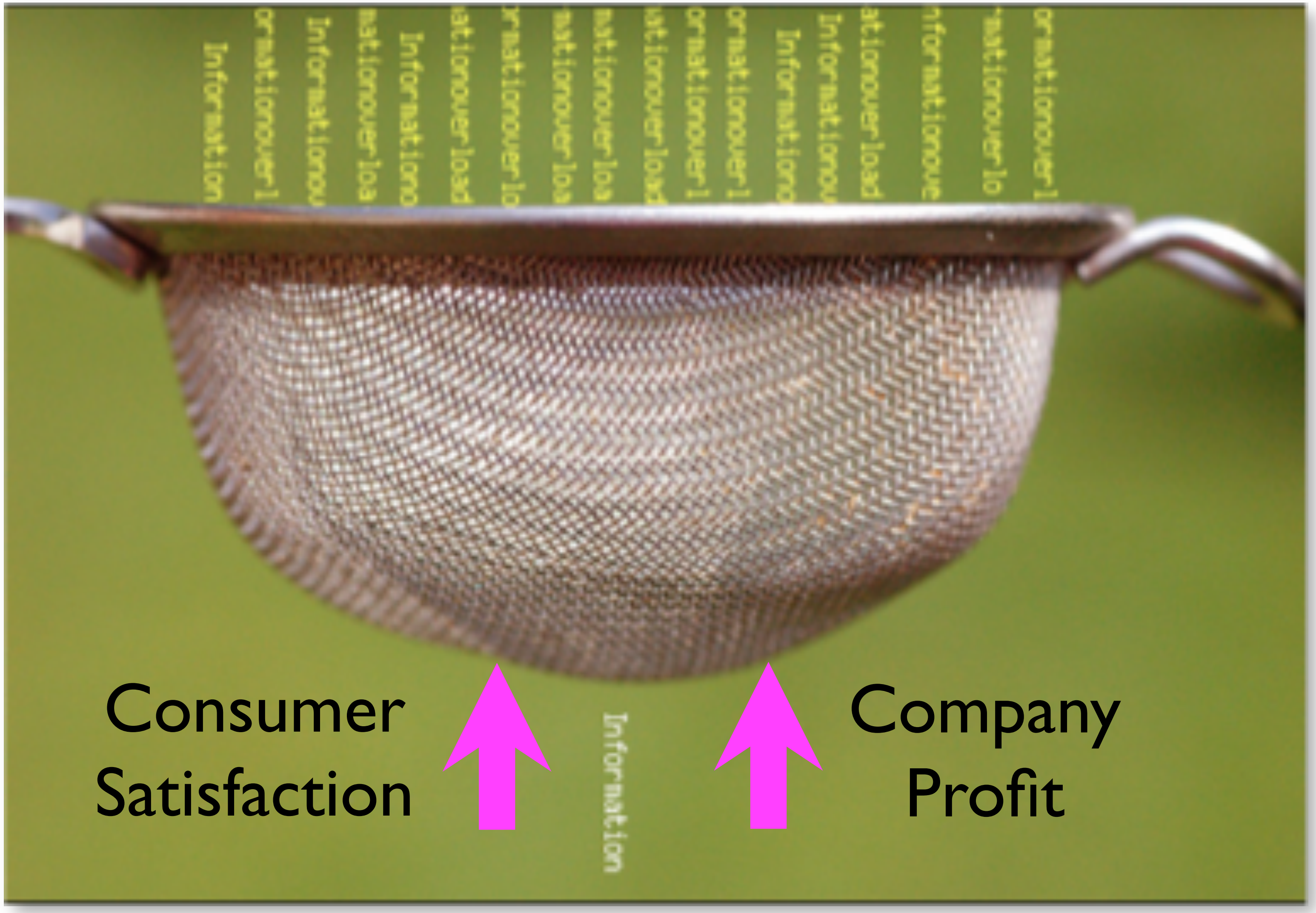
king@cse.cuhk.edu.hk

Information and more Information!

Information Overload

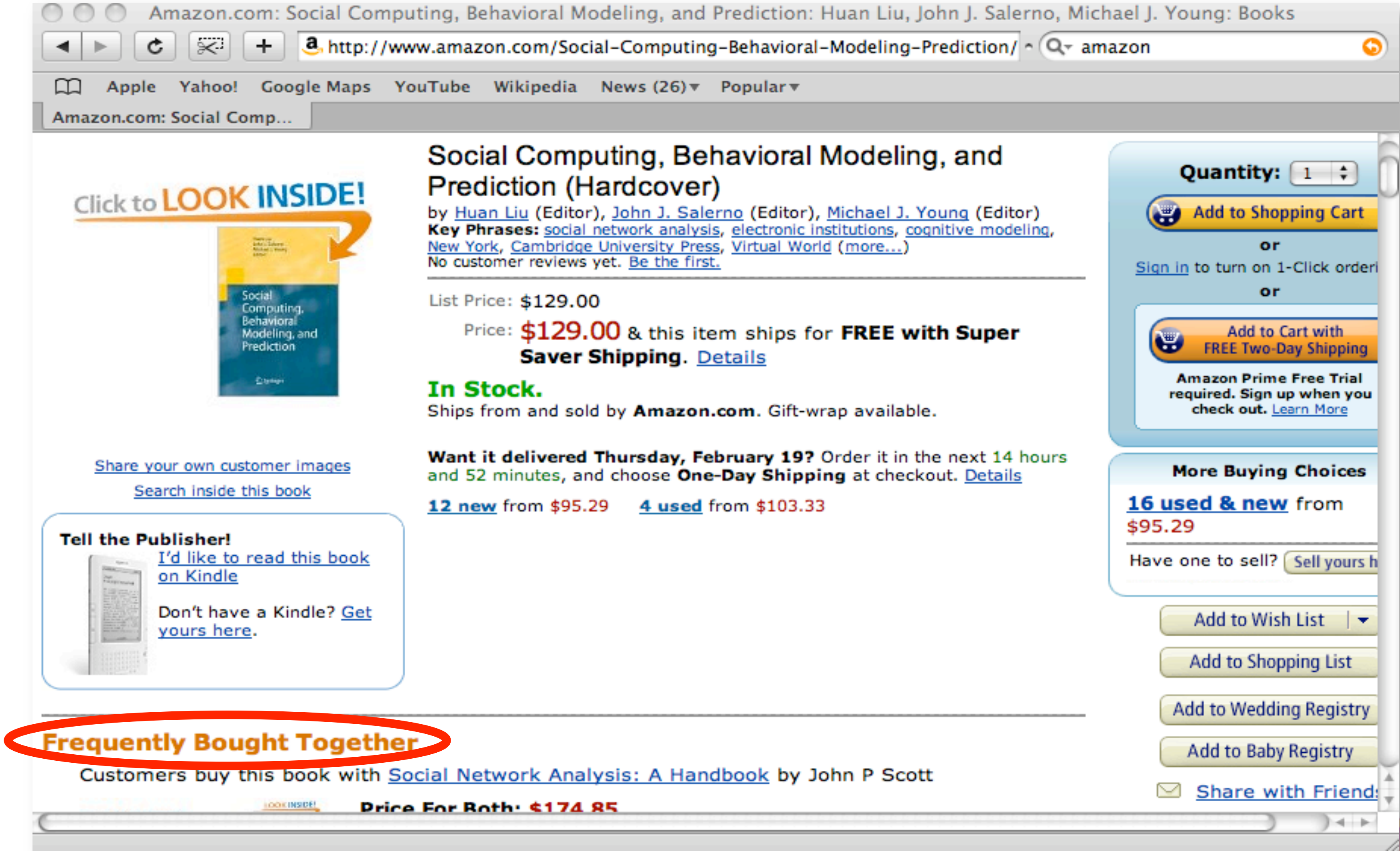




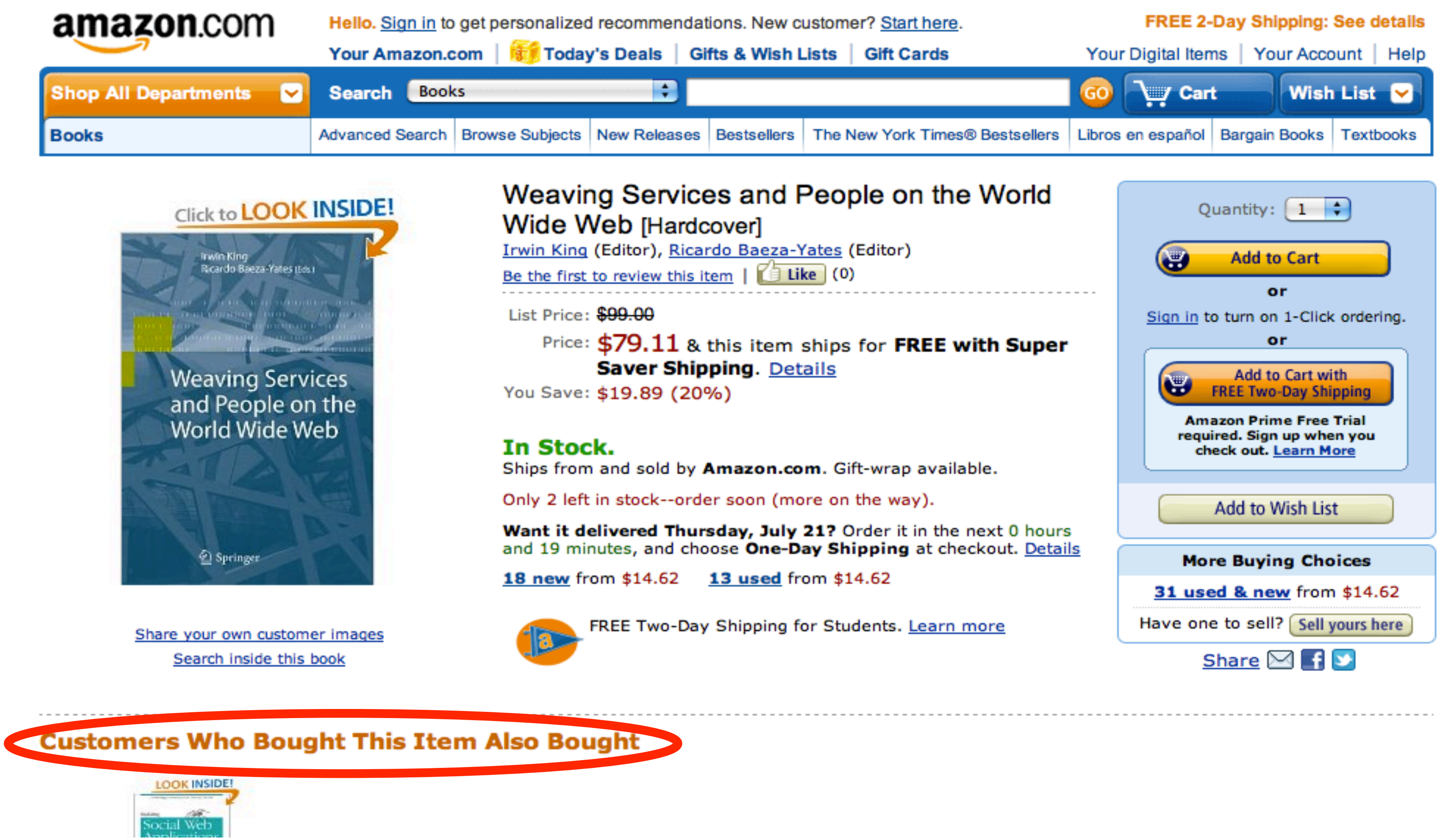

Consumer
Satisfaction
Company
Profit
Information


Information

Real Life Examples

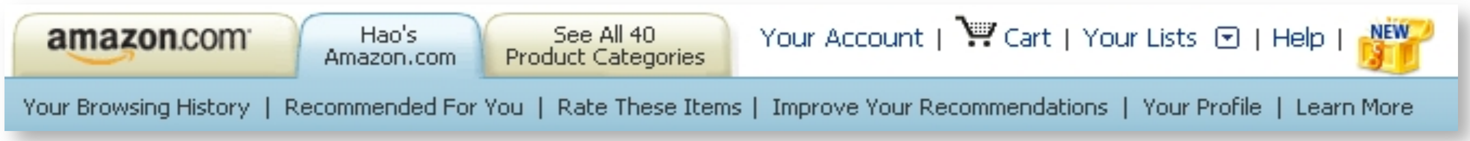


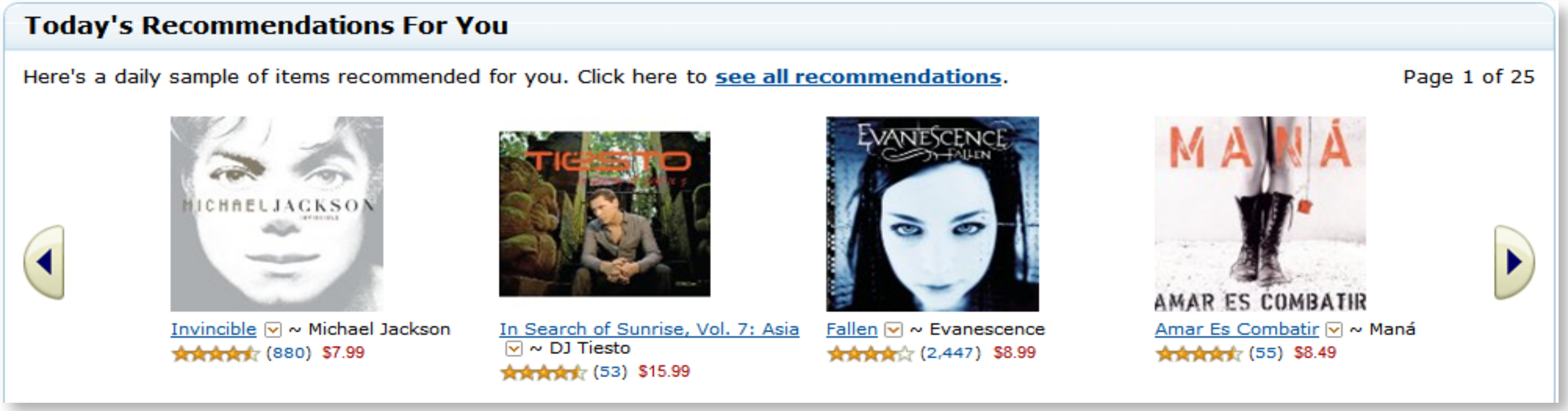
Real Life Examples



Real Life Examples

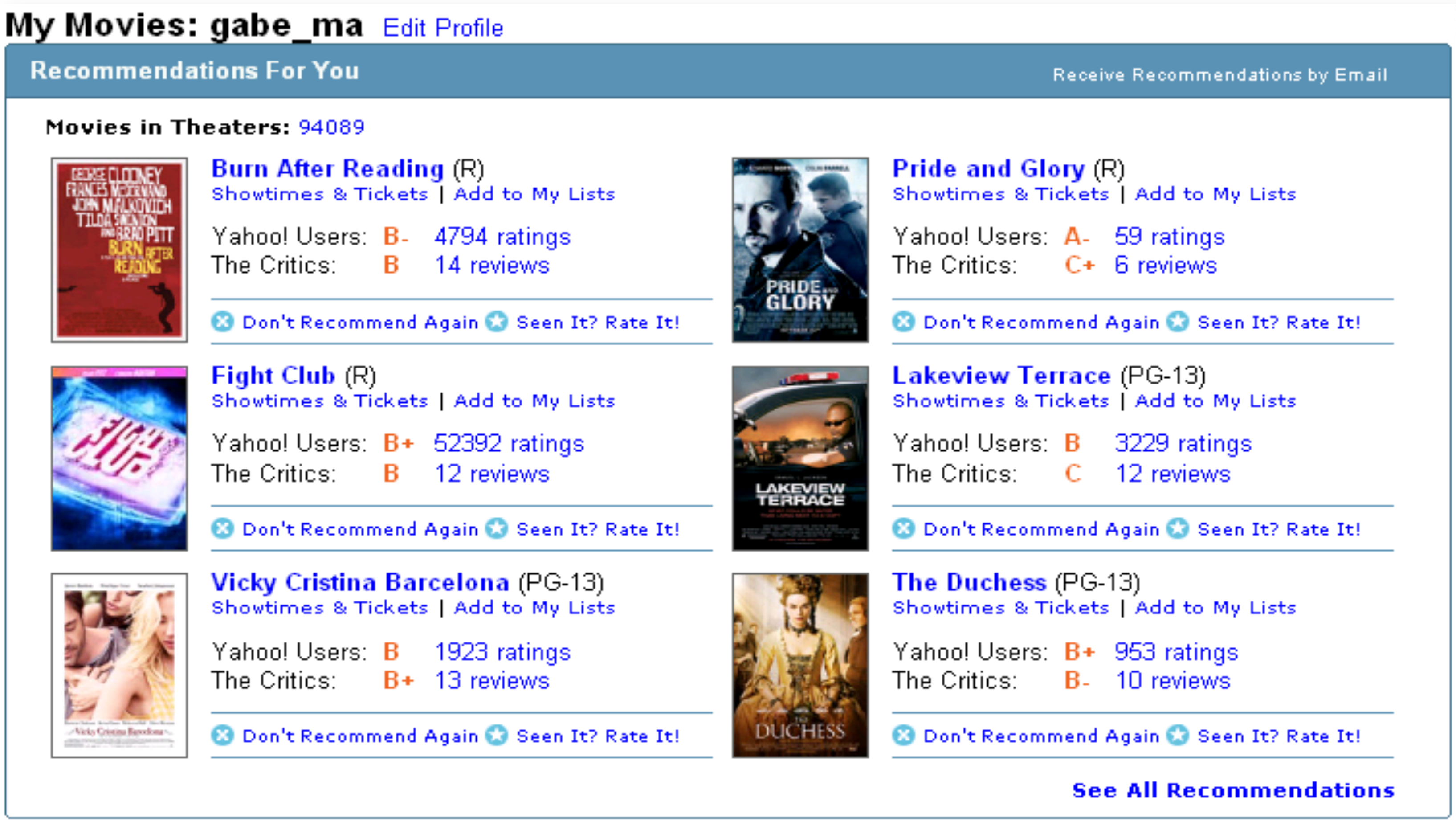
Real Life Examples





Real Life Examples

YAHOO! MOVIES

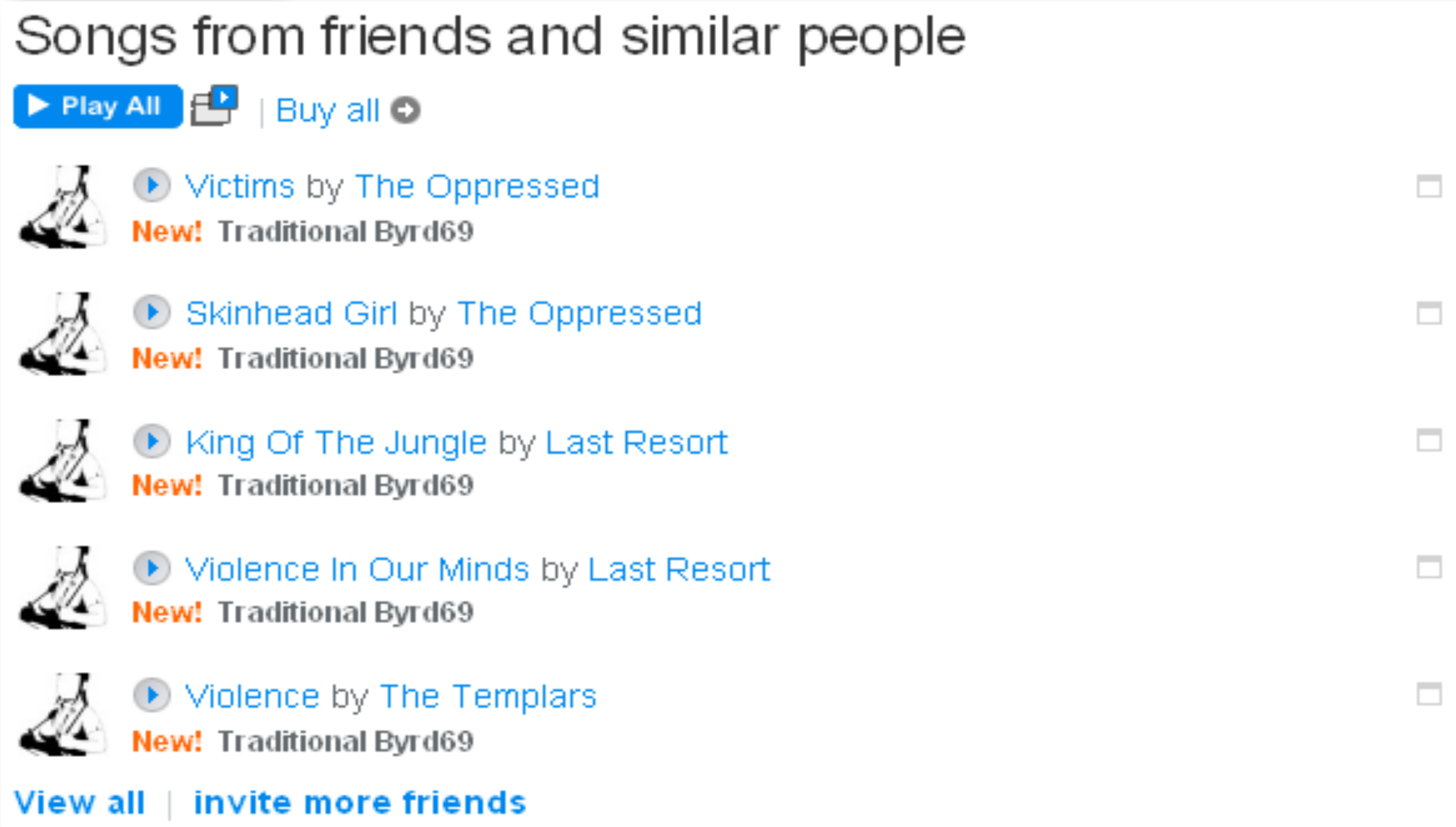


Real Life Examples

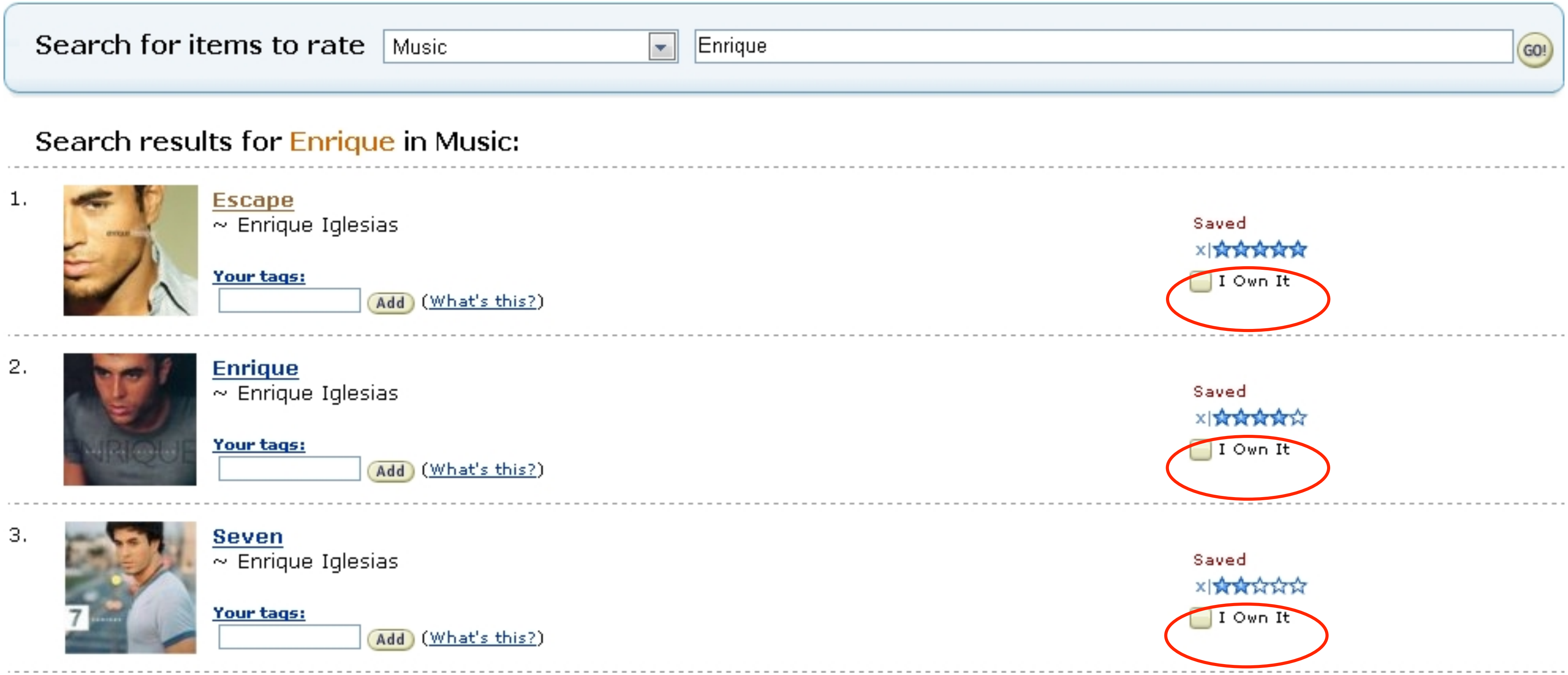
iTunes 8

Real Life Examples

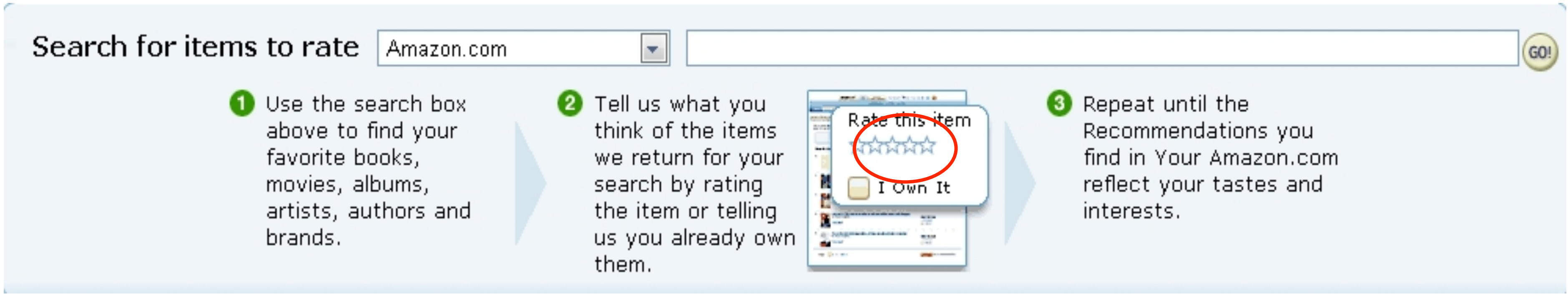
iLike

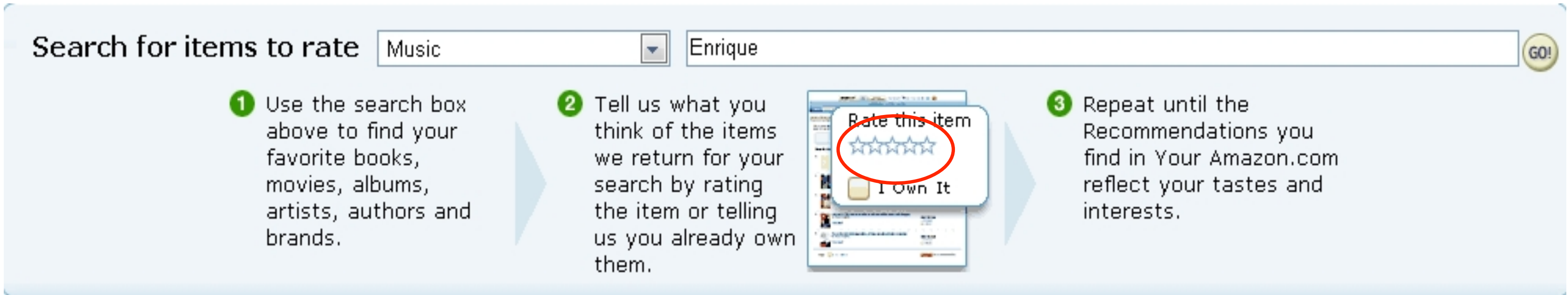


5-scale Ratings



5-scale Ratings





On The Menu

- Introduction
- Basic Techniques
  - Collaborative filtering
  - Matrix factorization
- Different Models
  - Social graph
  - Social ensemble
  - Social distrust
  - Website recommendation

Basic Approaches

- Content-based Filtering
  - Recommend items based on key-words
  - More appropriate for information retrieval
- Collaborative Filtering (CF)
  - Look at users with similar rating styles
  - Look at similar items for each item

Underling assumption: personal tastes are correlated--
Active users will prefer those items which the similar users prefer!

Framework

Items

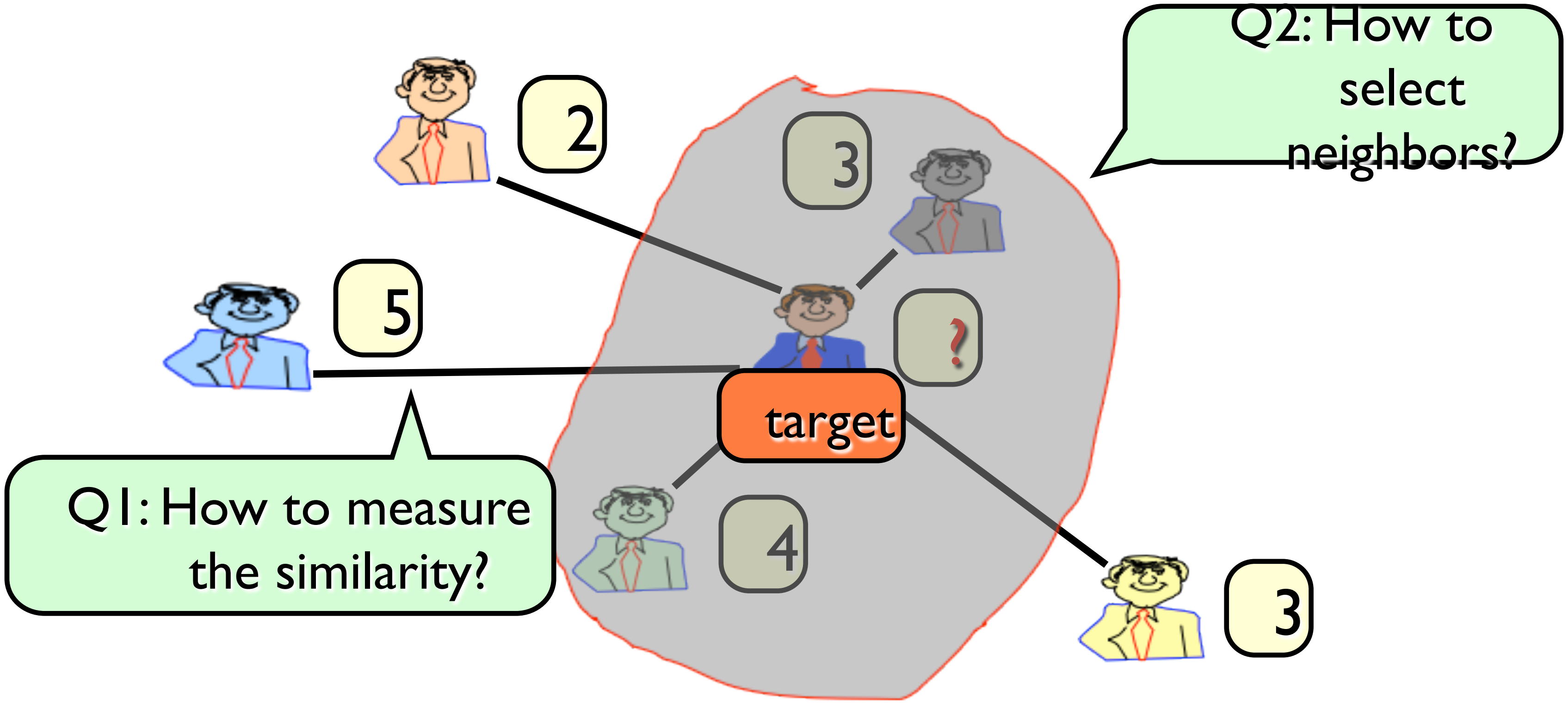
Users	i_1	i_2				i_j							i_m
u_1													
u_2	1	3		4		2		5			3	4	
u_i		3		4		r_{ij}	3	4		3	4		4
u_n	1			3	5	2		4	1			3	

- The tasks
  - Find the unknown rating!
  - Which item(s) should be recommended?

Collaborative Filtering

- Memory-based (Neighborhood-based)
  - User-based
  - Item-based
- Model-based
  - Clustering Methods
  - Bayesian Methods
  - Matrix Factorization
  - etc.

User-User Similarity



User-based Collaborative Filtering

Items

Users

u1													
u2	1	3		4		2		5			3	4	
u3													
u4		3		4			3	4		3	4		4
u5													
u6	1			3	5	2		4	1			3	

User-based Collaborative Filtering

Items

Users

u_1													
u_2	1	3		4		2		5			3	4	
u_3													
u_4		3		4			3	4		3	4		4
u_5													
u_6	1			3	5	2		4	1			3	

User-based Collaborative Filtering

Items

Users

u_1													
u_2	1	3		4		2		5			3	4	
u_3													
u_4		3		4			3	4		3	4		4
u_5													
u_6	1			3	5	2		4	1			3	

User-based Collaborative Filtering

Items

Users													
u_1													
u_2	1	3		4		2		5			3	4	
u_3													
u_4		3		4			3	4		3	4		4
u_5													
u_6	1			3	5	2		4	1			3	

User-based Collaborative Filtering

Items

Users

u_1													
u_2	1	3		4		2		5			3	4	
u_3													
u_4		3		4			3	4		3	4		4
u_5													
u_6	1			3	5	2		4	1			3	

User-based Collaborative Filtering

- Predict the ratings of active users based on the ratings of similar users found in the user-item matrix
  - Pearson correlation coefficient

$$w(a,i) = \frac{\sum_j (r_{aj} - \bar{r}_a)(r_{ij} - \bar{r}_i)}{\sqrt{\sum_j (r_{aj} - \bar{r}_a)^2 \sum_j (r_{ij} - \bar{r}_i)^2}} \qquad j \in I(a) \cap I(i)$$

  - Cosine measure

$$c(a,i) = \frac{r_a \cdot r_i}{||r_a||_2 * ||r_i||_2}$$

u_i	1	3		4		2		5			3	4	
u_a		3		4			3	4		3	4		4
	1			3	5	2		4	1			3	

Nearest Neighbor Approaches

[Sarwar, 00a]

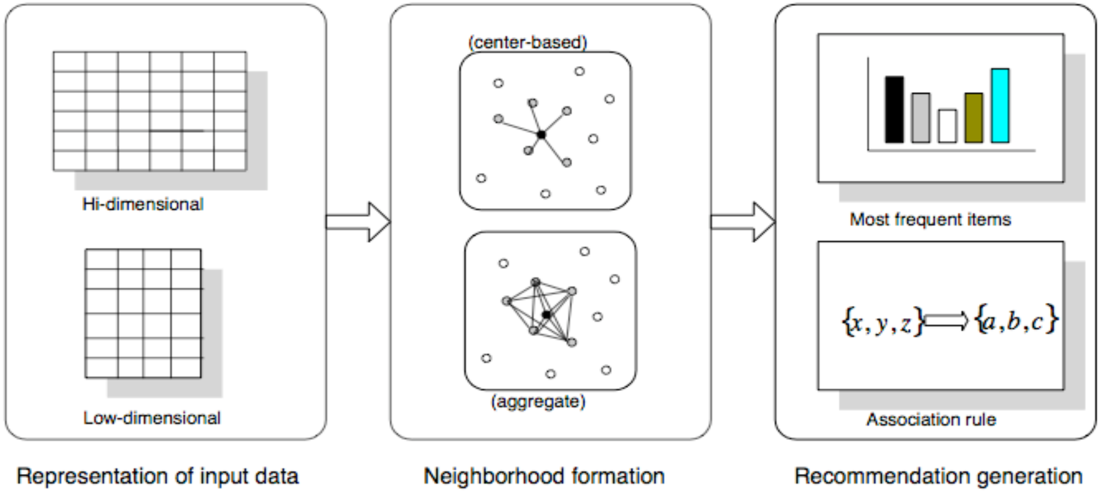


Figure 1: Three main parts of a Recommender System.

- Identify highly similar users to the active one
  - All with a measure greater than a threshold
  - Best K ones
- Prediction

$$r_{aj} = \bar{r}_a + \frac{\sum_i w(a,i)(r_{ij} - \bar{r}_i)}{\sum_i w(a,i)}$$

Collaborative Filtering

- Memory-based Method (Simple)
  - User-based Method [Xue et al., SIGIR '05]
  - Item-based [Deshpande et al., TOIS '04]
- Model-based (Robust)
  - Clustering Methods [Hkors et al, CIMCA '99]
  - Bayesian Methods [Chien et al., IWAIS '99]
  - Aspect Method [Hofmann, SIFIR '03]
  - Matrix Factorization [Sarwar et al., WWW '01]

Collaborative Filtering

- Memory-based (Neighborhood-based)
  - User-based
  - Item-based
- Model-based
  - Clustering Methods
  - Bayesian Methods
  - Matrix Factorization
  - etc.

Item-Item Similarity

- Search for similarities among items
- Item-Item similarity is more stable than user-user similarity

Correlation-based Method

[Sarwar, 2001]

- Same as in user-user similarity but on item vectors
- Pearson correlation coefficient
  - Look for users who rated both items

$$s_{ij} = \frac{\sum_u (r_{uj} - \bar{r}_j)(r_{ui} - \bar{r}_i)}{\sqrt{\sum_u (r_{uj} - \bar{r}_j)^2 \sum_u (r_{ui} - \bar{r}_i)^2}}$$

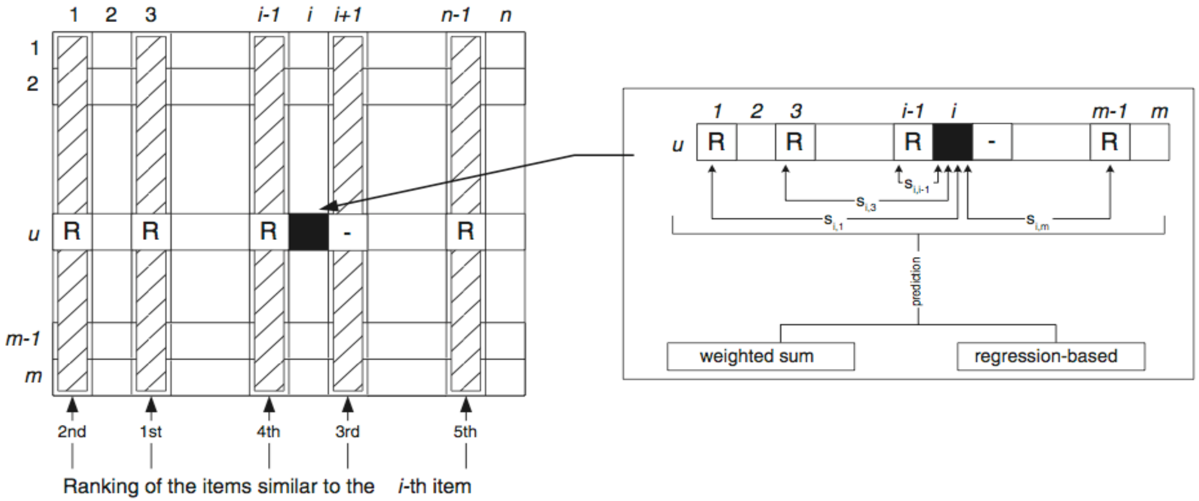
  - u: users rated both items

	i_1	i_2				i_i		i_j					i_m
u_1													
u_2	1	3		4		2		5			3	4	
u_i		3		4			3	4		3	4		4
u_n	1			3	5	2		4	1			3	

Correlation-based Method

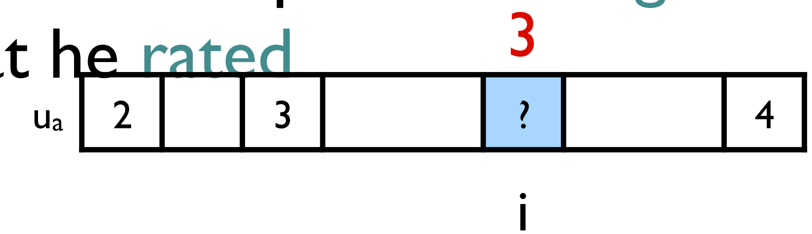
[Sarwar, 2001]

- Calculate item similarity, then determine its k-most similar items



- Predict rating for a given user-item pair as a weighted sum over similar items that he rated

$$r_{ai} = \frac{\sum_j s_{ij} r_{aj}}{\sum_j s_{ij}}$$



Collaborative Filtering

- Memory-based (Neighborhood-based)
  - User-based
  - Item-based
- Model-based
  - Clustering Methods
  - Bayesian Methods
  - Matrix Factorization
  - etc...

Matrix Factorization

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2		3		4		
u_2	4	3			5			
u_3	4		2				2	4
u_4								
u_5	5	1	2		4	3		
u_6	4	3		2	4		3	5

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2	**2.5**	3	**4.8**	4	**2.2**	**4.8**
u_2	4	3	**2.4**	**2.9**	5	**4.1**	**2.6**	**4.7**
u_3	4	**1.7**	2	**3.2**	**3.9**	**3.0**	2	4
u_4	**4.8**	**2.1**	**2.7**	**2.6**	**4.7**	**3.8**	**2.4**	**4.9**
u_5	5	1	2	**3.4**	4	3	**1.5**	**4.6**
u_6	4	3	**2.9**	2	4	**3.4**	3	5

$$U = \begin{bmatrix} 1.55 & 1.22 & 0.37 & 0.81 & 0.62 & -0.01 \\ 0.36 & 0.91 & 1.21 & 0.39 & 1.10 & 0.25 \\ 0.59 & 0.20 & 0.14 & 0.83 & 0.27 & 1.51 \\ 0.39 & 1.33 & -0.43 & 0.70 & -0.90 & 0.68 \\ 1.05 & 0.11 & 0.17 & 1.18 & 1.81 & 0.40 \end{bmatrix} \quad V = \begin{bmatrix} 1.00 & -0.05 & -0.24 & 0.26 & 1.28 & 0.54 & -0.31 & 0.52 \\ 0.19 & -0.86 & -0.72 & 0.05 & 0.68 & 0.02 & -0.61 & 0.70 \\ 0.49 & 0.09 & -0.05 & -0.62 & 0.12 & 0.08 & 0.02 & 1.60 \\ -0.40 & 0.70 & 0.27 & -0.27 & 0.99 & 0.44 & 0.39 & 0.74 \\ 1.49 & -1.00 & 0.06 & 0.05 & 0.23 & 0.01 & -0.36 & 0.80 \end{bmatrix}$$

Matrix Factorization

- Matrix Factorization in Collaborative Filtering
  - To fit the product of two (low rank) matrices to the observed rating matrix.
  - To find two latent user and item feature matrices.
  - To use the fitted matrix to predict the unobserved ratings.

$$Y \approx UV = \begin{pmatrix} u_{11} & \cdots & u_{1k} \\ \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{mk} \end{pmatrix} \begin{pmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{k1} & \cdots & v_{kn} \end{pmatrix}$$

User-specific latent feature vector

Item-specific latent feature column vector

Matrix Factorization

- Optimization Problem
  - Given a *m x n* rating matrix *R*, to find two matrices $U \in \mathbb{R}^{l \times m}$ and $V \in \mathbb{R}^{l \times n}$

$$R \approx U^T V,$$

where $l < \min(m, n)$, is the number of factors

Matrix Factorization

- Models
  - SVD-like Algorithm
  - Regularized Matrix Factorization (RMF)
  - Probabilistic Matrix Factorization (PMF)
  - Non-negative Matrix Factorization (NMF)

SVD-like Algorithm

- Minimizing

$$\frac{1}{2}||R - U^T V||_F^2,$$

- For collaborative filtering

$$\min_{U,V} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2$$

where I_{ij} is the indicator function that is equal to 1 if user u_i rated item v_j and equal to 0 otherwise.

Regularized Matrix Factorization

- Minimize the loss based on the observed ratings with regularization terms to avoid over-fitting problem

$$\min_{U,V} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2 + \boxed{\frac{\lambda_1}{2}||U||_F^2 + \frac{\lambda_2}{2}||V||_F^2}$$

Regularization terms

where $\lambda_1, \lambda_2 > 0$

- The problem can be solved by simple gradient descent algorithm.

Regularized Matrix Factorization

- Algorithm for RMF
  - Not convex & local optimal
  - Gradient-decent algorithm
    - Gradient computation with randomly initialized *U* and *V*

$$\frac{\partial L}{\partial \boldsymbol{u}_{il}} = \lambda \boldsymbol{u}_{il} - \sum_{j|(i,j)\in S} (y_{ij} - \widehat{y_{ij}})\boldsymbol{v}_{jl}$$

$$\frac{\partial L}{\partial \boldsymbol{v}_{il}} = \lambda \boldsymbol{v}_{il} - \sum_{j|(i,j)\in S} (y_{ij} - \widehat{y_{ij}})\boldsymbol{u}_{jl}$$

    - Update *U* and *V* alternatively

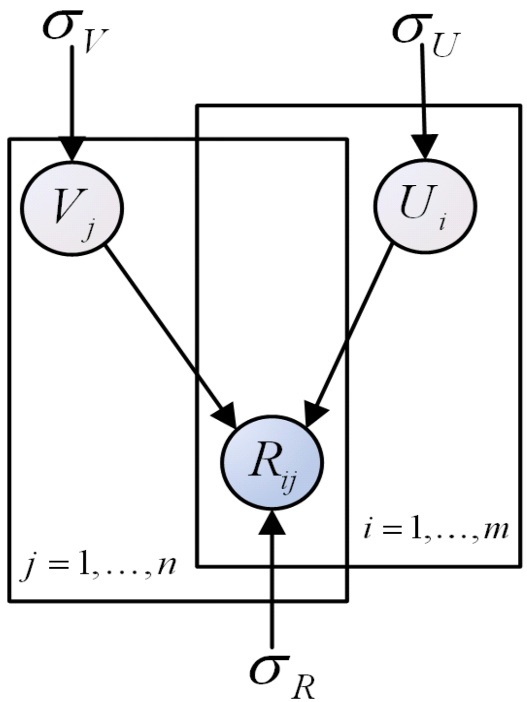
$$\boldsymbol{u}_{il}^{(t+1)} = \boldsymbol{u}_{il}^{(t)} - \tau \frac{\partial L}{\partial \boldsymbol{u}_{il}^{(t)}}$$

$$\boldsymbol{v}_{jl}^{(t+1)} = \boldsymbol{v}_{jl}^{(t)} - \tau \frac{\partial L}{\partial \boldsymbol{v}_{jl}^{(t)}}$$

τ is the step size of gradient decent.

Regularized Matrix Factorization

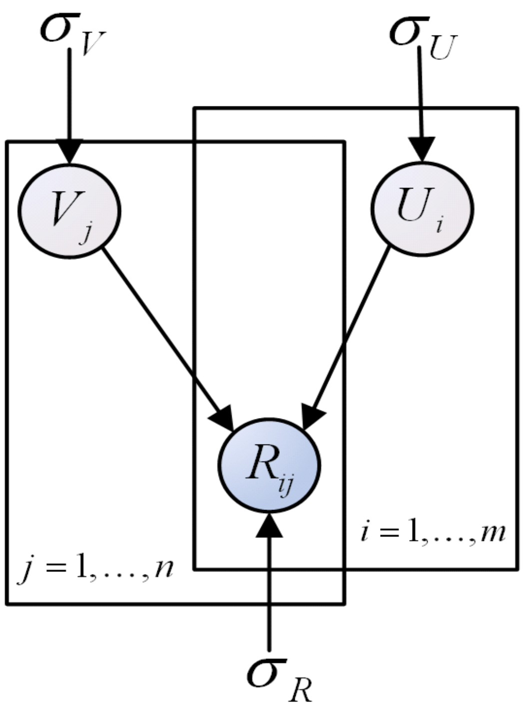
- PMF
  - Define a conditional distribution over the observed ratings as:



$$p(R|U,V,\sigma_R^2) = \prod_{i=1}^{m}\prod_{j=1}^{n}\left[\mathcal{N}\left(R_{ij}|g(U_i^T V_j),\sigma_R^2\right)\right]^{I_{ij}^R}$$

Regularized Matrix Factorization

- PMF
  - Assume zero-mean spherical Gaussian priors on user and item feature:

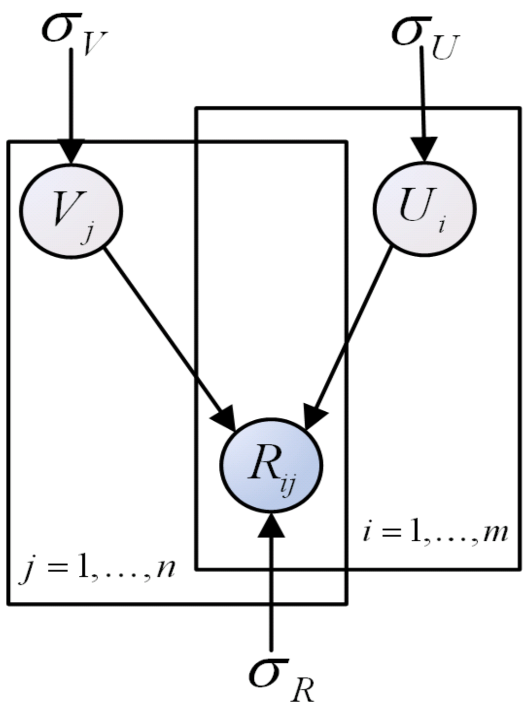


$$p(U|\sigma_U^2) = \prod_{i=1}^{m} \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I})$$

$$p(V|\sigma_V^2) = \prod_{j=1}^{n} \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I})$$

Regularized Matrix Factorization

- PMF
  - Bayesian inference



$$
\begin{aligned}
p(U, V | R, \sigma_R^2, \sigma_U^2, \sigma_V^2) &\propto p(R|U, V, \sigma_R^2) p(U|\sigma_U^2) p(V|\sigma_V^2) \\
&= \prod_{i=1}^{m} \prod_{j=1}^{n} \left[\mathcal{N} \left(R_{ij} | g(U_i^T V_j), \sigma_R^2 \right) \right]^{I_{ij}^R} \\
&\times \prod_{i=1}^{m} \mathcal{N}(U_i | 0, \sigma_U^2 \mathbf{I}) \times \prod_{j=1}^{n} \mathcal{N}(V_j | 0, \sigma_V^2 \mathbf{I}).
\end{aligned}
$$

RMF and PMF

- PMF is the probabilistic interpretation of RMF
- PMF and RMF have the same optimization objective function

Non-negative Matrix Factorization

- NMF
  - Non-negative constraints on all entries of matrices U and V

$$\begin{pmatrix} -10 & \cdots & u_{1d} \\ \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{md} \end{pmatrix} \times \begin{pmatrix} -10 & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{d1} & \cdots & v_{dn} \end{pmatrix} \Rightarrow$$

Non-negativity Constraint

(-10)*(-10)=100

Non-negative Matrix Factorization

- NMF
  - Given an observed matrix Y, to find two non-negative matrices U and V
  - Two types of loss functions
    - Squared error function

$$\sum_{ij} (R_{ij} - U_i^T V_j)^2$$

    - Divergence

$$D(R||U^T V) = \sum_{ij} (R_{ij} \log \frac{R_{ij}}{U_i^T V_j} - R_{ij} + U_i^T V_j)$$

- Solving by multiplicative updating rules

Non-negative Matrix Factorization

- Multiplicative updating rules
  - For divergence objective function

$$u_{il} \leftarrow u_{il} \frac{\sum_j v_{jl} y_{ij} / (\widehat{y_{ij}})}{\sum_a v_{al}}$$

$$v_{il} \leftarrow v_{il} \frac{\sum_j u_{jl} y_{ij} / (\widehat{y_{ij}})}{\sum_a u_{al}}$$