



Recommender Systems

Designed to **predict items that may be of interest** to a user.

Primarily used by **merchants** as a method to increase transactions

Can be thought of as related to IR systems, both are intended to identify items of interest/use. Stems from e-commerce sites in the 90s. *GroupLens* established for recommendation of Usenet News articles, followed by *BookLens* and *MovieLens*. *Amazon* and *Netflix* are major drivers of RS development. Netflix also offer explanations for their recommendations.

Base recommendations off the profile of an individual user and their feedback/ratings of previously viewed items (or from feedback and ratings from other users)

Challenge: reliable recommendations require large amount of feedback

RSs seek either to predict the **rating value** (like relevance in IR) assigned to an individual item or to rank the **top-k** items by their predicted rating values. Ranking version is generally easier since only the relative vals of the predicted ratings are needed.

Goals of RS:

- Relevance: recommend relevant items
- Novelty: recommend new items relevant to user which they have not seen before
- Serendipity: recommend unexpected/surprising items which the users find relevant
 - can be recommended based on choices of users with similar interest/purchase profiles to current user. Increases sale diversity but can also agitate a user
- Dersivity: a diverse list of recommended items increases chance that at least one is relevant to user

Ratings can either be (both similar to feedback used in RF)

- Explicit: actively provided by the user as their rating of an item
- Implicit: based on actions of the user - clicking on an item, dwell time, purchase history etc.

Facebook RS not based on selling products, but is **link prediction**, encouraging growth of the social network to encourage increased ad revenue. Recommendations based on structural relationships rather than rating data

3 categories of RS:

- **Content-Based** recommender systems (CBRS)
 - analyse the contents of a set of items rated by the user and use the contents as well as the ratings. Used to infer/build a user profile that can be used to recommend further items that may be of interest to the user
 - new items can be recommended w/o any user rating since the recommendation process is based on item contents. Recommendation in CBRS limited only to features which can be identified in the content of the item and the profile.
 - must determine the *threshold* over which items can be passed to the user (only after profile has been built with sufficient ratings). Over time a more reliable threshold can be determined from larger amounts of feedback
- **Collaborative Filtering** (CF)
 - use ratings from multiple previous users to make their recommendations and take no account of contents. If similar users like it, so might you.
 - does not depend on the content of an item and so is useful for recommendation of *any type* of item.
 - User profile = a set of items and corresponding rating info
 - Approach 1; Memory-based Algorithms
 - Compute similarity between each existing user and the current user, select the closest neighbours and recommend items rated highly by these nearest neighbours
 - simpler approach than other recommender algorithms
 - Approach 2; Model-based Algorithms
 - Use machine learning methods to construct a model to represent the behaviour of the users to predict ratings
 - more robust than mem-based. Don't need as large a number of ratings of items from other users.
- **Knowledge-Based** recommender systems (KBRS)
 - based on user interactions with the RS to specify their interests. User spec is used with domain knowledge to provide recommendations
 - Useful for items purchased infrequently eg cars, home
 - Sufficient ratings not available for these items to make reliable recommendations to a user using other methods
 - Make recommendations based on customer requirements and item descriptions.
 - Knowledge-bases contain data about rules and similarity functions for the matching process. Allow the user to *explicitly* specify what they want
 - *most like IR systems*, they are interactive. But needs are focused on precise user specs and attributes of defined classes of target items

Problems in Recommender Systems

- Sparsity of ratings: in most RSs, each user rates only a small subset of available items, so most remain unrated or sparsely rated. Makes reliable calculation of correlations between users and items of likely interest unreliable
- Cold Start problem: two types
 - *User-side*: relates to the difficulty of making recommendations for new users who have provided little rating info so far
 - *Item-side*: where new items have not been rated often enough to make reliable recommendations. Generally they will not be recommended very often and so will not build up rating information

Hybrid Methods combined CBRS, CF and/or KBRS, provide better recommendations than any one alone. reduces the problems above.

- where insufficient rating info for effective CF, CBRS helps by comparing interests of the user to each item based on content.
- for item side cold start, CBRS can provide recommendations based on new item's content
- KBRS can be used to specify types of items a user is interested in, used as a filter for recommendations by CBRS or CF
- Make the predictions separately and then combine OR
- unify the approaches into a single model

Shilling attacks

Attempt to mislead RS. Manufacturer/author may derive increased sales from over-positive fake reviews on Amazon, or a competitor may submit malicious reviews for competitive advantage

Adversary: person providing fake info to the RS. Coordinated fake input to an RS can disrupt predictions.

Must detect shilling attacks. Need to make sure fakes are removed without compromising real

- Unsupervised: basic idea determine characteristic between real and fake profiles, use these to detect fake accounts
- Supervised: machine learning used to train a classifier to detect fake accounts based on examples of real and fake accounts

Evaluation of RS

Requires a set of items to be recommended, with ratings of the items by a set of users, in which specific items are rated as relevant to a test user.

As in IR, set of items, user and their recommendations should be representative of the environment in which it is planned to operate the RS.

Depending on the task, eval should either:

- Test whether a specific item is correctly recommended to a selected target user
- measure precision and recall of the top-k ranked recommendations