CREATING SUGGESTIONS AND RECOMMENDATIONS

Unit 3

1

Recommender System

- We ask our friends or relatives for suggestions before making day-to-day decisions or buying things.
- To making decisions online to buy products, we read reviews about the products, compare the products' specifications with other similar products and then we make decisions to buy or not.
- Because of the growth of information at an exponential rate, looking for valid information in an online world will be a challenge. **Recommender systems** come to our rescue to provide relevant and required information.
- Some examples of recommender system include suggestions for products on Amazon, friends' suggestions on social applications such as Facebook, Twitter, and LinkedIn, video recommendations on YouTube, news recommendations on Google News, and so on

Recommender System

- **Recommendation engines**, a branch of information retrieval and artificial intelligence, are powerful tools and techniques to analyze huge volumes of data, especially product information and user information, and then provide relevant suggestions based on data mining approaches.
- A recommendation engine problem is to develop a mathematical model or objective function which can predict how much a user will like an item.
- If U = {users}, I = {items} then F = Objective function and measures the usefulness of item I to user U, given by:

  F:U X I = R, where R = {recommended items}.
- For each user u, we want to choose the item i that maximizes the objective function.

Recommender System

- The main goal of recommender systems is to provide relevant suggestions to online users to make better decisions from many alternatives available over the Web.
- A better recommender system is directed more toward personalized recommendations by taking into consideration the available digital footprint of the user, such as user-demographic information, transaction details, interaction logs, and information about a product, such as specifications, feedback from users, comparison with other products, and so on, before making recommendations.
- The most common recommender systems are: ***collaborative filtering recommender system*** and ***content-based recommender system***.

Collaborative Filtering Recommender Systems

- Collaborative filtering recommender systems are basic forms of recommendation engines.
- In this type of recommendation engine, filtering items from a large set of alternatives is done collaboratively by users' preferences.
- The basic assumption in a collaborative filtering recommender system is that if two users shared the same interests as each other in the past, they will also have similar tastes in the future.
- If, for example, user A and user B have similar movie preferences, and user A recently watched Titanic, which user B has not yet seen, then the idea is to recommend this unseen new movie to user B.
- The movie recommendations on Netflix are one good example of this type of recommender system

Collaborative Filtering Recommender Systems

- The idea is very simple: given the ratings of a user towards items, find all the users similar to the active user who had similar ratings in the past and then make predictions regarding all unknown items that the active user has not rated but are being rated in their neighborhood.
- In these types of systems, the main actors are the users, products, and user's rating information such as continuous, interval-based, ordinal, binary, and unary towards the products.
- The two common collaborative filtering recommender systems are: **User-based collaborative filtering** and **Item-based collaborative filtering**. These two are also called **memory-based** or **neighborhood-based collaborative filtering**.

Collaborative Filtering Recommender Systems

- **User-based collaborative filtering:**
  - In this case, the ratings provided by like-minded users of a target user A are used in order to make the recommendations for A.
  - The basic idea is to determine users, who are similar to the target user A, and recommend ratings for the unobserved ratings of A by computing weighted averages of the ratings of this peer group.
  - Therefore, if user A and user B have rated movies in a similar way in the past, then one can use user A's observed ratings on the movie M to predict B's unobserved ratings on this movie.
  - In general, the k most similar users to user B can be used to make rating predictions for user B.
  - Similarity functions (cosine similarity, Pearson correlation coefficient etc.) are computed between the rows of the ratings matrix to discover similar users.

Collaborative Filtering Recommender Systems

❖ **Example 1**: Suppose, the ratings of five users U1, U2, U3, U4, and U5 are indicated for six items denoted by I1, I2, I3, I4, I5, and I6. Each rating is drawn from the range {1 … 7}. Find the predictions of user U3 for items I1 and I6 on the basis of the ratings in the table below. Use Pearson correlation coefficient to find similarity between users and consider top-2 closest users.

$$\text{Sim}(x, y) = \text{Pearson}(x, y) = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2}\sqrt{\sum(y-\bar{y})^2}}$$

User/Item	I1	I2	I3	I4	I5	I6
U1	7	6	7	4	5	4
U2	6	7	?	4	3	4
U3	?	3	3	1	1	?
U4	1	2	2	3	3	4
U5	1	?	1	2	3	3

Collaborative Filtering Recommender Systems

Solution: The first step is to compute the similarity between user U3 and all the other users.

Pearson(U1, U3) =

$$\frac{(6-5.5)\times(3-2)+(7-5.5)\times(3-2)+(4-5.5)\times(1-2)+(5-5.5)\times(1-2)}{\sqrt{1.5^2+1.5^2+(-1.5)^2+(-0.5)^2}\times\sqrt{1^2+1^2+(-1)^2+(-1)^2}} = 0.89442719$$

Similarly, we can calculate,

Pearson(U2, U3) = 0.97072534

Pearson(U4, U3) = -1

Pearson(U5, U3) = -0.8660254

Hence, the top-2 closest users to user U3 are users U1 and U2 according to Pearson correlation coefficient. By using the Pearson-weighted average of the raw ratings of users U1 and U2, the following predictions are obtained for user U3 with respect to her unrated items I1 and I6.

Rank(U3, I1) = (7 * 0.89442719 + 6 * 0.97072534) / (0.89442719 + 0.97072534) = 6.48

Rank(U3, I6) = (4 * 0.89442719 + 4 * 0.97072534) / (0.89442719 + 0.97072534) = 4

Collaborative Filtering Recommender Systems

❖ **<u>Exercise 1</u>:** Suppose, the ratings of five users U1, U2, U3, U4, and U5 are indicated for six items denoted by I1, I2, I3, I4, I5, and I6. Each rating is drawn from the range {1 ... 7}. Find the predictions of user U3 for items I1 and I6 on the basis of the ratings in the table below. Use cosine similarity to find similarity between users and consider top-2 closest users.

$$\mathrm{Sim}(x, y) = \mathrm{Cosine}(x, y) = \frac{\sum xy}{\sqrt{\sum x^2}\sqrt{\sum y^2}}$$

User/Item	I1	I2	I3	I4	I5	I6
U1	7	6	7	4	5	4
U2	6	7	?	4	3	4
U3	?	3	3	1	1	?
U4	1	2	2	3	3	4
U5	1	?	1	2	3	3

Collaborative Filtering Recommender Systems

- **Item-based collaborative filtering:**
  - ❖In order to make the rating predictions for target item B by user A, the first step is to determine a set S of items that are most similar to target item B.
  - ❖The ratings in item set S, which are specified by A, are used to predict whether the user A will like item B.
  - ❖In general, the k most similar items to item B can be used to make rating predictions for item B.
  - ❖Similarity functions are computed between the columns of the ratings matrix to discover similar items.

Collaborative Filtering Recommender Systems

- **Example 2**: Suppose, the ratings of five users U1, U2, U3, U4, and U5 are indicated for six items denoted by I1, I2, I3, I4, I5, and I6. Each rating is drawn from the range {1 ... 7}. Find the predictions of item I1 for user U3 on the basis of the ratings in the table below. Use Pearson correlation coefficient to find similarity between users and consider top-2 closest items.

User/Item	I1	I2	I3	I4	I5	I6
U1	7	6	7	4	5	4
U2	6	7	?	4	3	4
U3	?	3	3	1	1	?
U4	1	2	2	3	3	4
U5	1	?	1	2	3	3

Collaborative Filtering Recommender Systems

Solution: The first step is to compute the similarity between item I1 and all the other items.

Pearson(I1, I2) = 0.94063416

Pearson(I1, I3) = 0.98782916

Pearson(I1, I4) = 0.89714996

Pearson(I1, I5) = 0.67675297

Pearson(I1, I6) = 0.57263713

Hence, the top-2 closest items to item I1 are I2 and I3. By using the Pearson-weighted average of the raw ratings of user U3 for items I2 and I3 is used to predict the rating of item I1 for user U3.

Rank(U3, I1) = (3 * 0.94063416 + 3 * 0.98782916) / (0.94063416 + 0.98782916) = 3

Collaborative Filtering Recommender Systems

- **Advantages:**
  - Easy to implement.
  - Neither the content information of the products nor the users' profile information is required for building recommendations.
  - New items are recommended to users giving a surprise factor to the users.
- **Disadvantages:**
  - This approach is computationally expensive as all the user, product, and rating information is loaded into the memory for similarity calculations.
  - This approach fails for new users where we do not have any information about the users. This problem is called the **cold-start problem**.
  - This approach performs very poorly if we have little data.
  - Since we do not have content information about users or products, we cannot generate recommendations accurately based on rating information only.

Content-based Recommender Systems

- In real life, recommendation should be based on contents of items, such as, genre, actor, director, story, and screenplay in case of movies.
- A recommendation that is targeted at a personalized level and that considers individual preferences and contents of the items for generating recommendations is called a content-based recommender system.
- Content-based recommendation engines solve the cold-start problem that new users face in the collaborative filtering approach. When a new user comes, based on the preferences of the person we can suggest new items that are similar to their tastes.

Content-based Recommender Systems

- Building content-based recommender systems involves three main steps, as follows:
  1. Generating content information for items (item profile generation)
  2. Generating a user profile and preferences with respect to the features of the items (user profile generation)
  3. Generating recommendations and predicting a list of items to the user
- **Item Profile Generation:**
  - ❖In this step, we extract the features that represent the item. Most commonly the content of the items is represented in the vector space model with item names as rows and features as columns.
  - ❖We extract relevant features and their relative importance score associated with the item. These scores may be Boolean or real valued.
  - ❖In the case of movies, item profile might be a Boolean vector where 1 represents presence of feature and 0 represents absence of feature.

Content-based Recommender Systems

- In case of text documents, item profile might be a real valued vector. Item profile is generated using term frequency inverse document frequency (tf-idf). Tf-idf shows the feature relative importance associated with the item.

- **User Profile Generation:**
  - In this step, we build the user profile matching the item feature using the utility matrix representing the connection between users and items.
  - The utility matrix entries could be just 1’s representing user purchases or a similar connection, or they could be arbitrary numbers representing a rating or degree of affection that the user has for the item.

- **Generating Recommendation:**
  - With profile vectors for both users and items, we can estimate the degree to which a user would prefer an item by computing the cosine similarity between user’s and item’s vectors.
  - The greater the cosine angle the more likely the user is to like a movie and hence it can be recommended to the user.

Content-based Recommender Systems

- **Example:** Find the predictions of Movie M4 for user U1 on the basis of the information given in the table below.

Movie/User	U1	U2	U3	U4	U5	U6	Genre
M1	1	1	?	1	1	?	Romance
M2	1	1	1	1	1	?	Thriller
M3	1	1	1	1	1	1	Action
M4	?	1	1	1	1	1	Romance
M5	1	1	1	1	1	?	Crime
M6	1	1	1	1	1	1	Crime

Content-based Recommender Systems

Item Profile Generation:

Movie/ Genre	Romance	Thriller	Action	Crime
M1	1	0	0	0
M2	0	1	0	0
M3	0	0	1	0
M4	1	0	0	0
M5	0	0	0	1
M6	0	0	0	1

Content-based Recommender Systems

User Profile Generation:

Movie/ Genre	Romance	Thriller	Action	Crime
U1	0.2	0.2	0.2	0.4
U2	0.33	0.17	0.17	0.33
U3	0.2	0.2	0.2	0.4
U4	0.33	0.17	0.17	0.33
U5	0.33	0.17	0.17	0.33
U6	0.33	0	0.33	0.33

$\text{Sim}(M4, U1) = \text{Cosine}(M4, U1) = 0.38$

Content-based Recommender Systems

- **Advantages:**
  - Content-based recommender systems target at an individual level
  - Recommendations are generated using the user preferences alone rather than the user community as with collaborative filtering
  - These approaches can be employed in real time as the recommendation model doesn't need to load all the data for processing or generating recommendations
  - Accuracy is high compared to collaborative approaches as they deal with the content of the products instead of rating information alone
  - The cold-start problem can be easily handled
- **Disadvantages:**
  - As the system is more personalized the generated recommendations will become narrowed down to only user preferences when more user information comes into the system
  - As a result, no new products that are not related to the user preferences will be shown to the user
  - User will not be able to look at what is happening around them or what's trending

Mining Frequent Patterns

- Apriori
- FP growth