

Introduction to Machine Learning

Recommendation Systems

Recommendation systems are everywhere

LinkedIn People Recommendations

PEOPLE YOU MAY KNOW



vidhya murali, MTS at Oracle
Connect



Nishan Jain, Research Assistant at University of
Connect



Dhwan Raj, Machine Learning, Natural Language Processing,
Connect

See more »

Facebook People Recommendations

HotJobs Job Recommendations

Marish

Location

Date

Sunnyvale, CA

Feb 9

Mountain View, CA

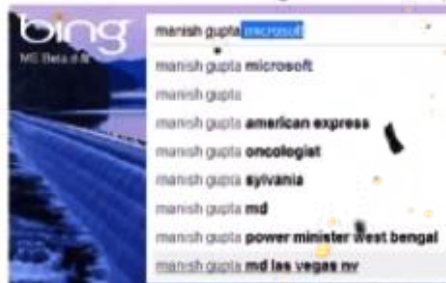
Feb 2

Fremont, CA

Jan 28

more jobs »

Bing Query Recommendations



Netflix Movie Recommendations



Introduction to Machine Learning

Recommendation Systems

Social overload

- Facebook – largest social network site
 - 600,000,000 users per day
 - 35,000,000,000 photos per day
 - 900,000,000 status updates per day
 - 30,000,000 new posts per month
- YouTube – largest video sharing site
 - 2,000,000,000 views per day
 - 1,000,000 videos uploaded per month
- Twitter – largest microblogging site
 - 200,000,000 users per month
 - 65,00,000 tweets per day (750 per second)
 - 8,000,000 followers of most popular user

Introduction to Machine Learning

Recommendation Systems

Social overload

- Information Overload
 - Blogs, microblogs, RSS feeds, etc.
- Interaction Overload
 - Friends, followers, fans, commenters, co-members, voters, likers, taggers, review writers, etc.

Introduction to Machine Learning

Recommendation Systems

Social Recommender Systems

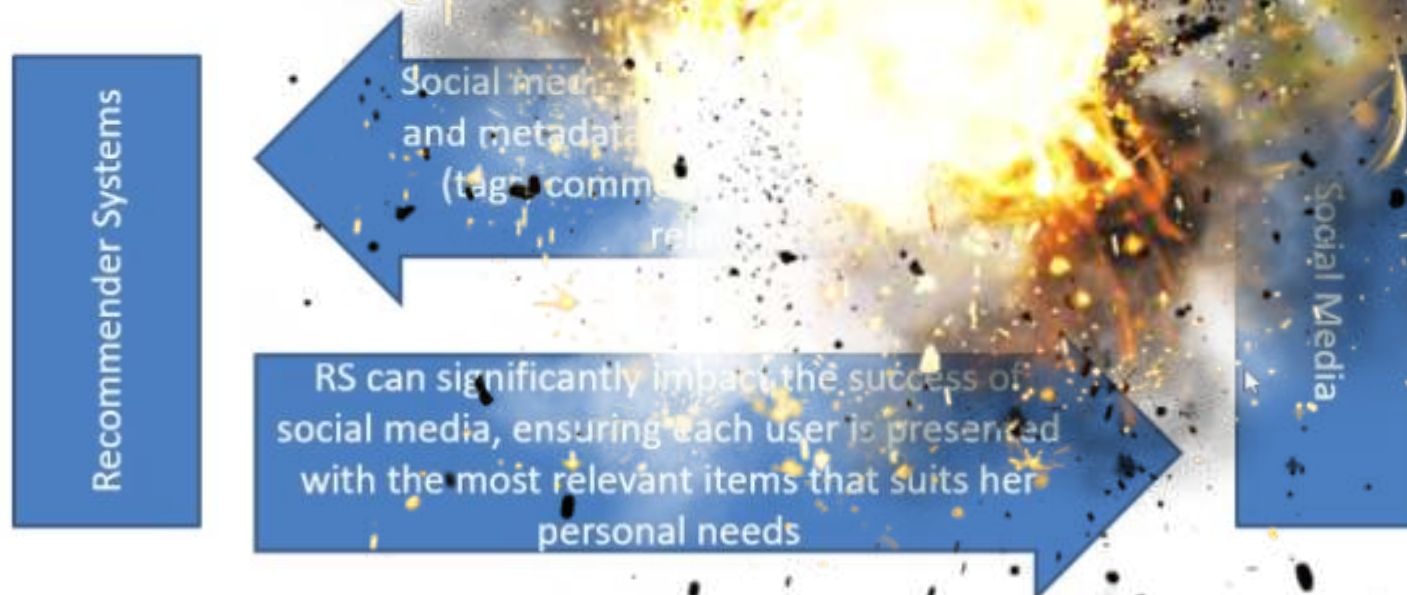
- Recommender systems in the social media domain
- Aim at coping with information overload by presenting selective and relevant content
- Also aim at increasing user interest and engagement
- Often apply personalization techniques

Introduction to Machine Learning

Recommendation Systems

Recommender Systems & Social Media

- Recommender Systems are an augmentation of the social process, in which we rely on advice or suggestions from other people
- Social Media and Recommender Systems refer to each other



Introduction to Machine Learning

Recommendation Systems

Fundamental Recommendation Approaches

- Collaborative filtering
 - Aggregate ratings of users to make recommendation based on similarity
- Demographic Recommendation
 - Categorize users based on (age, gender, income..) and make recommendation based on demographic classes
- Content-based recommendation
 - A user profile is constructed based on the features of the items the user has rated/consumed. This profile is used to identify new interesting items for the user (that match his profile)
- Hybrid methods
 - Combine several approaches together

Introduction to Machine Learning

Recommendation Systems

Collaborative Filtering

Customers Who Bought This Item Also Bought



- In the real world we seek advice from (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

Introduction to Machine Learning

Recommendation Systems

User based Collaborative Filtering Algorithm

- The User x Item Matrix

	Shrim
Alice	Like
Bob	?
Chris	Like
John	Like

- Shall we recommend Superman to John?
- John's taste is similar to both Chris and Alice tastes \Rightarrow Do not recommend Superman to John

User based Collaborative Filtering Algorithm

- Let R be the rating matrix
 - r_{uj} is then the rating given by user u to item j
- I_u be the set of items rated by user u , provided the rating is not zero
- Voting
 - Mean vote for user u : $\bar{r}_u = \frac{1}{|I_u|} \sum_{j \in I_u} r_{uj}$
 - Prediction rating: $p_{uj} = \bar{r}_u + \frac{1}{|I_v|} \sum_{v \in I_v} w(u, v)(r_{vj} - \bar{r}_v)$
 - $w(u, v)$ = similarity between users u and v
 - γ is a normalization constant $\gamma = \frac{1}{\sum_{v=1}^n w(u, v)}$

Similarity Functions

- Cosine based

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

- Pearson based

$$-w(u, v) = \frac{\sum_{i \in I} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{vi} - \bar{r}_v)^2}}$$

Introduction to Machine Learning

Recommendation Systems

Collaborative Filtering: Practical Challenges

- Ratings data is often sparse, and many users with few co-ratings are present
- Fails to incorporate the entire population as a whole
 - Agreement about a product is less important than agreement among many
 - Some algorithms account for this by including weights inversely proportional to the number of ratings
- Calculating a user's performance 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

Introduction to Machine Learning

Recommendation Systems

Item-Based Nearest Neighbor Algorithm

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

	Snow White	Shrek
Alice	5	4
Bob	1	5
Chris	4	4
John	5	4

- Bob dislikes Snow White (which is why we 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