Recommendation System

What is it?

- Recommender systems are a technological proxy for a social process.
- Recommender systems are a way of suggesting like or similar items and ideas to a users specific way of thinking.
- Recommender systems try to automate aspects of a completely different information discovery model where people try to find other people with similar tastes and then ask them to suggest new things.

Personalization

- Recommenders are instances of personalization software.
- Personalization concerns adapting to the individual needs, interests, and preferences of each user.
- Includes:
  - Recommending
  - Filtering
  - Predicting
- From a business perspective, it is viewed as part of Customer Relationship Management (CRM).

Where is it used?

- Massive E-commerce sites use this tool to suggest other items a consumer may want to purchase
- Web personalization

What are the different types of recommendations?

There are basically three important types of recommendation engines:

- Collaborative filtering
  - User Based Collaborative Filtering
  - Item Based Collaborative Filtering
- Content-Based Filtering
- Hybrid Recommendation Systems

Collaborative filtering

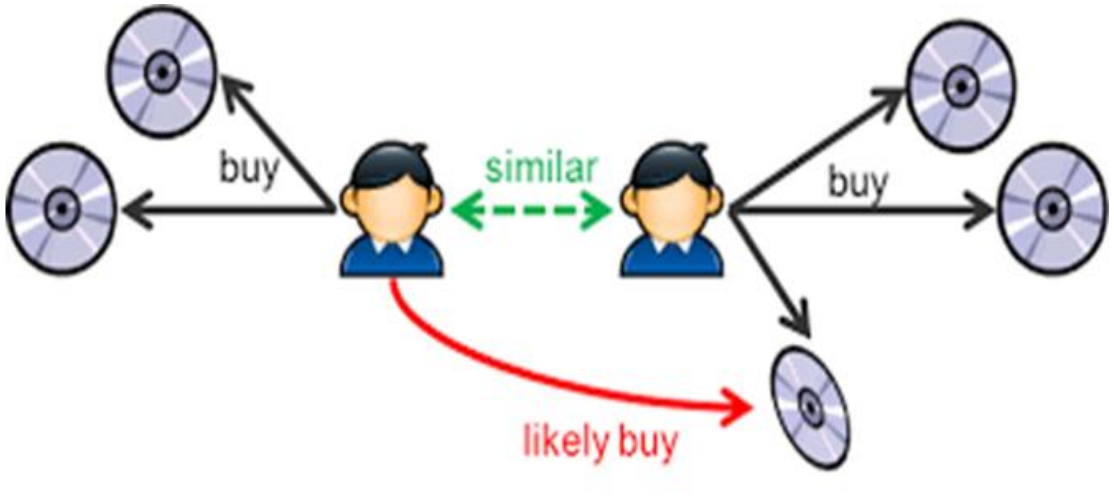
- Maintain a database of many users' ratings of a variety of items.
- For a given user, find other similar users whose ratings strongly correlate with the current user.
- Recommend items rated highly by these similar users, but not rated by the current user.
- Almost all existing commercial recommenders use this approach (e.g. **Amazon**).

User Based Collaborative Filtering:

- Measure how similar each user is to the new one. Popular similarity measures are **correlation** and **cosine**.
- Identify the most similar users. Take account of the users whose similarity is above a defined threshold
- Rate the items purchased by the most similar users. The rating is the average rating among similar users and the approaches are:
    1. Average rating
    2. Weighted average rating, using the similarities as weights
    3. Pick the top-rated items.

How does Collaborative Filtering work?

- Users rate items – user interests recorded. Ratings may be:
  - Explicit, e.g. buying or rating an item
  - Implicit, e.g. browsing time, no. of mouse clicks
- *Cosine Similarity* matching method used to find people with similar interests
- Items that similar users rate highly but that you have *not* rated are recommended to you
- User can then rate recommended items



Drawbacks of Collaborative Filtering

Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity

- **Cold start:** The New user problem, these systems often require a large amount of existing data on a user in order to make accurate recommendations.
- **Scalability:** In many of the environments in which these systems make recommendations, there are millions of users and products. Thus, a large amount of computation power is often necessary to calculate recommendations.
- **Sparsity:** The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

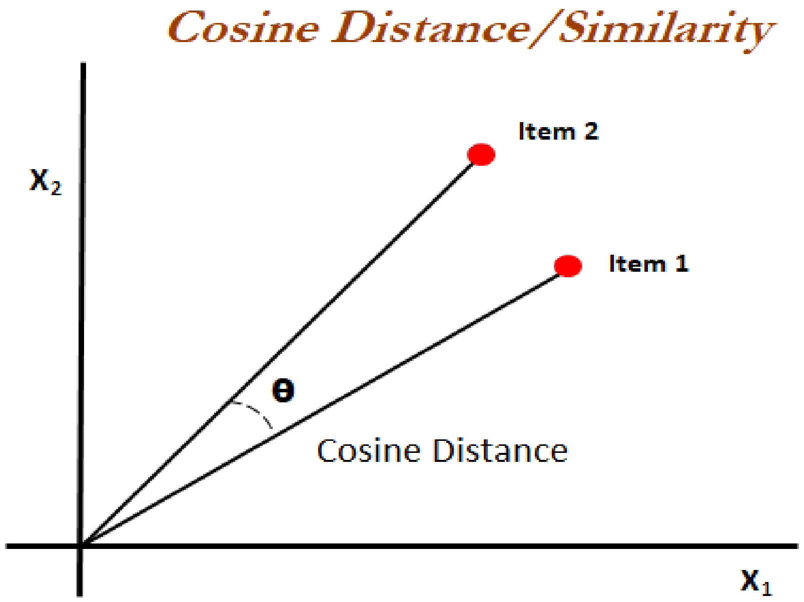
Example of M x N Matrix with M users and N items

- Can construct a vector for each user (where ? implies an item is unrated)
  - E.g. for user1: <2,3,?,...,5>
  - E.g. for user2: <?,4,3,...,?>
- On average, user vectors are sparse, since users rate (or buy) only a few items.

	Item 1	Item 2	Item 3	...	Item n
User 1	2	3	?	...	5
User 2	?	4	3	...	?
User 3	3	2	?	...	3
...	...	...	...	...	...
User m	1	?	5	...	4

Cosine Similarity

- Cosine similarity finds how two vectors are related to each other using measuring cosine angle between these vectors.
- Two vectors with the same orientation have a cosine similarity of 1, two vectors oriented at 90° relative to each other have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude.
- Unit vectors are maximally "similar" if they're parallel and maximally "dissimilar" if they're Orthogonal



Cosine similarity (contd..)

$$sim(u,u') = cos(\theta) = \frac{\mathbf{r}_u \cdot \mathbf{r}_{u'}}{\|\mathbf{r}_u\| \, \|\mathbf{r}_{u'}\|} = \sum_i \frac{r_{ui} r_{u'i}}{\sqrt{\sum_i r_{ui}^2} \sqrt{\sum_i r_{u'i}^2}}$$

Example:

a = (1,4,4)

b = (2,3,4)

CosSimilarity(a,b) =

$$\frac{((1*2)+(4*3)+(4*4))}{sqrt((1^2)+(4^2)+(4^2)) * sqrt((2^2)+(3^2)+(4^2))}$$

= .9697

Prediction Function

- Users rate items on a scale of 1 to 5
- p – prediction, r – rating, *r-bar* – average rating, *sim(i,j)* -cosine similarity

u – user, i – item

Item based: $$pred(u,i) = \frac{\sum_{j \in ratedItems(u)} sim(i,j) \cdot r_{ui}}{\sum_{j \in ratedItems(u)} sim(i,j)}$$

User based: $$pred(u,i) = \bar{r}_u + \frac{\sum_{n \subset neighbors(u)} sim(u,n) \cdot (r_{ni} - \bar{r}_n)}{\sum_{n \subset neighbors(u)} sim(u,n)}$$

Evaluating Prediction Accuracy

Mean Squared Error:

Mean Squared Error (MSE) is used as cost function.

MSE measures the average squared difference between an observation's actual and predicted values.

For linear equation, **y=mx + b,** we can calculate MSE as:

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \tilde{y}_i \right)^2$$

Evaluating Prediction Accuracy (Contd..)

RMSE (Root Mean Square Error) is a quadratic scoring rule that also measures the average magnitude of the error.

- It is the square root of the average of squared differences between prediction and actual observation.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum\left(\hat{Y}_i - Y_i\right)^2}$$

Other Similarity Measures

- **Pearson Correlation**

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- **Jaccard Similarity**

$$jac(x, y) = \frac{|x \cap y|}{|X| + |y| - |x \cap y|}$$

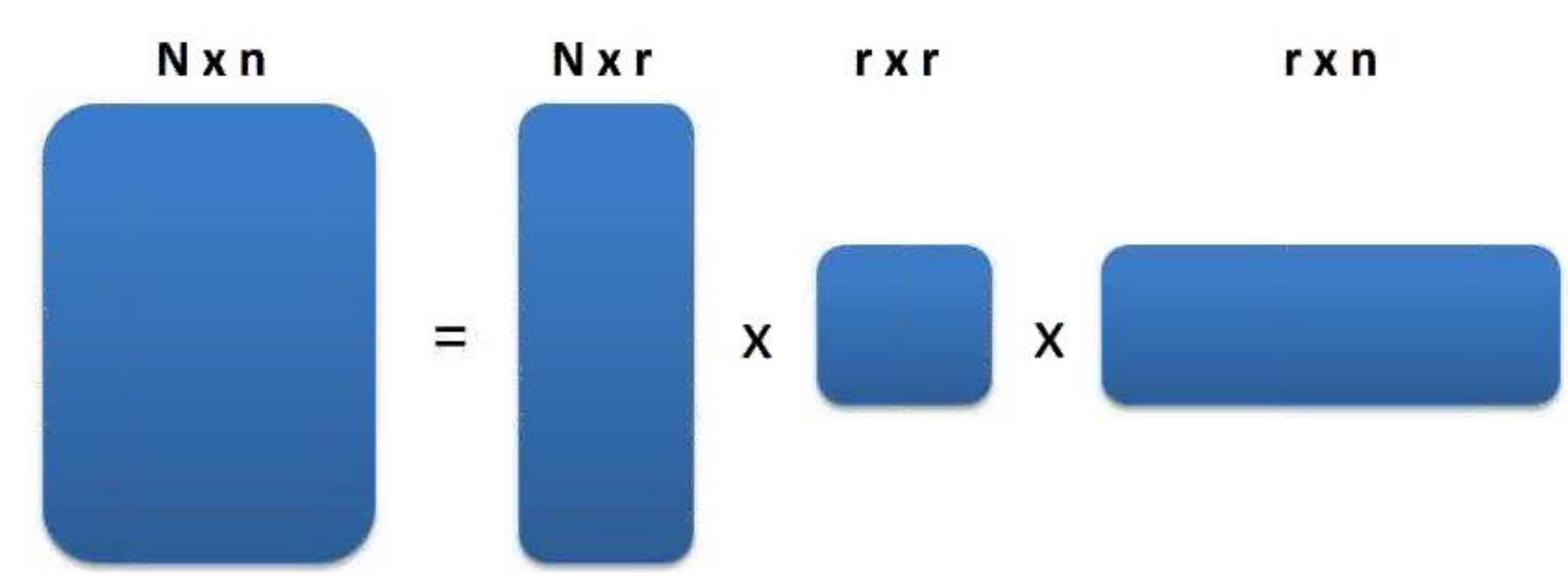
- **Euclidean Distance**

$$Euclidean\ Distance = d = \sqrt{\sum_{i=1}^{N}(Xi - Yi)^2}$$

Challenges of Collaborative Filtering

- *First rater problem* – what happens if an item has not been rated by anyone.
- Privacy problems.
- Can combine CF with CB recommenders
  - Use CB approach to score some unrated items.
  - Then use CF for recommendations.
- *Serendipity* - recommend to me something I do not know already

SVD



Thank you!