International Journal of Future Generation Communication and Networking
Vol. 13, No. 3, (2020), pp. 3209–3216

Recommender Systems: Basics, Methods and Applications

***Vandana Mohan Patil[1], J. B. Patil[2]**

[1]Research Scholar, Department of Computer Engineering, R. C. Patel Institute of Technology, Shirpur (MS), India.
[2]Professor, Department of Computer Engineering, R. C. Patel Institute of Technology, Shirpur (MS), India.
E-mail
1shraddhasam@rediffmail.com, 2jbpatil@hotmail.com

Abstract

Recommender systems efficiently deal with the information overload problems on World Wide Web. The recommender systems are software tools and techniques that recommend only the interesting and relevant items to the user by filtering the huge information on Internet. Thus, resulting in improved users' satisfaction during navigation. The conventional, pure Web usage data based recommender systems are used efficiently. However, research from last few decades is focusing on improving the recommender systems' performance by integrating domain knowledge along with the usage data. This paper discusses the recommender systems in detail. It explains the basic concepts of recommender systems and its types such as Collaborative Filtering (CF) based, Content based (CB), Hybrid, and Knowledge Based (KB) etc. Further CF based approach is discussed in detail. The paper also presents the major research problems in recommender systems, applications of recommender systems and the performance measures used for evaluation of recommender systems.

Keywords: *Recommender systems, CF-based Recommender systems, CB-based Recommender systems, Item-based collaborative filtering.*

1. Introduction

Huge amount of information on World Wide Web (WWW) results in information overload problems for the users throughout the navigation. This may cause the users' disappointment. Retaining users' interest is a significant aspect. Recommender systems are efficient means for dealing with such problems through generating personalized and limited to users [1].

The recommender systems are software tools and techniques that filter the huge information on Internet and suggest only the relevant and interesting items to the user. This results in improved users' satisfaction during navigation. The conventional, pure Web usage data based recommender systems are used efficiently. However, research from last few decades is focusing on improving the recommender systems' performance by integrating domain knowledge along with the usage data [2].

Recommender Systems recommend items which will be useful to the user. The recommendations are helpful to the users in variety of decision-making processes [1]. The recommended items may be the grocery or food items to buy, music to be listened, or books/news/articles/research paper to be read etc. Item refers to anything that the system wants to recommend to the users.

The recommendations can be personalized and non-personalized. But the nonpersonalized recommendations are useful in limited situations hence they are not typically addressed by researchers. Personalized recommendations are very useful for the users as well as service providers by several means; such as 1) Increase the sell, 2) Sell wide variety of items, 3) Improve the users' satisfaction, 4)Improve user reliability, 5)Understand the users' demands/interests. Hence, personalized recommender systems are focused by researchers. In their simplest form, personalized recommendations are offered as ranked lists of items [1].

Recommender systems use three different types of objects: users, transactions and items. Items are objects to be suggested or recommended. A transaction is an interaction between the recommender system and a Web user. Transactions collect vital information generated during the users' navigation like log data. A transaction log may contain a reference to the item selected by the user, a description of the context, explicit feedback by the user, rating given to the item, tags given by the user for the item, reviews of items by user etc. Among all of these, ratings data are the most widely used type of transaction data. These ratings may be ordinal ratings, numerical ratings, or binary ratings (e.g. agree or disagree, interested or not interested, dislike or like, good or bad) etc.

According to the core technology used in the recommender system, different features and properties of the items can be used. Items can be represented in a minimalist way as a single id code, or in a richer form, as a set of attributes, or even as a concept in an ontological representation of the domain [1, 3].

2. Types of Recommender Systems

Core function of a recommender system is to identify or predict the useful items for the user [4]. Recommender systems are categorized based upon the prediction technique used to generate recommendations. There are six different recommendation methods:

2.1 Content-based recommender system

This approach is based on the principle of similar items that the user preferred during the past navigations [5]. The similarity of items is calculated based on the attributes of the items under consideration. The content-based recommender system stores interests and preferences of a user in a user profile. Then, the content-based recommender system compares the characteristics of a user profile with the attributes of an item, in order to recommend new interesting items to the user [5].

2.2 Collaborative filtering based recommender system

This method of recommendation identifies the users who have similar interests as of the active user and suggest the items liked by these similar users in the past [6]. The similarity in interests of two users is computed using the past rating similarity of the users. Collaborative filtering is the most popular and widely implemented technique in recommender system. In this technique, the users' past behavior is used to build a model that is used to predict items or ratings on items that user may be interested in [6].

2.3 Demographic recommender system

This approach uses the users' demographic profile and ratings data to generate recommendations [7]. Demographic means group of population based on some characteristics e.g. young population, male or female population etc. The items to be recommended depend on the type of demographic group.

2.4 Knowledge-based recommender system

Knowledge-based recommender systems use the domain knowledge to generate recommendations [3]. There are various ways to represent the domain knowledge such as ontology, resource description framework (RDF) etc.

2.5 Community-based recommender system

The community-based recommender system [1] uses the preferences of the same community users such as friends, relatives or colleagues. The assumption is that people generally have a tendency to rely on recommendations from their friends.

2.6 Hybrid recommender systems

These hybrid recommender system refers to the combination of any of the above mentioned techniques [8]. It aims at reducing the disadvantages of one technique by combining it with other.

There are seven different techniques for hybridization:

- Weighted – This technique combines the numerical score of different recommendation components.
- Switching – It chooses one among the different recommendation components to be applied next.
- Mixed – A recommendation is generated by combining the results of different recommenders altogether.
- Feature Combination - Features derived from different knowledge sources are combined together and given to a single recommendation algorithm.
- Feature Augmentation - A feature or set of features is computed by one recommendation technique is given as input or part of input to the next recommendation technique.
- Cascade - Recommenders are assigned strict priority for the selection basis.Meta level - One recommendation technique is applied and produces a model, which is then applied as input to the next technique.

3. Collaborative Filtering Based Recommender Systems

Collaborative filtering is an intelligent approach for implementing personalization [1]. The basic principle in collaborative filtering method is based on identifying similar users or similar items.

Accordingly, there are two types of collaborative filtering [6]:
User Based CF-based Recommender System (UB-CF)

 Item Based CF-based Recommender Systems (IB-CF)

3.1 User Based CF-based Recommender System (UB-CF)

UB-CF is a memory-based technique that uses entire database to recommend items by identifying similar users to the active user. UB-CF fills a User-Item Matrix, predicting the ratings on items the active user has not seen, based on the other similar users. This technique performs two tasks:

1. Find the similar users to the active user a. A similarity function is used to compute the similarity between each pair of users.

2. Predict the rating that the active user a will give to the items the similar users have given but the active user a has not rated. Identify the item j with the best predicted rating.

3.2 Item Based CF-based Recommender System (IB-CF)

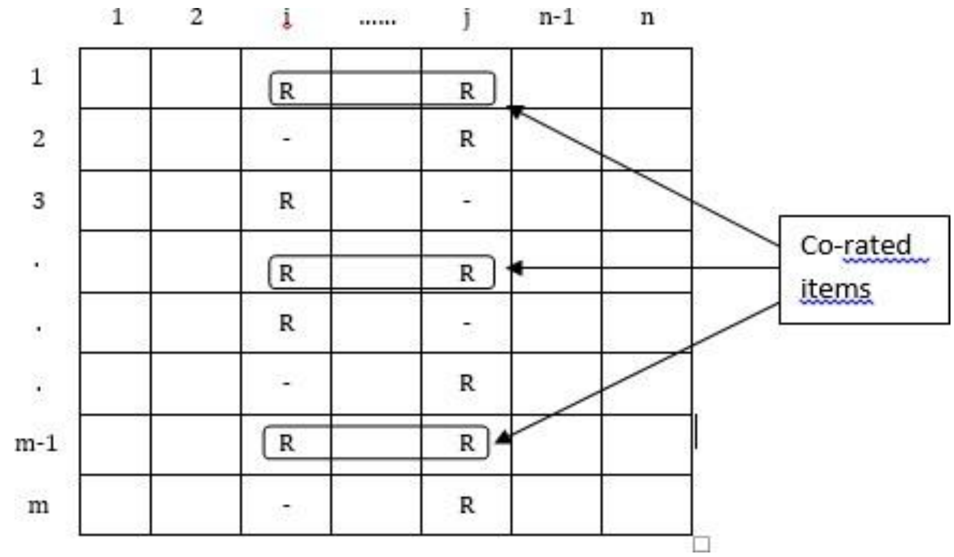
Item-Based Collaborative Filtering (IB-CF) is model-based technique that focus on similar items rather than similar users. IB-CF focus on items in which the active user seems to be interested in. IB-CF fills an Item-Item matrix, and recommends based on similar items.

IB-CF performs two sub tasks:

1. Calculate similarity among the items. Number of methods is there to calculate similarity among the items using the historical rating data such as Jaccard distance, Correlation-Based Similarity, Cosine-Based Similarity, Adjusted Cosine Similarity, etc.
2. Calculation of Prediction: The prediction is done using either Weighted Sum method or Regression based method.

4. Item-Based Collaborative Filtering

This model based technique identifies [8] the similar items by computing the similarity measure between the two items and then performs prediction. The model based techniques involve building a model based on the dataset of ratings, to make recommendations without having to use the complete dataset every time. Hence this technique offers the benefits of speed and scalability.



	1	2	i	......	j	n-1	n
1			R		R		
2			-		R		
3			R		-		
.			R		R		
.			R		-		
.			-		R		
m-1			R		R		
m			-		R		

Figure 1. Co-Rated Items Used In IB-CF Recommender System

i. Identify similar items

The similarity values between items are measured by observing all the users who have rated both the items. The similar items are identified by calculating the similarity metrics between the two items under consideration. Similarity measure play a crucial role in identifying similar items in item based collaborative filtering. Different similarity measures are used conventionally for similarity measure computation [10] e.g. Pearson Correlation, Euclidian Distance, Cosine Similarity, Adjusted Cosine Similarity, Jaccard Similarity etc. These conventional methods are purely based on past rating data.

- Euclidean distance

Euclidean distance is the simplest and most common example of a distance measure [11] and is given by equation (1):

$$d(x, y) = \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2} \qquad (1)$$

where –

- n is the number of dimensions (attributes)
- x_k and y_k are the k^{th} attributes (components) of data objects x and y, respectively.

- Pearson Correlation

The Pearson correlation measures the similarity between items by their *correlation* which measures the linear relationship between objects [11]. It is calculated by the equation (2) as stated in [11]:

$$Sim_{i,j} = \frac{\sum_{u\in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u\in U}(R_{u,i} - \bar{R}_i)^2} * \sqrt{\sum_{u\in U}(R_{u,j} - \bar{R}_j)^2}} \qquad (2)$$

where-

- U is set of users who rated both items i and j.
- $Sim_{i,j}$ is similarity score between items i and j. - $R_{u,i}$ is Rate of user u to item i.
- $\bar{R}_i$ is Mean Rating of item i.
- $\bar{R}_j$ is Mean Rating of item j.

- Cosine Similarity

The similarity measure of item i and item j is a function of rating to item i and j by all the users who rated both the items and mean rating of the user. . It is calculated by the equation (3) as stated in [6],

$$Sim_{i,j} = \frac{\sum_{u\in U}(R_{u,i})(R_{u,j})}{\sqrt{\sum_{u\in U}(R_{u,}{}^{i})} * \sqrt{\sum_{u\in U}(R_{u,}{}^{j})}} \qquad (3)$$

where-

- U is set of users who rated both items i and j - $Sim_{i,j}$ is similarity score between items i and j - $R_{u,i}$ is Rate of user u to item i.
- $R_{u,j}$ is Rate of user u to item j.

- Adjusted Cosine Similarity

The similarity measure of item i and item j is a function of rating to item i and j by all the users who rated both the items and mean rating of the user. It is calculated by the equation (4) as stated in [10],

$$Sim_{i,j} = \frac{\sum_{u\in U}(R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{} \quad \sqrt{}} \qquad (4)$$

$$\sum_{u \in U} (R_{u,i} - R_u)^2 * \sum_{u \in U} (R_{u,j} - R_u)^2$$

where-

- U is set of users who rated both items i and j.
- $Sim_{i,j}$ is similarity score between items i and j. - $R_{u,i}$ is Rate of user u to item i.
- $\overline{R_u}$ is Mean Rating of user u.

ii. Rate Prediction

After the similarity measure calculation, the rates are predicted for unseen items by the active user using the similarity measures. The weighted sum method is widely used for rate prediction that is based on the similarity measure between two items. Similarity metric or measure between item i and j can be computed using different techniques such as Pearson Correlation, Euclidian Distance, Cosine Similarity, Adjusted Cosine Similarity, Jaccard Similarity etc. The weighted sum for rate prediction is calculated by the equation (5) as stated in [10]:

$$Pr_{u,i} = \frac{\sum_{j \in similarTo(i)} S_{i,}{}^{j} * R_{u,j}}{\sum_{j \in similarTo(i)} S_{i,j}} \tag{5}$$

where,

- $Pr_{u,i}$ - Predicted rate of item i by user u.
- $similarTo(i)$- Set of items similar to item i.
- $S_{i,j}$ - Similarity measure between item i and j where $j \in similarTo(i)$.
- $R_{u,j}$ - Rate given by user u to item j where $j \in similarTo(i)$.

5. Open Research Problems

Collaborative based systems require a huge volume of information about a user to generate accurate recommendations. This is called as cold-start problem; whereas content based systems needs very less information to start, but suffers from limited scope i.e. generating only those recommendations that are similar to original seed. Further, pure usage-based approach to collaborative filtering based recommendation suffers from "new-item-problem" that can be solved by integrating content characteristics with user ratings. Techniques for computing similarity between objects are an open research issue [11]. Domain ontologies can be integrated at any stage or all stages of Web usage mining process with different challenges; in preprocessing stage, the challenges are in automatic methods for the extraction and learning of the ontologies and in the mapping of users' activities in the click stream level to more abstract concepts and classes. At the data mining stage, the primary goal is to develop new approaches that take into account complex semantic relationship such as those present in relational databases with multiple relations. At personalization stage, challenge is in developing techniques that can successfully and efficiently measure semantic similarities among objects.

6. Applications

The most common recommender systems applications[1,7] are in various domains such as entertainment, e-commerce, services, education for recommending movies, music, personalized newspapers, documents, research papers, Web pages, e-learning applications, e-mail filters, products to buy such as books, cameras, PCs, travel services, experts for consultation, houses to rent, or matchmaking services etc.

7. Performance Measures

Performance evaluation is important in accessing the effectiveness of recommender systems. The goal of an evaluation metric is to measure the performance of a recommender system [12]. Different metrics are used to evaluate the performance of recommender systems. Two main and popular evaluation metrics are execution time and accuracy. As a recommender system consists of two main tasks: 1) Rate Prediction, and 2) Recommendation. Accuracy metrics used for both tasks are different.

To evaluate prediction accuracy [12,13], different error metrics are used such as Mean Absolute Error (MAE), Mean Square Error(MSE), Root Mean Square Error (RMSE) etc. Mean Absolute Error is one of such metrics. Mean Absolute Error (MAE) is a statistical accuracy metric. It is a measure of deviance between predicted rating *Pi* and actual rating *Ri*. Lesser the value of MAE, more is the accuracy.

To evaluate recommendation accuracy, many evaluation metrics are available. However, Recall@k, Precision@k, and F-measure@k are quite suitable [13, 14] for the rating data. Recall@k represents the proportion of relevant items found in the top-N recommendations. Precision@k represents the proportion of recommended items in the top-N set that are relevant. F-measure is a collective measure of accuracy [14]. It is a weighted mean of the recall and precision to represent the overall utility of the generated recommendation list. The value 1 represents the best value for F-measure and the value 0 represents the worst value. In recommendations domain, the precision value of 1.0 indicates that each item in the generated recommendation list is good, although it does not say anything about whether all good recommendations were suggested. A recall value of 1.0 designates that all good items are recommended in the list; however it does not say anything about how many bad recommendations are there in the list. Hence, F-measure is used to represent the collective accuracy of the system.

8. Conclusion

Recommender systems are very efficient in providing the information of interest to the Web users. There are various methods for recommendation. The collaborative filtering is a widely studied by the research community. The main task in CF-based recommendation is finding the similarities among objects. Objects may be users or items. Based on this, the CF-based recommender systems are further classified as user-based and item-based. The user-based approach is based on the principle of finding similar users; while item-based approach identifies similar items. The item-based approach is model based technique that builds a model based on the dataset of ratings, to make recommendations without having to use the complete dataset every time. Hence, this technique offers the benefits of speed and scalability.

References

[1] Francesco Ricci, Lior Rokach, and Bracha Shapira, "Introduction to recommender systems handbook", in Recommender Systems Handbook, Springer, 2011, pp. 1-35.W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 1–38. [2] E. Magdalini, and M. Vazirgiannis, "Web mining for web personalization", ACM Transactions on Internet Technology (TOIT), vol. 3, no. 1, (2003), pp. 1-27.

[3] Robin Burke, "Knowledge-based recommender systems", Encyclopedia of library and information Systems, 69(Supplement 32):1, 2000, pp. 75-186.

[4] Mingang Chen and Pan Liu, "Performance evaluation of recommender systems", International Journal of Performabilty Engineering, 13(8), 2017, pp. 1246-1256.

[5] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro, "Content-based recommender systems: State of the art and trends", in Recommender Systems Handbook, Springer, 2011, pp. 73-105. [6] Shuyu Luo, "Intro to Recommender System: Collaborative Filtering",[Online]. Available: https://towardsdatascience.com/intro-to-recommender-system-collaborative-filtering-64a238194a26, Dec. 2018.

[7] P. N. Vijaya Kumar , Dr. V. Raghunatha Reddy, "A Survey on Recommender Systems (RSS) and Its Applications", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 8, August 2014, pp. 5254-5260.

[8] Daniel Lew, Ben Sowell, Leah E. Steinberg, Amrit S. Tuladhar, "Recommender Systems", [Online]. Available: https://www.cs.carleton.edu/cs_comps/0607/recommend/recommender/modelbased.html, 2007. [9] Burke, R., "Hybrid web recommender systems", *The Adaptive Web*, 2007, pp. 377–408.

[10] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, "Item-based collaborative filtering recommendation algorithms", in Proceedings of the 10th International Conference on World Wide Web, WWW '01, New York, NY, USA, ACM, 2001, pp. 285-295.

[11] Xavier Amatriain, Alejandro Jaimes, Nuria Oliver, and Josep M. Pujol, "Data Mining Methods for Recommender Systems", in Recommender Systems Handbook, Springer, 2011, pp. 1-35.W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 39-72.

[12] Honghua Dai, "Integrating semantic knowledge with web usage mining for personalization",in Web mining:Applications and techniques, IGI Global, 2005, pp. 276-306.

[13] F.O. Isinkaye, Y.O. Folajimi, and B.A. Ojokoh, "Recommendation systems: Principles, methods and evaluation", Egyptian Informatics Journal, 16(3), 2015, pp. 261 273.

[14] Maher Malaeb, "Recall and Precision at k for Recommender Systems", https://medium.com/@m n malaeb/ recall-and-precision-at-k-for-recommender-systems-618483226c54/, 2017. [Online; accessed 253- 2019].

[15] Marcel Caraciolo," Evaluating Recommender Systems - Explaining FScore, Recall and Precision using Real Data Set from Apontador", http://aimotion.blogspot.com/2011/05/evaluating-recommender-systems. html/, 2011. [Online; accessed 2-4-2019].