A Study of Effectiveness of Collaborative Filter in Web Mining

DR. VISHAL H. BHEMWALA[2]
DR. JAYESH N. MODI[1]
Assistant Professor, Deptartment Of Computer Science, Hemchandracharya North Gujarat University, Gujarat, India.
Deptartment Of Computer Science, Hemchandracharya North Gujarat University, Gujarat, India.

Abstract.
Everytime user surprised with visiting websites. How web comes to know the interest of any individual interest. How would it is possible for the web to remember all user likes and dislikes, buying patterns, interest in purchasing stuff. It's all because of techniques known as collaborative filters. Whenever user first visit to the web, ever website creating profile about the user. They maintained records in matrix form so, retrival can be very easily done. Collaborative filtering (CF) is a technique used by recommender systems. collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. Applications of collaborative filtering typically involve very huge data records as the input. Collaborative filtering methods have been applied to many different kinds of data including: sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data, such as financial service institutions that integrate many financial sources; or in electronic commerce and web applications where the focus is on user data, etc. Though the software used for collaborative filter, there are pre requirements for it. user must have to take active participation, it is quite efficient way to capture input from user and algortihm that enables reads and analyze the input data. actually it is a collective efforts of human interaction with software to derive knowledge from the stored data record set.

Keywords: *Web mining, Web Content Mining, Data Mining, Data Cleansing*

Introduction

The growth of the Internet has made it much more difficult to effectively extract useful information from all the available online information. The overwhelming amount of data necessitates mechanisms for efficient information filtering. Collaborative filtering is one of the techniques used for dealing with this problem.

The motivation for collaborative filtering comes from the idea that people often poeple gets best recomendation from some other persons likes and dislikes. Human always learns from the other person experiences. Collaborative filtering encompasses techniques for matching people with similar interests and making recommendations on this basis.

Collaborative filtering algorithms often require (1) users' active participation, (2) an easy way to represent users' interests, and (3) algorithms that are able to match people with similar interests.
Typically, the workflow of a collaborative filtering system is:

1. A user expresses his or her preferences by rating items (e.g. books, movies or CDs) of the system. These ratings can be viewed as an approximate representation of the user's interest in the corresponding domain.

2. The system matches this user's ratings against other users' and finds the people with most "similar" tastes.
3. With similar users, the system recommends items that the similar users have rated highly but not yet being rated by this user (presumably the absence of rating is often considered as the unfamiliarity of an item)

A key problem of collaborative filtering is how to combine and weight the preferences of user neighbors. Sometimes, users can immediately rate the recommended items. As a result, the system gains an increasingly accurate representation of user preferences over time. Collaborative Filtering is the most common technique used when it comes to building intelligent recommender systems that can learn to give better recommendations as more information about users is collected.

Most websites like Amazon, YouTube, and Netflix use collaborative filtering as a part of their sophisticated recommendation systems. You can use this technique to build recommenders that give suggestions to a user on the basis of the likes and dislikes of similar users. These days whether you look at a video on YouTube, a movie on Netflix or a product on Amazon, you're going to get recommendations for more things to view, like or buy. You can thank the advent of machine learning algorithms and recommender systems for this development. Recommender systems are far-reaching in scope, so we're going to zero in on an important approach called collaborative filtering, which filters information by using the interactions and data collected by the system from other users. It's based on the idea that people who agreed in their evaluation of certain items are likely to agree again in the future.

2. Methodology

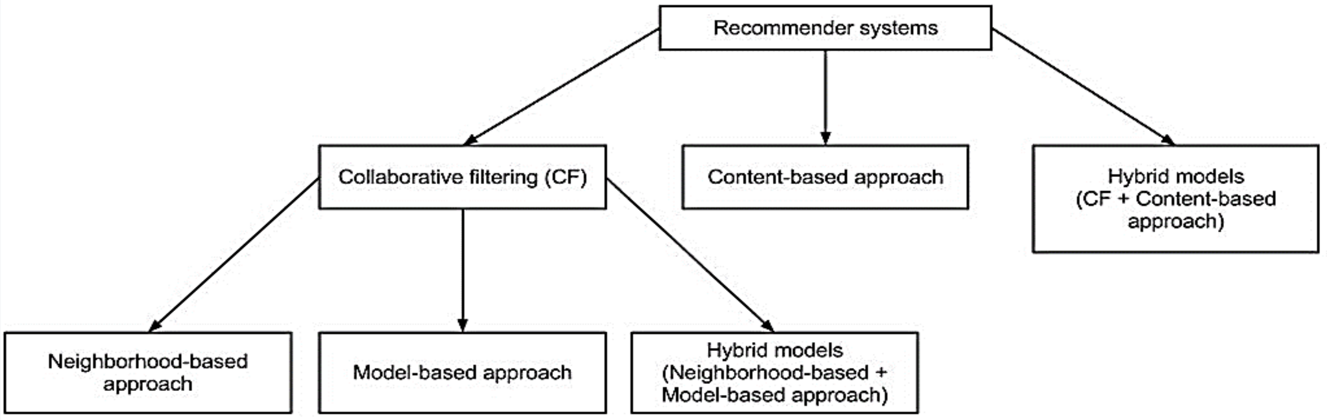
Recommender systems
Collaborative filtering (CF)
Content-based approach
Hybrid models (CF + Content-based approach)
Neighborhood-based approach
Model-based approach
Hybrid models (Neighborhood-based + Model-based approach)

Fig. 1 Collaborative Filtering in Recommender Systems

Collaborative filtering systems have many forms, but many common systems can be reduced to two steps:

1. Look for users who share the same rating patterns with the active user (the user whom the prediction is for).
2. Use the ratings from those like-minded users found in step 1 to calculate a prediction for the active user

This falls under the category of user-based collaborative filtering. A specific application of this is the user-based Nearest Neighbor algorithm. Alternatively, item-based collaborative filtering (users who bought x also bought y), proceeds in an item-centric manner:

1. Build an item-item matrix determining relationships between pairs of items
2. Infer the tastes of the current user by examining the matrix and matching that user's data

Another form of collaborative filtering can be based on implicit observations of normal user behavior (as opposed to the artificial behavior imposed by a rating task). These systems observe what a user has done together with what all users have done (what music they have listened to, what items they have bought) and use that data to predict the user's behavior in the future, or to predict how a user might like to behave given the chance. These predictions then have to be filtered through business logic to determine how they might affect the actions of a business system. For example, it is not useful to offer to sell somebody a particular album of music if they already have demonstrated that they own that music.

Relying on a scoring or rating system which is averaged across all users ignores specific demands of a user, and is particularly poor in tasks where there is large variation in interest (as in the recommendation of music). However, there are other methods to combat information explosion, such as web search and data clustering.

3.Types of Collaborative Filter

There are Four Types of Collabrative filter as below.

1. Memory Based
2. Model Based
3. Hybrid
4. Deep Learning

3.1 Memory Based

The memory-based approach uses user rating data to compute the similarity between users or items. Typical examples of this approach are neighborhood-based CF and item-based/user-based top-N recommendations. The user based top-N recommendation algorithm uses a similarity-based vector model to identify the k most similar users to an active user. After the k most similar users are found, their corresponding user-item matrices are aggregated to identify the set of items to be recommended. A popular method to find the similar users is the Locality-sensitive hashing, which implements the nearest neighbor mechanism in linear time. The advantages with this approach include: the explainability of the results, which is an important aspect of recommendation systems; easy creation and use; easy facilitation of new data; content-independence of the items being recommended; good scaling with co-rated items.

There are also several disadvantages with this approach. Its performance decreases when data gets sparse, which occurs frequently with web-related items. This hinders the scalability of this approach and creates problems with large datasets. Although it can efficiently handle new users because it relies on a data structure, adding new items becomes more complicated since that representation usually relies on a specific vector space. Adding new items requires inclusion of the new item and the re-insertion of all the elements in the structure.

3.2 Model Based

In this approach, models are developed using different data mining, machine learning algorithms to predict users' rating of unrated items. There are many model-based CF algorithms. Bayesian networks, clustering models, latent semantic models such as singular value decomposition, probabilistic latent semantic analysis, multiple multiplicative factor, latent Dirichlet allocation and Markov decision process based models.

Through this approach, dimensionality reduction methods are mostly being used as complementary technique to improve robustness and accuracy of memory-based approach. In this sense, methods like singular value decomposition, principal component analysis, known as latent factor models, compress user-item matrix into a low-dimensional representation in terms of latent factors. One advantage of using this approach is that instead of having a high dimensional matrix containing abundant number of missing values we will be dealing with a much smaller matrix in lower-dimensional space. A reduced presentation could be utilized for either user-based or item-based neighborhood algorithms that are presented in the previous section. There are several advantages with this paradigm. It handles the sparsity of the original matrix better than memory based ones. Also comparing similarity on the resulting matrix is much more scalable especially in dealing with large sparse datasets.

3.3 Hybrid

A number of applications combine the memory-based and the model-based CF algorithms. These overcome the limitations of native CF approaches and improve prediction performance. Importantly, they overcome the CF problems such as sparsity and loss of information. However, they have increased complexity and are expensive to implement. Usually most commercial recommender systems are hybrid, for example, the Google news recommender system.

3.4 Deep Learning

In recent years a number of neural and deep-learning techniques have been proposed. Some generalize traditional Matrix factorization algorithms via a non-linear neural architecture , or leverage new model types like Variational Autoencoders. While deep learning has been applied to many different scenarios: context-aware, sequence-aware, social tagging etc. its real effectiveness when used in a simple collaborative recommendation scenario has been put into question. A systematic analysis of publications applying deep learning or neural methods to the top-k recommendation problem. The article also highlights a number of potential problems in today's research scholarship and calls for improved scientific practices in that area. Similar issues have been spotted also in sequence-aware recommender systems.

4 Application on Social Web

Unlike the traditional model of mainstream media, in which there are few editors who set guidelines, collaboratively filtered social media can have a very large number of editors, and content improves as the number of participants increases. Services like Reddit, YouTube, and Last.fm are typical examples of collaborative filtering based on media. One scenario of collaborative filtering application is to recommend interesting or popular information as judged by the community. As a typical example, stories appear in the front page of Reddit as they are "voted up" (rated positively) by the community. As the community becomes larger and more diverse, the promoted stories can better reflect the average interest of the community members.

Another aspect of collaborative filtering systems is the ability to generate more personalized recommendations by analyzing information from the past activity of a specific user, or the history of other users deemed to be of similar taste to a given user. These resources are used as user profiling and helps the site recommend content on a user-by-user basis. The more a given user makes use of the system, the better the recommendations become, as the system gains data to improve its model of that user.

Problem: A collaborative filtering system does not necessarily succeed in automatically matching content to one's preferences. Unless the platform achieves unusually good diversity and independence of opinions, one point of view will always dominate another in a particular community. As in the personalized recommendation scenario, the introduction of new users or new items can cause the cold start problem, as there will be insufficient data on these new entries for the collaborative filtering to work accurately. In order to make appropriate recommendations for a new user, the system must first learn the user's preferences by analysing past voting or rating activities. The collaborative filtering system requires a substantial number of users to rate a new item before that item can be recommended.

5. Challenges in Collaborative Filter

There are Several Challenges as Explained below.

1. Data Sparasity
2. Scalability
3. Synonyms
4. Gray Sheeps
5. Shilling Attack
6. Diversity and Long Tail

5.1 Data Sparasity

In practice, many commercial recommender systems are based on large datasets. As a result, the user-item matrix used for collaborative filtering could be extremely large and sparse, which brings about the challenges in the performances of the recommendation. One typical problem caused by the data sparsity is the cold start problem. As collaborative filtering methods recommend items based on users' past preferences, new users will need to rate sufficient number of items to enable the system to capture their preferences accurately and thus provides reliable recommendations.

Similarly, new items also have the same problem. When new items are added to the system, they need to be rated by a substantial number of users before they could be recommended to users who have similar tastes to the ones who rated them. The new item problem does not affect content-based recommendations, because the recommendation of an item is based on its discrete set of descriptive qualities rather than its ratings.

5.2 Scalability

As the numbers of users and items grow, traditional CF algorithms will suffer serious scalability problem. For example, with tens of millions of customers {\displaystyle O(M)} and millions of items {\displaystyle O(N)}, a CF algorithm with the complexity of {\displaystyle n} is already too large. As well, many systems need to react immediately to online requirements and make recommendations for all users regardless of their purchases and ratings history, which demands a higher scalability of a CF system. Large web companies such as Twitter use clusters of machines to scale recommendations for their millions of users, with most computations happening in very large memory machines.

5.3 Synonyms

Synonyms refers to the tendency of a number of the same or very similar items to have different names or entries. Most recommender systems are unable to discover this latent association and thus treat these products differently. For example, the seemingly different items "children's movie" and "children's film" are actually referring to the same item. Indeed, the degree of variability in descriptive term usage is greater than commonly suspected. The prevalence of synonyms decreases the recommendation performance of CF systems. Topic Modeling (like the Latent Dirichlet Allocation technique) could solve this by grouping different words belonging to the same topic.

5.4 Gray Sheeps

Gray sheep refers to the users whose opinions do not consistently agree or disagree with any group of people and thus do not benefit from collaborative filtering. Black sheep are a group whose idiosyncratic tastes make recommendations nearly impossible. Although this is a failure of the recommender system, non-electronic recommenders also have great problems in these cases, so having black sheep is an acceptable failure. Although this is a failure of the recommender system, non-electronic recommenders also have great problems in these cases, so having black sheep is an acceptable failure.

5.5 Shilling Attack

In a recommendation system where everyone can give the ratings, people may give lots of positive ratings for their own items and negative ratings for their competitors. It is often necessary for the collaborative filtering systems to introduce precautions to discourage such kind of manipulations.

5.6 Diversity and Long Tail

Collaborative filters are expected to increase diversity because they help us discover new products. Some algorithms, however, may unintentionally do the opposite. Because collaborative filters recommend products based on past sales or ratings, they cannot usually recommend products with limited historical data. This can create a rich-get-richer effect for popular products, akin to positive feedback. This bias toward popularity can prevent what are otherwise better consumer-product matches. A Wharton study details this phenomenon along with several ideas that may promote diversity and the "long tail." Several collaborative filtering algorithms have been developed to promote diversity and the "long tail" by recommending novel, unexpected, and serendipitous items.

6. Conclusion

In this paper, various method for the collaborative techninques named memory based, model based, hybrid and deep learning are detailed discussed with it merits and demirts. paper also discussed the use of Collaboratie filter in social media. During the paper discussion we come to encounter various challenges. The techniques named memory based requires mathematical calculation to count the rank of any individual item or inventories. Model based techniques is based on preassumption and build model. Hybrid model is going to take advantages of both of the method memory based and model based colabrative filter. The deep learning is based on the Advanced artificial intelligence techniqu es and machine learning techniques. Various challenges are also discussed in this paper.

It is also concluded that such challenges play major role in success of collaborative filter techniques. The challenge get even worse when the number of records are going to increase now a days because of advent use of internet and internet related equipments. E-commerce attracts all the user, alternatively extensive load on the data storage, effective cleansing policy, back up and maintance required. The system once launched what about the performance when the number of users are going to increase instantnously is major throwback for the collaborative filter. It is also observed that number of object in the internet have common name but used in different places which may create a confusion in the mind of the targeted audiance. There is also huge number of user who has not made any judgement about the product. such user wrongly feed opinion. That makes system even further complicated. There are also group of free member who intentially forces others to give opinions in one direction to win the compition. Such issues must be resolved.

References

1. Ferrari Dacrema, Maurizio; Cremonesi, Paolo; Jannach, Dietmar (2019). "Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches". Proceedings of the 13th ACM Conference on Recommender Systems. ACM: 101–109. arXiv:1907.06902. doi:10.1145/ 3298689.3347058. hdl:11311/ 1108996. ISBN 9781450362436. Retrieved 16 October 2019.
2. Liang, Dawen; Krishnan, Rahul G.; Hoffman, Matthew D.; Jebara, Tony (2018). "Variational Autoencoders for Collaborative Filtering". Proceedings of the 2018 World Wide Web Conference. International World Wide Web Conferences Steering Committee: 689–698. doi:10.1145/3178876.3186150. ISBN 9781450356398.
3. Adamopoulos, Panagiotis; Tuzhilin, Alexander (January 2015). "On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected". ACM Transactions on Intelligent Systems and Technology. 5 (4): 1–32. doi:10.1145/2559952.

4. Adomavicius, Gediminas; Tuzhilin, Alexander (1 January 2015). Ricci, Francesco; Rokach, Lior; Shapira, Bracha (eds.). Recommender Systems Handbook. Springer US. pp. 191–226. doi:10.1007/978-1-4899-7637-6_6. ISBN 9781489976369.
5. An integrated approach to TV & VOD Recommendations Archived 6 June 2012 at the Wayback Machine
6. Collaborative Filtering: Lifeblood of The Social Web Archived 22 April 2012 at the Wayback Machine.
7. Das, Abhinandan S.; Datar, Mayur; Garg, Ashutosh; Rajaram, Shyam (2007). "Google news personalization". Proceedings of the 16th international conference on World Wide Web - WWW '07. p. 271. doi:10.1145 /1242572.1242610 . ISBN 9781595936547.
8. Fleder, Daniel; Hosanagar, Kartik (May 2009). "Blockbuster Culture's Next Rise or Fall: The Impact of Recommender Systems on Sales Diversity". Management Science. 55 (5): 697–712. doi:10.1287/mnsc.1080.0974. SSRN 955984.
9. Francesco Ricci and Lior Rokach and Bracha Shapira, Introduction to Recommender Systems Handbook, Recommender Systems Handbook, Springer, 2011, pp. 1-35
10. Ghazanfar, Mustansar Ali; Prügel-Bennett, Adam; Szedmak, Sandor (2012). "Kernel-Mapping Recommender system algorithms". Information Sciences. 208: 81–104. CiteSeerX 10.1.1.701.7729. doi:10.1016/j.ins.2012.04.012.
11. He, Xiangnan; Liao, Lizi; Zhang, Hanwang; Nie, Liqiang; Hu, Xia; Chua, Tat-Seng (2017). "Neural Collaborative Filtering". Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee: 173–182. arXiv:1708.05031. doi:10.1145/ 3038912.3052569. ISBN 9781450349130. Retrieved 16 October 2019.
12. John S. Breese, David Heckerman, and Carl Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, 1998 Archived 19 October 2013 at the Wayback Machine
13. Jump up to:a b c Recommender Systems - The Textbook | Charu C. Aggarwal | Springer. Springer. 2016. ISBN 9783319296579.
14. Jump up to:a b Terveen, Loren; Hill, Will (2001). "Beyond Recommender Systems: Helping People Help Each Other" (PDF). Addison-Wesley. p. 6. Retrieved 16 January 2012.
15. Ludewig, Malte; Mauro, Noemi; Latifi, Sara; Jannach, Dietmar (2019). "Performance Comparison of Neural and Non-neural Approaches to Session-based Recommendation". Proceedings of the 13th ACM Conference on Recommender Systems. ACM: 462–466. doi:10.1145/3298689.3347041. ISBN 9781450362436. Retrieved 16 October 2019.
16. Pankaj Gupta, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Bosagh Zadeh WTF: The who-to-follow system at Twitter, Proceedings of the 22nd international conference on World Wide Web
17. Xiaoyuan Su, Taghi M. Khoshgoftaar, A survey of collaborative filtering techniques, Advances in Artificial Intelligence archive, 2009.