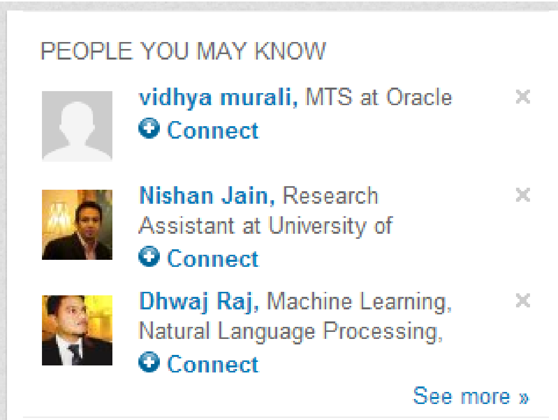
Recommendation Systems

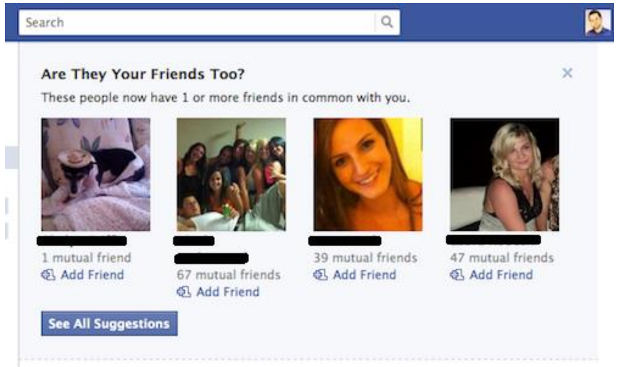
Slides borrowed (and modified) from http://www.slideshare.net/idoguy/social-recommender-systems-tutorial-www-2011-7446137

Recommendation Systems Everywhere

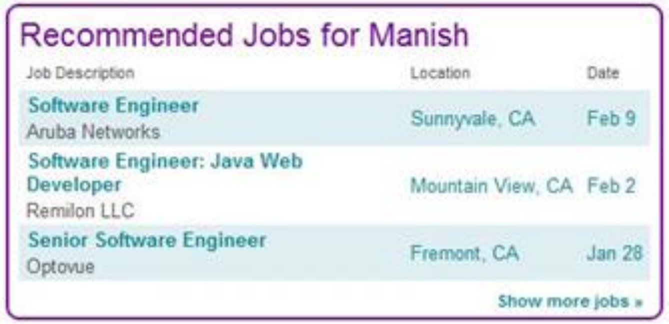
LinkedIn People Recommendations



Facebook People Recommendations

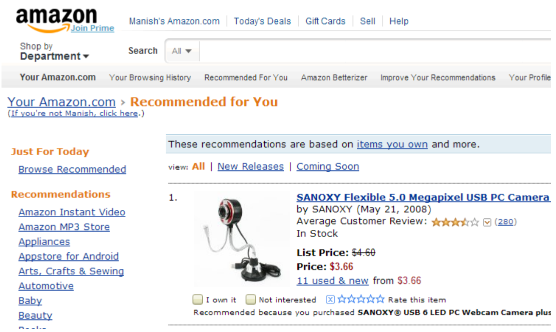


HotJobs Job Recommendations

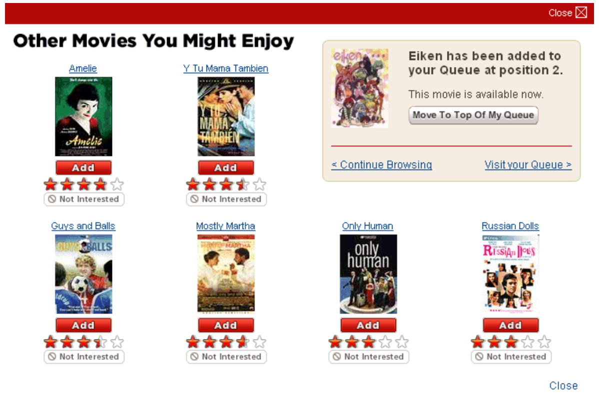


Bing Query Recommendations

Amazon Product Recommendations



Netflix Movie Recommendations



Social Overload

- Information Overload
  - Blogs, microblogs, forums, wikis, news, bookmarked webpages, photos, videos, etc.
- Interaction Overload
  - Friends, followers, followees, commenters, co-members, voters, likers, taggers, review writers, etc.

Social Recommender Systems

- Recommender Systems that target the social media domain
- Aim at coping with the challenge of social overload by presenting the most attractive and relevant content
- Also aim at increasing adoption and engagement
- Often apply personalization techniques

Collaborative Filtering

- In the real world we seek advices from our trusted people (friends, colleagues, experts)
- CF automates the process of “word-of-mouth”
  - Weight all users with respect to similarity with the active user.
  - Select a subset of the users (neighbors) to use as recommenders
  - Predict the rating of the active user for specific items based on its neighbors’ ratings
  - Recommend items with maximum prediction

User-based CF Algorithm

- The User x Item Matrix

	Shrek	Snow-white	Superman
Alice	Like	Like	Dislike
Bob	?	Dislike	Like
Chris	Like	Like	Dislike
John	Like	Like	?

- Shall we recommend Superman for John?
- John’s taste is similar to both Chris and Alice tastes ⇒ Do not recommend Superman to John

User-based CF Algorithm

- Let R be the rating matrix
  - r_{uj} is then the vote of user u for item j

- I_u be the set of items for which user u has provided the rating

- Voting
  - Mean vote for user u: $\bar{r_u} = \frac{1}{|I_u|} \sum_{i \in I_u} r_{ui}$
  - Prediction rating: $p_{uj} = \bar{r_u} + \gamma \sum_{v=1}^{n} w(u, v)(r_{vj} - \bar{r_v})$
    - $w(u, v)$ = similarity between users u and v
    - γ is a normalization constant $\gamma = \frac{1}{\sum_{v=1}^{n} w(u,v)}$

User-based CF Algorithm

- Cosine based similarity between users
  - $w(u,v) = \frac{\sum_{i \in I} r_{ui} r_{vi}}{\sqrt{\sum_{i \in I} r_{ui}^2} \sqrt{\sum_{i \in I} r_{vi}^2}}$

- Pearson based similarity between users
  - $w(u,v) = \frac{\sum_{i \in I} (r_{ui} - \overline{r_u})(r_{vi} - \overline{r_v})}{\sqrt{\sum_{i \in I} (r_{ui} - \overline{r_u})^2} \sqrt{\sum_{i \in I} (r_{vi} - \overline{r_v})^2}}$

CF - Practical Challenges

- Ratings data is often sparse, and pairs of users with few co-ratings are prone to skewed correlations
- Fails to incorporate agreement about an item in the population as a whole
  – Agreement about a universally loved item is much less important than agreement for a controversial item
    - Some algorithms account for global item agreement by including weights inversely proportional to an item's popularity
- Calculating a user's perfect neighborhood is expensive – requiring comparison against all other users
  – Sampling: a subset of users is selected prior to prediction computation
  – Clustering: can be used to quickly locate a user's neighbors

Enhancing CF with Friends

- The user's network of friends and people of interest has become more accessible in the Web 2.0 era (Facebook, LinkedIn, Twitter,...)
- Such social relationships can be very effective for recommendation compared to traditional CF
  - Recommendation from people the user knows
  - Spare explicit feedback such as ratings
  - Effective for new users
- Various works have shown the effectiveness of friend-based recommendation over CF, e.g.:
  - Movie and book recommendation - Comparing Recommendations Made by Online Systems and Friends [Sinha & Swearingen, 2001]
  - Friends as trusted recommenders for movies [Golbeck, 2006]
  - Club recommendation within a German SNS - Collaborative Filtering vs. Social Filtering [Groh & Ehmig, Group 2007]

Item-Based Nearest Neighbor Algorithms

- The transpose of the user-based algorithms
  - Generate predictions based on similarities between items
  - The prediction for an item is based on the user's ratings for similar items

	Shrek	Snow-white	Superman
Alice	Like	Like	Dislike
Bob	?	Dislike	Like
Chris	Like	Like	Dislike
John	Like	Like	?

- Bob dislikes Snow-white (which is similar to Shrek) ⇒ do not recommend Shrek to Bob
- Predicted rating: $p_{uj} = \gamma \sum_{i=1}^{m} w(i,j) r_{ui}$
- Traverse over all m items rated by user u and measure their rating, averaged by their similarity to the predicted item
- w(i,j) is a measure of item similarity - usually the cosine measure
- Average correction is not needed because the component ratings are all from the same target user

Dimensionality Reduction Algorithms

- Reduce domain complexity by mapping the item space to a smaller number of underlying "dimensions"
  – Represent the latent topics present in those items
  – Improve accuracy in predicting ratings in most cases
  – Reduce run-time performance needs and lead to larger numbers of co-rated dimensions
- Popular techniques: Singular Value Decomposition and Principal Component Analysis
  – Require an extremely expensive offline computation step to generate the latent dimensional space

Hybrid Recommendation Methods

- Any Recommendation approach has pros and cons
  - e.g. CF & CB both suffer from the cold start problem
  - but CF can recommend “outside the box” compared to Content-based approaches
- Hybrid recommender system combines two or more techniques to gain better performance with fewer drawbacks
- Hybrid methods:
  - Weighted: scores of several recommenders are combined together
  - Switching: switch between recommenders according to the current situation
  - Mixed: present recommendations that are coming from several recommenders
  - Cascade: One recommender refines the recommendations given by another

The Cold Start Problem

- The Cold Start problem concerns the issue where the RS cannot draw inferences for users or items for which it has not yet gathered sufficient information
- New items
  – e.g., a newly created document w/o tags or bookmarks
  – e.g., a newly created community w/o members
- New users
  – e.g., a user that has just signed up to a new site
  – e.g., a new member or employee
- Typically addressed by applying a hybrid approach

The Cold Start Problem of New Items

- Traditional CF systems are based on item ratings
  - Until rated by a substantial number of users, the system will not be able to recommend the item
- a.k.a the “early rater” problem – first person to rate an item gets little benefit
- Same for implicit feedback over items – clicks, searches, comments, tags
- Even more acute for activity streams, where items quickly come and go
- Typically addressed by integrating CB similarity measurements
  - Recommendation based on the data of older similar items

The Cold Start Problem for New Users

- Sometimes also referred to as the “New User Problem”
- User needs to rate sufficient items for a CB recommender to really understand the user’s preferences
- Mitigated by CF – similar users who rated more items can yield more recommendations
- Traditional CF still faces an issue if the user did not provide any explicit feedback (or very small amount of feedback)
- Typically resolved through building a user profile by integrating other user activity (implicit feedback)
  - Browsing history, click-through data, searches
- Social media introduces new ways to learn about the user from external sources
  - Friends (“social filtering”), tags, communities, …
  - More public information which is less sensitive to privacy issues

Trust in Recommendation (by Explanations)

- **MoviExplain:** A Recommender System with Explanations (Symeonidis 09)
- Good explanations could help inspire user trust and loyalty, increase satisfaction, make it quicker and easier for users to find what they want, and persuade them to try or purchase a recommended item

Our Justified Recommendations

[Movie id]	[Movie title]	[The reason is]	[because you rated]
1526	Witness (1985)	Ford, Harrison (I)	21 movies with this feature
1273	Color of Night (1994)	Willis, Bruce	7 movies with this feature
1004	Geronimo: An American Legend (1993)	Hackman, Gene	7 movies with this feature
1442	Scarlet Letter, The (1995)	Oldman, Gary	7 movies with this feature
1044	Paper, The (1994)	Close, Glenn	7 movies with this feature
693	Casino (1995)	De Niro, Robert	6 movies with this feature
274	Sabrina (1995)	Pollack, Sydney	6 movies with this feature
1092	Dear God (1996)	Kinnear, Greg	5 movies with this feature

Explanation Types

- Nearest neighbor explanation
  - Customers who bought item X also bought items Y, Z
  - Item Y is recommended because you rated related item X
- Content based explanation
  - This story deals with topics X, Y which belong to your topic of interest
- Social based explanation
  - People leverage their social network to reach information and make use of trust relationships to filter information
    - Your friend X wrote that blog
    - 50% of your friends liked this item (while only 5% disliked it)

Offline Evaluation

- Based on a pre-collected data set of users choosing or rating items
  - Usually done by recording historical user data, and then hiding some of these interactions in order to compare the user predicted rating with her actual rating
- No interaction with real users, thus allow comparing a wide range of candidate algorithms at a low cost
- Mostly useful for evaluating the prediction power of the system and for system tuning

Online Evaluation

- Evaluate the system by real users that perform real tasks
  - Provides the strongest evidence for the true value of the system to its users
  - The real effect of the recommendation system depends on a variety of user's dependent factors that are changed dynamically
    - The user current intent
    - The user's current context
- Feedback from the users is collected by observing their feedback to the system's recommendation
  - Systems are evaluated according to the acquired vs. non-acquired ratio
- Such a live user experiment may be controlled
  - Randomly assign users to different conditions
    - e.g. test a new version of your system on a test set of users
  - A/B testing: split users to test groups and measure effectiveness of different conditions/algorithms on the groups
- On-line evaluation studies are done on a regular basis by commercial Recommendation Systems

Thanks

Questions?

References: Cold Start Problem

- Freyne J., Jacovi M., Guy I., and Geyer, W. Increasing engagement through early recommender intervention. Proc. RecSys '09, 85-92.
- Schein A.I., Popescul A., Ungar L.H, & Pennock D.M. Methods and metrics for cold-start recommendations. Proc. SIGIR ’02, 253-260.
- Rashid A.M., Albert I., Cosley D., Lam S.K., McNee S.M., & Konstan, J.A. Getting to know you: learning new user preferences in recommender systems. Proc. IUI ’02, 127-134.

References: Evaluation Methods

- The Netflix Prize, Bernett et al. KDD CUP 2007
- Evaluating collaborative filtering recommender systems. Herlocker et al . TOIS 2004
- Personalized social search based on the user's social network. Carmel et al., CIKM 2009
- User Evaluation Framework of recommender Systems, Chen et al. (SRS 2010)
- Performance of recommender algorithms on top-n recommendation tasks, Cremonesi et al. RecSys 2010

References: Fundamental Recommendation Approaches

- Recommender Systems: An Introduction, Jannach et al. 2011.
- Recommender Systems Handbook, Ricci et al. 2010
- Hybrid web recommender systems, : Survey and Experiments. Burke, User Modeling and User-Adapted Interaction. 2002
- Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. Adomavicius et al., IEEE Transactions on Knowledge and Data Engineering. 2005

References: Content Recommendation

- Arguello, J., Elsas, J., Callan, J., & Carbonell J. Document representation and query expansion models for blog recommendation. Proc. ICWSM'08.
- Davidson, J. Liebal, B., Junning L., et al. The YouTube video recommendation system. Proc. RecSys '10, 293-296.
- Groh, G., & Ehmig, C. Recommendations in taste related domains: collaborative filtering vs. social filtering. Proc. GROUP '07, 127-136.
- Golbeck J. Generating predictive movie recommendations from trust in social networks. Proc. 4th Int. Conf. on Trust Management. Pisa, Italy
- Guy, I., Zwerdling, N., Carmel, D., et al. Personalized recommendation of social software items based on social relations. Proc. RecSys '09, 53-60.
- Guy, I., Zwerdling N., Ronen, I, et al. Social media recommendation based on people and tags. Proc SIGIR '10, 194-201.

References: Content Recommendation

- Lerman, K. Social networks and social information filtering on Digg. Proc. ICWSM ’07.
- Liu, J., Dolan, P. & Pederson E.B. Personalized news recommendation based on click behavior. Proc. IUI ’10, 31-40.
- McNee M.S., Riedl, J., & Konstan J.A. 2006. Being accurate is not enough: how accuracy metrics have hurt recommender systems. Proc CHI '06, 1097-1101.
- Sen, S., Vig, J., & Riedl, J.Tagommenders: connecting users to items through tags. Proc. WWW '09, 671-680.
- Sinha, R. & Swearingen, K. Comparing recommendations made by online systems and friends. 2001 DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries.

Thanks!!

Questions?