Web Personalization & Recommender Systems

COSC 488

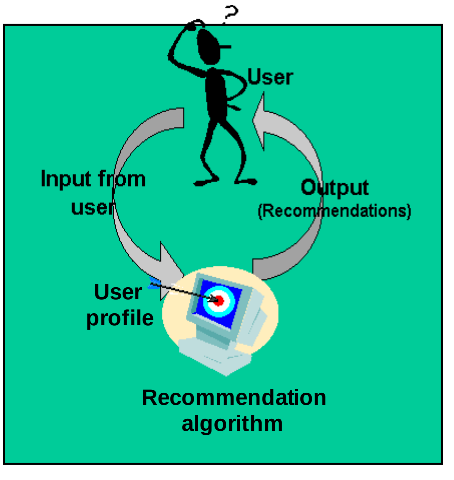
Slides are based on:

- Bamshad Mobasher, Depaul University

- Recent publications: see the last page (Reference section)

Web Personalization & Recommender Systems

- Most common type of personalization: Recommender systems



2

Recommender Systems

"Recommender systems are information filtering systems where users are recommended "relevant" information items (products, content, services) or social items (friends, events) at the right context at the right time with the goal of pleasing the user and generating revenue for the system. Recommender systems are typically discussed under the umbrella of "People who performed action X also performed action Y" where the action X and Y might be search, view or purchase of product, or seek a friend or connection."

Neel Sundaresan
eBay Research Labs
RecSys'11

3

RecSys'11- eBay

- eBay Example:
  - Over 10 million items listed for sale daily
  - Items are listed in explicitly defined hierarchy of categories
  - Over 30,000 nodes in this category tree. Only a fraction of the items are cataloged.
  - Hundreds of millions of searches are done on a daily basis.
  - Language gap between buyers and sellers in search
  - Recommender system tries to fill-in the language gap using knowledge mined from buyer/seller
    - Unlike a typical Web search, context from user behavior is used (user query, history of past queries,…)
    - Example: Identifying query relationships (within a session)
      - Q1: "Apple ipod mp3 player"
      - Q2: "creative mp3 player"
        - » Using co-occurrence: apple ipod & crative are related BUT apple ipod and apple dishes are not!

4

The Recommendation Task

- Basic formulation as a prediction problem

> Given a profile P_u for a user u, and a target item i_t, predict the preference score of user u on item i_t

- Typically, the profile P_u contains preference scores by u on some other items, $\{i_1, \ldots, i_k\}$ different from i_t
  - preference scores on $i_1, \ldots, i_k$ may have been obtained explicitly (e.g., movie ratings) or implicitly (e.g., time spent on a product page or a news article)

5

Notes on User Profiling

- Utilizing user profiles for personalization assumes
  - 1) past behavior is a useful predictor of the future behavior
  - 2) wide variety of behaviors amongst users
- Basic task in user profiling: Preference elicitation
  - May be based on explicit judgments from users (e.g. ratings)
  - May be based on implicit measures of user interest
- Automatic user profiling
  - Use machine learning techniques to learn models of user behavior, preferences
  - May include keywords, categories, …
  - May build a model for each specific user or build group profiles

Similarity of profile(s) to incoming documents, news, advertisements are measured by comparing the document vector to the profile s' indicies

6

Common Recommendation Techniques

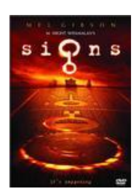
- **Rule-Based (Knowledge-Based) Filtering**
  - Provides recommendations to users based on predefined (or learned) rules
  - age(x, 25-35) and income(x, 70-100K) and children(x, >=3) → recommend(x, Minivan)
- **Content-Based Filtering**
  - Gives recommendations to a user based on items with "similar content" in the user's profile
- **Collaborative Filtering**
  - Gives recommendations to a user based on preferences of "similar" users
  - Preferences on items may be explicit or implicit

7

Content-Based Recommenders

- **Predictions for unseen (target) items are computed based on their similarity (in terms of content) to items in the user profile.**
- **E.g., user profile P_u contains**

recommend highly: 

and recommend "mildly": 

8

Content-Based Recommender Systems

9

Content-Based Recommenders: Personalized Search

- How can the search engine determine the "user's context"?

Query: "Madonna and Child"

?

?

- **Need to "learn" the user profile:**
  - User is an art historian?
  - User is a pop music fan?

10

Content-Based Recommenders

- **Similarity of user profile to each item**
  Example:
  - User profile: vector of terms from user high ranked documents
  - Use cosine similarity to calculate similarity between profile & documents
- **Disadvantage:**
  - Unable to recommend new items (different to profile)

11

Collaborative Recommender Systems

12

Collaborative Systems: Approaches

User-based & Item-based are most commonly used:

- User-based (neighborhood-based): *[Resnick et al. 1994; Shardanand 1994],*
  1) Calculate the similarity between the active user and the rest of the users.
     Pearson correlation, cosine vector space, …
  (2) Select a subset of the users (neighborhood) according to their similarity with the active user.
     - Similarity threshold, or *N* most similar
  (3) Compute the prediction using the neighbor ratings.

  Can Cluster the users to reduce the sparsity & improve scalability

- Item-based: similar to the user-based but, instead of looking for neighbors among users, they look for similar items.
  - Advantage over user-based: more static, thus can be calculated off-line

14

Collaborative System (User-based)

	Item1	Item 2	Item 3	Item 4	Item 5	Item 6	Correlation with Alice
Alice	5	2	3	3		?	
User 1	2		4		4	1	-1.00
User 2	2	1	3		1	2	0.33
User 3	4	2	3	2		1	.90
User 4	3	3	2		3	1	0.19
User 5		3		2	2	2	1.00
User 6	5	3		1	3	2	[illegible]
User 7		5		1	5	1	[illegible]

Prediction

Best match

Compare target user with all user records (Using k-nearest neighbor with k = 1)

15

Using Clusters for Personalization

Original Session/user data

	A.html	B.html	C.html	D.html	E.html	F.html
user0	1	1	0	0	0	1
user1	0	0	1	1	0	0
user2	1	0	0	1	1	0
user3	1	1	0	0	0	1
user4	0	0	1	1	0	0
user5	1	0	0	1	1	0
user6	1	1	0	0	0	1
user7	0	0	1	1	0	0
user8	1	0	1	1	1	0
user9	0	1	1	0	0	1

Result of Clustering

		A.html	B.html	C.html	D.html	E.html	F.html
Cluster 0	user 1	0	0	1	1	0	0
	user 4	0	0	1	1	0	0
	user 7	0	0	1	1	0	0
Cluster 1	user 0	1	1	0	0	0	1
	user 3	1	1	0	0	0	1
	user 6	1	1	0	0	0	1
	user 9	0	1	1	0	0	1
Cluster 2	user 2	1	0	0	1	1	0
	user 5	1	0	0	1	1	0
	user 8	1	0	1	1	1	0

PROFILE 0 (Cluster Size = 3)

1.00 C.html
1.00 D.html

PROFILE 1 (Cluster Size = 4)

1.00 B.html
1.00 F.html
0.75 A.html
0.25 C.html

PROFILE 2 (Cluster Size = 3)

1.00 A.html
1.00 D.html
1.00 E.html
0.33 C.html

Given an active session A → B, the best matching profile is Profile 1. This may result in a recommendation for page F.html, since it appears with high weight in that profile.

Item-based Collaborative Filtering

- **Find similarities among the items based on ratings across users**
  - Often measured based on a variation of Cosine measure
- **Prediction of item I for user *a* is based on the past ratings of user *a* on items similar to *i*.**

	Star Wars	Jurassic Park	Terminator 2	Indep. Day
Sally	7	6	3	7
Bob	7	4	4	6
Chris	3	7	7	2
Lynn	4	4	6	2
Karen	7	4	3	?

- **Suppose:** sim(Star Wars, Indep. Day) > sim(Jur. Park, Indep. Day) > sim(Termin., Indep. Day)
- **Predicted rating for Karen on Indep. Day will be 7, because she rated Star Wars 7**
  - That is if we only use the most similar item
  - Otherwise, we can use the k-most similar items and again use a weighted average

Collaborative Filtering (Item-based)

Pair-wise comparison of items across users

	Item1	Item 2	Item 3	Item 4	Item 5	Item 6
Alice	5	2	3	3		?
User 1	2		4		4	1
User 2	2	1	3		1	2
User 3	4	2	3	2		1
User 4	3	3	2		3	1
User 5		3		2	2	2
User 6	5	3		1	3	2
User 7		5		1	5	1
Item similarity	0.76	0.79	0.60	0.71	0.75	

Prediction

Best match

Collaborative Systems (Cont'd)

- Hybrid: combines the item ratings of similar users to the active user , the ratings of the active user on similar items, the ratings of similar items by similar users, semantic information.

- SVD-based: Matrix factorization techniques (variations exist):
  - each item is represented as a set of features (aspects)
  - each user as a set of values indicating his/her preference for the various aspects of the items.
  - The number of features to consider, *K, is a model parameter.*
  - *The rating prediction is:* $P_{ij} = x_i^T y_j$

- Tendency-based: calculates tendency of users / items *[ACM Transactions on the Web, 2011]*
  - tendency of a user: *the average difference between* his/her ratings and the item mean.
  - tendency of an item: *the average difference between an item's rating by users and the users'* mean rating; (that is if the users find an item specially good or bad).

$$t_u = \frac{\sum_{i \in I_u} v_{ui} - \bar{v}_i}{|I_u|} \qquad t_i = \frac{\sum_{u \in U_i} v_{ui} - \bar{v}_u}{|U_u|}$$

Semantically Enhanced Hybrid Recommendation

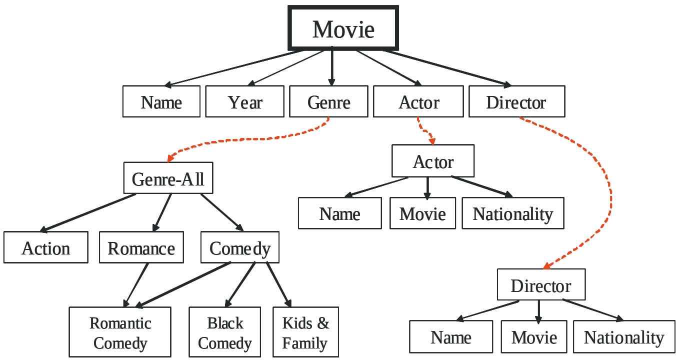
- **Sample extension of the item-based algorithm**
  - Use a combined similarity measure to compute item similarities:

$$CombinedSim(i_p, i_q) = (1 - \alpha) \cdot SemSim(i_p, i_q) + \alpha \cdot RateSim(i_p, i_q)$$

  - where,
    - *SemSim* is the similarity of items i_p and i_q based on semantic features (e.g., keywords, attributes, etc.); and
    - *RateSim* is the similarity of items i_p and i_q based on user ratings (as in the standard item-based CF)
  - α is the semantic combination parameter:
    - $\alpha = 1$ → only user ratings; no semantic similarity
    - $\alpha = 0$ → only semantic features; no collaborative similarity

Semantically Enhanced CF

- **Movie data set**
  - Movie ratings from the movielens data set
  - Semantic info. extracted from IMDB based on the following ontology



Hybrid Recommender Systems

	A.html	B.html	C.html	D.html	E.html
user1	1	0	1	0	1
user2	1	1	0	0	1
user3	0	1	1	1	0
user4	1	0	1	1	1
user5	1	1	0	0	1
user6	1	0	1	1	1

User Pageview matrix UP

Feature-Pageview Matrix FP

	A.html	B.html	C.html	D.html	E.html
web	0	0	1	1	1
data	0	1	1	1	0
mining	0	1	1	1	0
business	1	1	0	0	0
intelligence	1	1	0	0	1
marketing	1	1	0	0	1
ecommerce	0	1	1	0	0
search	1	0	1	0	0
information	1	0	1	1	1
retrieval	1	0	1	1	1

24

User-Feature Matrix

Note that: $UF = UP \times FP^T$

	web	data	mining	business	intelligence	marketing	ecommerce	search	information	retrieval
user1	2	1	1	1	2	2	1	2	3	3
user2	1	1	1	2	3	3	1	1	2	2
user3	2	3	3	1	1	1	2	1	2	2
user4	3	2	2	1	2	2	1	2	4	4
user5	1	1	1	2	3	3	1	1	2	2
user6	3	2	2	1	2	2	1	2	4	4

Example: users 4 and 6 are more interested in concepts related to Web information retrieval, while user 3 is more interested in data mining.

25

Some facts & observations to motivate Trust-based approaches…*[Golbeck, ACM Transactions on the Web, Sept 2009]*

- Collaborative filtering (CF)
  - Effective when users have a common set of rated items
  - Difficult to compute similarity between users, if ratings are sparse
  - Use *overall similarity* of user profiles to make recommendations

- Online social networks
  - A limited snapshot of the users and their interactions
  - Typically, a list of friends for each user
  - Few show the last date that a given user was active, and none show more detailed information about a user's history of interaction with the Web site.
  - No networks publicly share a history of interactions between users
  - Communications are kept private, and the formation or removal of relationships are not shown or recorded in a user's profile.

26

Collaborative Systems (Cont'd)

- Trust-based approaches: consider *social trust* relationships between users

[Andersen et al. 2008; Bedi et al. 2007; Ma et al. 2008; Massa and Avesani 2004; 2007], [Ziegler and Golbeck, 2006],…

  - Explicit trust – specified by users
  - Implicit trust – calculated via Pearson, friendship factors,…

Trust & Profile Similarity

- Strong and significant correlation between trust and user similarity (the more similar two people, the greater the trust between them). [Ziegler and Golbeck, 2006]
- Methods to infer trust on social network:
  - Explicitly defined trust
  - A function of corrupt vs. valid files that a peer provides (in P2P). *EigenTrust* algorithm [Kamvar et al. 2004]
  - The perspective of authoritative nodes.
- Example -- "Recommended Rating" that is personalized for each user *[ACM Transactions on the Web, 2011]*

  - Alice trusts Bob 9.

  - Alice trusts Chuck 3.

  - Bob rates the movie "Jaws" with 4 stars.

  - Chuck rates the movie "Jaws" with 2 stars.

  Alice's recommended rating for "Jaws" is calculated as follows:

  $t_{Alice\rightarrow Bob} * r_{Bob\rightarrow Jaws} + t_{Alice\rightarrow Chuck} * r_{Chuck\rightarrow Jaws} / t_{Alice\rightarrow Bob} + t_{Alice\rightarrow Chuck}$

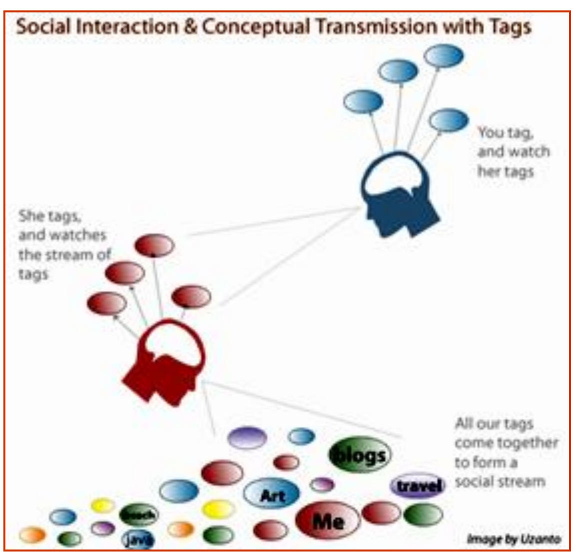
  $= ((9 * 4) + (3 * 2)) / (9 + 3) = 3.5.$

Collaborative Filtering (Tag-based)

- Social Tagging
  - Social exchange goes beyond collaborative filtering
  - people add free-text tags to their content
    - Del.icio.us, Flickr, Last.fm
- Data record: <user, item, tag>
  - Tag: user-item interaction can be annotated by multiple tags, indicating the reason of user's interest

29

Folksonomies



30

References

- *ACM Transactions on the Web, 2011*
- *Golbeck, ACM Transactions on the Web, Sept 2009*
- *Andersen et al. 2008;*
- *Bedi et al. 2007;*
- *Ma et al. 2008;*
- *Massa and Avesani 2004; 2007*
- Ziegler and Golbeck, 2006,…
- *Resnick et al. 1994;*
- *Shardanand 1994*