A SURVEY ON CLASSIFICATION OF RECOMMENDATION SYSTEMS

M. kotinaik[1], *Mrs. N. Rajeswari* [2]

[1]*M.Tech, CSE Dept, Gudlavalleru Engineering College, Gudlavalleru, India*

[2] *Associate Professor, CSE Dept, Gudlavalleru Engineering College, Gudlavalleru, India*

Abstract: *The current generation is finding it difficult to find the right information from the enormous amount of data they are presented with in the online platforms. It is hard to spent time online searching for information in such a scenario and it craves for the need of an information filtering system that could help them discover the information they seek. A research field that does this has emerged in the last few years called as recommender systems. Recommendation systems are a subset of data filtering system that used to foresee the preference that client would encourage to an item. As of late E-business applications are generally utilizing Recommender system. Generally the most popular E-commerce sites are probably music, news, books, inquire about articles, and items. Recommender systems are likewise accessible for business specialists, jokes, financial services, life coverage policies and twitter followers. Recommender systems have defined in parallel with the web. At first recommender systems depend on demographic, content-based filtering and collaborative filtering. Recommendation systems deals with enormous amount of data to identify interest of users and makes the information search easily. A lot of extensive research is happening in the field which is trying to incorporate more attributes to give more precise and relevant personalized recommendations to a user. This paper is focused on reviewing some significant works in the three basic recommender system types including collaborative filtering, content based filtering and hybrid filtering. This paper also have identified and listed the major challenges faced by recommender systems.*

Keywords: Recommender Systems (RS); collaborative filtering(CF); content based filtering(CBF); hybrid filtering(HF); K-nearest neighbors

I. INTRODUCTION

The explosive growth of internet has resulted in a phenomenon known as information abundance. In a way we are drowning in information but starving for knowledge, and it is mainly due to influx of data into the internet caused by people on one side and the scarcity of techniques to process the data to knowledge on the other side. So the current scenario demands new techniques that can assist us to discover resources of interest among the enormous options we are presented with. All of this paved way for the introduction of recommender systems which attempt to recommend items of interest to particular users by foreseeing a client's interest for an item based on related data about the items, the clients and the associations among items and users[1].

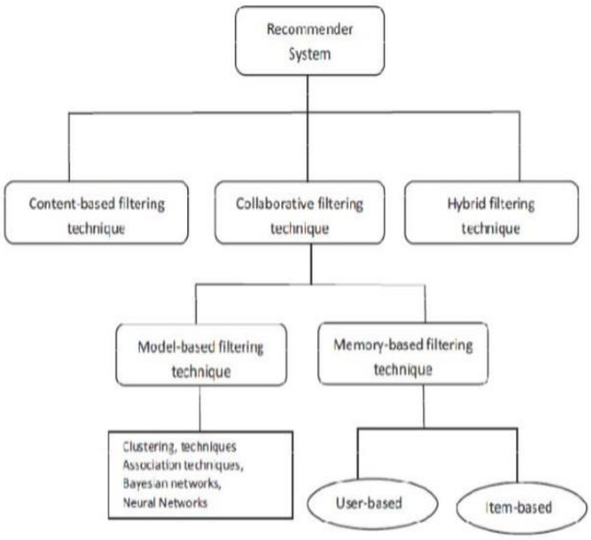
The first research paper in recommender systems came out in the mid 1990s and since then research in this area got diversified and various approaches [2] were introduced to present better recommendations. Recommender System algorithms basically performs information filtering and can be classified into three types, namely CF, CBF and HF [3][4]. With time newer strategies evolved from the basic categories with improved recommendations by including the social information, information from internet of things, location information, and genetic algorithm based methods etc. A lot of work has happened in this area over the last decade on both industry and academia. Recommender systems still remains an area of high interest as it constitutes an issue rich area and the possibilities it offer for practical applications. A wide range of applications including recommendations in web search, books, movies, music, restaurants, food, apparels, vehicles, targeted advertisements, medicines, news, potential customers for companies and many more. Recommender systems are broadly utilized by e-commerce sites to improve the user experience and there by benefitting the stores. The system is able to convert browsers to buyers and cross-sell more items by means of suggestions while shopping. It increases the user loyalty by enabling them to purchase items in fewer clicks and also providing the frequent customers with good deals and Offers. In short a recommender system is able to attract the interest of the customers by providing them fast and accurate recommendations.

A. Recommender System Fundamentals

Recommender Systems are able to neutralize the effect of information overload to a great extend by filtering vital information or data fragment from the big amount of dynamically generated information or data. A recommender system is intelligent enough to predict for a user his preference of one item over another [5]. It is this property of Recommender systems that enables it to give personalized recommendations to users. A recommender system takes into account a combination of multiple factors to provide good recommendations. They include the type of data available for the system, the algorithm used for filtering, model used, the technique used including Bayesian networks, genetic algorithms, probabilistic approaches, nearest neighbor strategy etc. The results of the recommender system is also influenced by the system performance, sparsity of the database, objective of the system and finally the quality of results the system has targeted [6].

B. Types of Recommender Systems

A broad classification of recommender system classifies it into three categories, namely CF, CBF and HF. Figure 1 gives an outline of the classification.



a) Collaborative Filtering(CF)

It is largely based on the human psychology of a person asking friends and family for suggestions about something they own, so that it helps the person to make a decision.

Collaborative filtering method can be divided into two categories: memory-based and model-based [8][9]. Predictions for memory based approach make use of the user database completely. Statistical methods are used by the system to find the like- minded set of clients or neighbors who share related interests with the active user or client [10]. The implementation of a memory based system can either be item-item or user-user based. Recommendation Systems make use of different sources of data for providing users with prediction and recommendation of items. They try to sense of balance factors like correctness, novelty; dispersity and stability in the recommendations.CF methods play a main role in the recommendation.

In item- item collaborative filtering we are interested in the relationship between different items that are purchased together. If two or more items appear in the shopping cart very frequently of different users then those items most probably share a close relationship(Eg: Bread and Jam or Bread and Butter or Peanut Butter and Jam etc). So if once the relationship is established then the next time a user adds bread to his cart he'll be given jam or peanut butter as recommendations. These recommendations make more sense than recommending something totally unrelated.

In user-user collaborative filtering the focus is shifted towards the users rather than items. We find the similarity between the users based on their purchase behavior and ratings. This is done by having a user profile defined for every user which grows with the communication of the user with the system. Similarity shared between the users is one of the driving factors of recommender systems. If a group of users share similar interest then some items liked by one user might not be rated or used by the other user, so recommending that item to the user not rated it has a very high probability of acceptance by the new user. This is also a very successful way of recommending items to users.

In contrast to memory-based approach, the model based collaborative filtering method uses the user database to study a model which is in turn used for making predictions. When designing a model that is capable of making predictions to a user the strength of both data mining and machine learning algorithms are collectively used [11].

CF approaches often suffer from three problems: cold start, scalability, and sparsity.

Cold start: These systems often require a large measure of existing information on a user with the end goal to make accurate recommendations.

Scalability: In huge numbers of the environments in which these systems make recommendations, there are a huge number of users and items. Thus, a large amount of computation power is often necessary to calculate recommendations.

Sparsity: The number of items sold on real e-commerce sites is extremely large. The most active users will just have rated a little subset of the overall database. In this way, even the most prevalent items have very few evaluations.

When building a model from a user's performance, a distinction is often made between explicit and implicit forms of data gather.

Examples of explicit data collection include the following:

- ✓ Ask a user to rate an item on a descending scale.
- ✓ Ask a user to search.
- ✓ Ask a user to rank a collection of items from most wanted to least most wanted.
- ✓ Presenting two items to a user and asking him or her to decide the better one of them.
- ✓ Ask a user to make a list of items that he or she likes.

Examples of implicit data collection include the following:

- ✓ Observing the items that a user view in an online store.
- ✓ Analyzing item or user viewing times.[35]
- ✓ Keeping an evidence of the items that a user purchases online.

- ✓ Obtaining a list of items that a user has listened to or watched on his or her computer.
- ✓ Analyzing the user's social network and discovering related likes and dislikes.

(i) Model based technique

Memory-based recommendation systems are not generally as quick and adaptable as we might want them to be, particularly with regards to genuine systems that create ongoing recommendations based on substantial datasets. To accomplish these objectives, model-based recommendation systems are used.

Model-based recommendation systems include building a model based on the dataset of evaluations. As it were, we separate some data from the dataset, and utilize that as a "model" to make recommendations without utilizing the total dataset every time. This methodology potentially offers the advantages of both speed and scalability.

A model-based system such as this also often allows trimming of the model to make the system more scalable. In particular, we can limit the number of similar entities (users or items) that we store for each entity; in other words we store only k most similar entities. Researchers have found that storing a limited number of similar entities often has little effect on the accuracy of predictions.

Advantages

Scalability: Most models coming about because of model-based algorithms are considerably littler than the genuine dataset, so that even for very large datasets, the model ends up being little enough to be utilized productively. This confers scalability to the overall system.

Prediction speed: Model-based systems are likewise liable to be quicker, in any event in contrast with memory-based systems in light of the fact that, the time required to question the model (instead of the entire dataset) is typically substantially littler than that required to inquiry the entire dataset.

Avoidance of over fitting: If the dataset over which we build our model is agent enough of genuine information, it is less demanding to attempt to maintain a strategic distance from over fitting with model-based systems.

Disadvantages

Inflexibility: Because building a model is often a period and asset expending process, it is normally more hard to add information to model-based systems, making them inflexible.

Quality of predictions: The fact that we are not using all the information (the whole dataset) available to us, it is possible that with model-based systems, we don't get predictions as accurate as with model-based systems. It should be noted, however, that the quality of predictions depends on the way the model is built. In fact, as can be seen from the results page, a model-based system performed the best among all the algorithms we tried.

(ii) Memory based technique

Memory-based algorithms approach the collaborative filtering problem by using the entire database. it tries to find users that are similar to the active user , and uses their preferences to predict ratings for the active user.

Advantages

- The quality of predictions is rather good.
- This is a relatively simple algorithm to implement for any situation.
- It is very easy to update the database, since it uses the entire database every time it makes a prediction.

Disadvantages

- It uses the entire database every time it makes a prediction, so it needs to be in memory it is very, very slow.
- Even when in memory, it uses the entire database every time it makes a prediction, so it is very slow.
- It can sometimes not make a prediction for certain active users/items. This can occur if the active user has no items in common with all people who have rated the target item.

- Over fits the data. It takes all random variability in people's ratings as causation, which can be a real problem. In other words, memory-based algorithms do not generalize the data at all.

b) ***Content based Filtering***

This method of filtering relies on two significant piece of information to provide recommendations. The first information used by the method is the attributes that are assigned to each of the items which give additional information regarding the items. The second information which is used is the user profile which gives the details of the items with which the user has interacted in the past along with its attributes. The more commonly occurring attributes among multiple objects for the user is weighted high over the others. These attribute weighting along with the history of the user is used to make a user preference model. This model is compared with all objects in the database and scores are assigned based in its correspondence to the user profile. Recommendations are made based on this scoring [12] [13].

c) ***Hybrid Filtering***

This method of filtering combines the advantages of both Collaborative and content based filtering and can avoid their Individual limitations [7]. There are different possible ways of combining collaborative and content based filtering methods into a hybrid system. The classifications are as follows:

1. Collaborative and content based filtering implemented separately and later combining their results.
2. Use of collaborative filtering properties in content based method and vice versa.
3. A combination model which developed which combines both collaborative filtering and content based filtering properties.

This paper attempts to review the various algorithms that were proposed for providing recommendations and classifying them based on the methodology used and the application area. Many papers in the area of recommender systems that were published in the past few years were collected and studied. The review also proposes a novel recommender technique which is expected to offer better results over the existing popular methods.

The various sections in the paper are organized as section II discusses the various existing methods followed by the proposed work presented in section III. In the next section various challenges addressed by recommender systems are listed. And finally in section V the conclusion and future scope is presented. A hybrid recommender system is one that combines multiple techniques together to get some synergy between them.

Collaborative: The system produces recommendations utilizing just data about rating profiles for various clients or things. Collaborative systems find peer clients/items with a rating history like the present client or item and create recommendations utilizing this neighborhood. The user based and the item based closest neighbor algorithms can be joined to manage the cold start problem and enhance recommendation results.

Content-based: The system creates recommendations from two sources: the features related with items and the rating that a user has given them. Content-based recommenders regard recommendation as a client particular order issue and take in a classifier for the client's preferences based on item features.

Demographic: A demographic recommender gives recommendations based on a demographic profile of the client. Suggested items can be delivered for various demographic specialties, by joining the evaluations of clients in those specialties.

Knowledge-based: A knowledge-based recommender proposes items based on inferences about a client's needs and preferences. This knowledge will some of the time contain unequivocal useful knowledge about how certain item features address client issues.

The term hybrid recommender system is utilized here to depict any recommender system that consolidates various recommendation methods together to deliver its output. There is no motivation behind why a few distinct methods of a similar kind couldn't be hybridized. Content-based recommenders could cooperate, and various activities have examined this kind of hybrid: NewsDude, which utilizes both guileless Bayes and kNN classifiers in its news recommendations, is only one example.

Seven hybridization strategies:

Weighted: The score of various recommendation components are joined numerically.

Switching: The system picks among recommendation components and applies the chosen one.

Mixed: Recommendations from various recommenders are exhibited together to give the recommendation.

Feature Combination: Features got from various knowledge sources are joined together and given to a solitary recommendation calculation.

Feature Augmentation: One recommendation procedure is utilized to process a feature or set of features, which is then piece of the contribution to the following method.

Cascade: Recommenders are given strict need, with the lower need ones breaking ties in the scoring of the higher ones.

Meta-level: One recommendation method is connected and creates a type of model, which is then the info utilized by the next method.

II. BACKGROUND STUDY

The initial works on recommenders were using collaborative filtering that recommended news articles to users and music album and artist recommendations from social information [15]. It was followed by a lot of works in the field of recommender systems which helped users to find products, services or content such as books, movies, music, television shows, electronic or digital products etc by applying various algorithms which reviews the different users and items to give proper suggestions [16] [17] [18]. The research works in the ground of recommender systems will be organized inside this paper based on its type.

A. Collaborative filtering

Majority of the works in recommender systems is concentrated on collaborative filtering based techniques. A work proposed by Sarwar et al [19] makes use of the entire user database and also applies statistical methods on the database to find out similar users who share similar interest. G Zhuo et al proposed a framework which combines both collaborative filtering and case based reasoning to improve the recommendations of the system. They have made use of two different algorithms MIFA and RAA to ensure the improved performance and validated the same. In [21] the method was able to forecast the votes of the active user based on partial information about the user or client and the weights calculated from user database. Konstan et al [22] used the Pearson Correlation Coefficient to calculate the weights showing the relation between the active user and other users. A personalized collaborative filtering was proposed in [23] that apply to web services implemented by computing the similarity. A hybrid collaborative technique was developed by them which combines both user and item based concept.Qian Wang et al [24]developed a user model which uses a combination of demographic information and item combination features. The model searches for a set of neighboring users who share similar interest. The accuracy is improved by using genetic algorithm to compute the weights resembling similarity among users. An Association Cluster Filtering (ACF) was proposed in [25] which uses ratings matrix to establish cluster models and assumes users in the same cluster share similar interests and different users in different clusters have less interests in common.

Unknown rating prediction is possible if an item in a cluster has more ratings to its credit. It will also enable to deduce conclusions about the item. This works well on a sparse dataset.In [26] a cascading hybrid approach was proposed which combines the features, demographic information and ratings about an item and claimed to have addressed the shortcomings of both collaborative and content based filtering. A method that was proposed in [27] adds the concept of time context to its collaborative filtering algorithm. This enhancement has improved the performance and accuracy of the recommendations. Another method [28] effective on sparse data was proposed by Ibrahim et. al. used a combination of global data and item based values to provide better results. This score was used in objection to the explicit ratings which were normally used. The results showed significant improvement over the Netflix's system for movie recommendations. Netflix also conducted a very popular competition [32] aimed at improving its existing algorithm.

*B. **Content-based Filtering***

The very first works on content based filtering was expected to be the contributions of [29] [30] [31] which were information retrieval and filtering techniques and later on it was extended by other researchers to introduce more innovations. Normally content based techniques are used on text based data for their recommendations and the content being mainly contributed by the keywords. A Fab system proposed by M. Balabanovic et al.[33]recommends web pages and it achieves its recommendation by representing web page content with the 100 most main words. Another work [34] again that recommends documents uses the most informative 128 words to represent a document. The importance of a keyword is calculated by using a weighting scheme which can be implemented in many ways, but the popular one being term frequency or opposite document frequency (TF-IDF) estimate [35]. Content based recommender system derives its recommendations mainly based on the previous ratings of the user and hence it maintains a content based summary for every user. In order to build the content based profile many techniques are available one of which Rocchio algorithm [36] and it is using an averaging approach that calculates the average vector from individual content vectors. Another work [34] estimates the probability of an item or thing to be liked by the user using Bayesian classifier. If an item or thing has listed many features a work by N. Littlestone et al. [37] has exhibited good results. Other machine learning methods like clustering and neural networks can also be used [34].Other works in the field of text retrieval has also contributed to content based filtering . Research, one such being adaptive filtering [38][39] which identifies related documents by scanning the documents one by one from a flow of documents. Another work by S. Robertson and S. Walker [40] uses a threshold to determine the relevance of a document to the user. The query has to satisfy a certain degree of match with the documents to become relevant.

C. ***Hybrid Recommender Systems***

Hybrid recommender system is gaining popularity recently as it is not confined to one method alone and uses a combination of methods to offer better results and accuracy. It is able to counter the disadvantage caused by using a single method. Robin Burke [49] observed that any hybrid technique that is used will fall under one of the seven categories namely weighted, switching, mixed, feature combination, cascade, feature augmentation and meta-level. A work in [41] used a combination of single Valued Decomposition technique and demographic information to improve the collaborative filtering technique. A.B. Barragáns-Martı´nez et. al [42] proposed a method which combines the properties of both collaborative and content based filtering. Genetic algorithms [43] have also inspired works on hybrid filtering. Another technique proposed by Al-Shamri et. al. in [44]is a hybrid system. It made use of a fuzzy based genetic approach. A method demonstrated by M. Lee and Y. Woo [45] used a collaboration of neural networks and collaborative filtering concepts. An effective use of Bayesian networks was used in [46] for implementing a hybrid

Approach with offered better results over the existing collaborative or content based schemes on individual implementation. A clustering algorithm based on centering-bunching was used in [47] to implement a hybrid personalized recommender system. M. Saranya and T. Atsuhiro[48] came up with their version of hybrid system by using latent features which was highly appreciated.

III. CHALLENGES IN RECOMMENDER SYSTEMS

Recommender System recommendations are not perfect .And it faces many shortcomings out of which few are listed below.

A. *Cold Start Problem*

It refers to a condition where the recommender system is not able to make relevant predictions or recommendations due to the lack of initial ratings about a user or an item[49]. It can commonly occur in two situations; a new item or a new user gets added into the system. The new item issue occurs because a new item added in the system is not having any ratings initially [50][51]. The probability of recommending an unrated item is very low and hence they might go unnoticed. One of the possible ways of tackling this situation is by having a rest of motivated users who will be answerable for rating every new item.

The reason for new user problem is the lack of ratings for a user new into the environment. In that scenario, it is not possible to recommend anything to such a user [52] [53]. Also when the user enters their first ratings into the system they expect to start getting recommendations which does not happen. It is because the number of ratings given to the system by the user is not sufficient enough to make good recommendations. So the probability of a new user leaving the System is high.

B. *Sparsity*

Sparsity is yet another problem encountered by recommender systems and this occurs mainly due to the fact that the no of items available to be rated is very high when compared to the amount of items previously rated by the user. So when a user item matrix is populated only a very few entries will be marked which causes the matrix to be sparse leading to poor recommendations [54]. One of the possible solutions to this problem is by giving recommendations to a user by referring to the similarity in user profiles which assumes that if two users share similar interests, it is not really necessary to deduct conclusions solely based on the similarity of items they rated. This type of filtering is known as demographic filtering [55]. Another method of addressing the sparsity problem was proposed in [56], which used Singular Value Decomposition (SVD) to reduce the dimensionality of sparse rating matrix.

C. *Scalability*

Scalability issue arises as the number of users, items and ratings information grows day by day. Even with the growing amount of information recommender systems are expected to respond quickly with recommendations for the online customers and it demands a higher scalability. The implementation of such system becomes complex and costly. The key challenge is in designing an efficient learning algorithm which is capable of handling such large datasets which keeps on growing.

One of the solutions proposed is to use an online learning algorithm [57] which processes the updates related to each user immediately and sequentially.Another method [58] proposed to address the scalability issue uses a distributed algorithm where the computations are done in parallel in multiple machines.

This paper attempts to review the various algorithms that were proposed for providing recommendations and classifying them based on the methodology used and the application area. Many papers in the area of recommender systems that were published in the past few years were collected and studied. The review also proposes a novel recommender technique which is expected to offer better results over the existing popular methods.

The various sections in the paper are organized as section II discusses the various existing methods followed by the proposed work presented in section III. In the next section various challenges addressed by recommender systems are listed. And finally in section V the conclusion and future scope is presented.

D. ***Overspecialization***

This is one of those challenges of recommender system which causes a user to lose interest of the system. Here the items similar to those rated high by the user is given as recommendations which also means that the user might have already bought or experienced the item. Hence the recommendations will not shed much interest on the user and there is a very high probability that user might leave using the system because it is not able to be of much use to the user. One way of handling the issue was proposed in [59] which uses a neighborhood based collaborative filtering. Other solutions include introducing some randomness using various randomness measures, using genetic algorithms or by eliminating similar items.

E. ***Serendipity***

Serendipity is a very crucial objective that every recommender system strives to achieve. It is all about gaining the user trust and loyalty. The user will be provided with novel and relevant recommendations which are considerably variance from the items that the user has already rated. It is difficult to conceive the idea of serendipity completely as the concept itself is very subjective and such encounter is very rare in real world scenarios. It is worth noting that there is no consent on which serendipity meaning and assessment metric to be used. Various solutions have been proposed which attempts to introduce serendipity in the recommendations; they include re-ranking the results of any accuracy oriented algorithms [59] to produce relevant scores, Full Aura list [60] algorithm which generates rank and integrates them into the ranked list of items using their linear combination etc.

IV. CONCLUSION AND FUTURE WORK

The research in recommender system is directed on the right path of improving the relevance and accuracy of personalized recommendations. Many promising works are also proposed and implemented in the last few years. But the challenges faced by recommender systems were not addressed completely and there is a lot of room for improvement. This paper has attempted to list some of the significant works in the field and propose a novel hybrid approach that can confront some of the drawbacks of recommender systems. This paper has made use of fuzziness, dimensionality reduction and clustering approaches to improve the recommendation quality. Always a good input leads to better results.

IV. REFERENCES

[1] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowledge-Based Systems, 46 (2013) 109-132.

[2] Deshpande, M., and Karypis, G. Item-based top-n recommendation algorithms. ACM Trans. Inf. Syst. 22, 1 (2004), 143-177.

[3] M. Pazzani, A framework for collaborative, content-based, and demographic filtering, Artificial Intelligence Review-Special Issue on Data Mining on the Internet 13 (5-6) (1999) 393–408.

[4] R. Burke, Hybrid recommender systems: survey and experiments, User Modeling and User-Adapted Interaction 12 (4) (2002) 331–370.

[5] Pan C, Li W. Research paper recommendation with topic analysis. In Computer Design and Applications IEEE 2010;4, pp. V4-264.

[6] www.jstage.jst.go.jp