



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 4, Issue 12, December 2016

A Contemporary Study in Development Trustworthy Recommender Systems

K.A.Balasubramaniam¹, Dr.M. Chidambaram²

Ph.D Research Scholar, Bharathiyar University, Coimbatore, Tamil Nadu, India.

Assistant Professor, Department of Computer Science, Rajah Serfoji Government Arts College, Thanjavur, Tamil Nadu, India.

ABSTRACT: The modern technology that has been adopted by mainly recommender systems is content based collaborative filtering. Currently, these systems are incorporating social information. However, with the open nature of collaborative filtering recommender systems, they suffer significant easily persuadable from being attacked by malicious raters, who inject profiles consisting of biased ratings. In recently several attack detection algorithms have been proposed to handle the issue. In the future, they will use implicit, local and personal information from the Internet of things. Recommendation techniques are very important in the fields of E-commerce and other Web-based services. One of the main difficulties is dynamically providing high-quality recommendation on bare data. In this paper, a novel dynamic personalized recommendation algorithm is proposed, in which information contained in both ratings and profile contents are utilized by exploring latent relations between ratings, a set of dynamic features are designed to describe user preferences in multiple phases, and finally a recommendation is made by adaptively weighting the features. This paper provides an overview of recommender systems as well as collaborative filtering methods and algorithms; it also explains their evolution, provides an original classification for these systems, identifies areas of future implementation and develops certain areas selected for past, present or future importance. Unfortunately, their applications are restricted by various constraints. Recommender systems suggest people items or services of their interest and proved to be an important solution to information overload problem.

KEYWORDS: Recommender System, Collaborative based Filtering, Content-based filtering, Hybrid Recommender Systems and Hybridization Techniques.

I. INTRODUCTION

Nowadays the internet has become an indispensable part of our lives, and it provides a platform for enterprises to deliver information about products and services to the customers conveniently. As the amount of this kind of information is increasing rapidly, one great challenge is ensuring that proper content can be delivered quickly to the appropriate customers. Personalized recommendation is a desirable way to improve customer satisfaction and retention. There are mainly three approaches to recommendation engines based on different data analysis methods, i.e., rule-based, content-based and collaborative filtering. Among them, collaborative filtering (CF) requires only data about past user behavior like ratings, and its two main approaches are the neighborhood methods and latent factor models. The neighborhood methods can be user-oriented or item-oriented. They try to find like-minded users or similar items on the basis of co-ratings, and predict based on ratings of the nearest neighbors. Latent factor models try to learn latent factors from the pattern of ratings using techniques like matrix factorization and use the factors to compute the usefulness of items to users. CF has made great success and been proved to perform well in scenarios where user preferences are relatively static.

In most dynamic scenarios, there are mainly two issues that prevent accurate prediction of ratings – the sparsity and the dynamic nature. Since a user could only rate a very small proportion of all items, the $U \times I$ rating matrix is quite sparse and the amount of information for estimating a candidate rating is far from enough. While latent factor models involve most ratings to capture the general taste of users, they still have difficulties in catching up with the drifting signal in dynamic recommendation because of sparsity, and it is hard to physically explain the reason of the involving. The

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 4, Issue 12, December 2016

dynamic nature decides that users' preferences may drift over time in dynamic recommendation, resulting in different taste to the items in different phases of interest, but it is not well studied in previous studies. In our experiences, the interest cycle differs from user to user, and the pattern how user preferences changes cannot be precisely described by several simple decay functions. Moreover, CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high.

Some researchers have previously attempted to solve the above problems. Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsity problem where more information were mined than just in each of them. A classified item into many categories using content information and chose recent categories to perform Item-Based Collaborative Filtering (IBCF). An introduced group similarity by clustering and used it to modify original item-item similarity matrix. Some approaches emphasize utilization of time information to deal with the dynamic nature. The proposed to model temporal dynamics to separate transient factors from lasting mones.

In this paper, we present a hybrid dynamic recommender system. First we use more information while keeping the data consistency; we use user profiles and item contents to extend the co-rate relations between ratings through each element of users, as show in Fig.1. The ratings can reflect similar users' preferences and provide useful information for recommendation. Correspondingly, in order to enable the algorithm to maintain the changing of signals quickly and to be updated conveniently, based on time series analysis(TSA) technique a set of dynamic features are proposed, and relevant ratings in each phase of interest are added up by applying TSA to describe user's preferences and item's reputations. Then we proposed a personalized recommendation algorithm by adaptively weighting. The result of the proposed algorithm is effective with dynamic data and per- forms better than the previous algorithms.

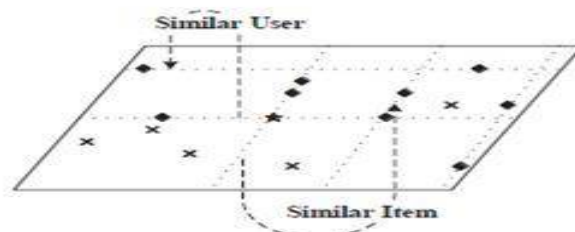


Figure1. Hybrid dynamic recommender system

II. RELATED WORK

Dynamic recommendation in traditionally RMSE evaluations (even for the Netflix competition), training and testing data are randomly sampled and the train and test split is not based on time. This would produce current prediction based on future data. Even if it is guaranteed that testing instances of each user/item come later than its training instances, the aforementioned issue still exists in algorithms like IBCF and latent factor models due to the utilization of other users' future ratings. The CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high. Some researchers have previously attempted to solve the above problems. Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsely problem where more information were mined than just in each of them. et al. classified items into many categories using content information and chose recent categories to perform Item-Based Collaborative Filtering (IBCF). Kim and Li introduced group similarity by clustering and used it to modify original item-item similarity matrix

Ontology based Recommender System: In the peer to peer network (P2P network) is based on decentralized architecture has the progress of ontology based recommender system. This is basically works with dynamically changing large scale environment. In a ontology based multilayered semantic social network, is introduced. This model works on a set of users having similar interest and the correlation at different semantic levels.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirce.com

Vol. 4, Issue 12, December 2016

Collaborative Tagging based Recommender System: In the collaborative tagging based recommender allows users particularly consumers to freely connect tags or keywords to data contents. In a generic model of collaborative tagging to recognize the dynamics behind it. The tag based system suggests the use of high quality tags, by which spam and noise can be avoided.

Dynamic Content: We consider not only the item set undergoes insertions and deletions frequently, but also the content value and then the appraisalment from users are changing rapidly as well. For example, the lifetime of breaking news on the Internet is usually a couple of hours, and the value of the news (such as click through rate) is decaying temporally as people get to know it. Traditional recommender systems usually treat users' feedback static, so that feedback on the same items given at different time stamps is still comparable. This assumption doesn't hold on dynamic content. Rebuilding the model on very recent data is typically an expensive task, and tends to lose long-term interests of users. On dynamic content, recommender systems always face the cold start problem for new items.

Rule based content: Rule-based filtering creates a user-specific utility function and then applies it to the items under consideration. This approach is closely related to customization, which requires users to identify themselves, configure their individual settings, and maintain their personalized environment over time. It is easy to fail since the burden of responsibility falls on the users.

III. FOREMOST APPROACHES IN RECOMMENDER SYSTEMS

A. COLLABORATIVE BASED FILTERING: One of the best approaches to the design of recommender systems that has wide use is collaborative filtering. Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, activities or preferences and predicting what users will like based on their similarity to other users. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems. For example, the k-nearest neighbor (k-NN) approach and the Pearson Correlation Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past. Collaborative based filtering maintains a database of many users' ratings of a variety of items. For a given user, find other similar users whose ratings strongly correlate with the current user. Recommender system to create data items rated highly by these similar users, but not rated by the current user. Almost all existing commercial recommenders use this approach (e.g. Amazon). The recommender system compares the collected data to similar and dissimilar data collected from others and calculates a list of recommended items for the user. One of the most famous examples of collaborative filtering is item-to-item collaborative filtering (people who buy x also buy y), an algorithm popularized by Amazon.com's recommender system. Collaborative filtering approaches often undergo from three problems: cold start, scalability, and sparsity.

- ❖ Cold start: These systems often require a large amount of existing data on a user in order to make accurate recommendations. The cold-start problem occurs when it is not possible to make reliable recommendations due to an initial lack of ratings. We can differentiate three categories of cold-start problems: i) new community, ii) new item iii) new user. The last kind is the most important in recommender systems
 - i. The new community problem refers to the difficulty, when starting up recommender systems in obtaining, a sufficient amount of data (ratings) for making reliable recommendations.
 - ii. The new item problem occurs because the new items entered in recommender systems do not usually have initial ratings, and therefore, they are not likely to be recommended.
 - iii. The new user problem represents one of the great difficulties faced by the recommender systems in operation.
- ❖ Scalability: In many of the environments in which these systems make recommendations, there are millions of users and products. Thus, a large amount of computation power is often necessary to calculate recommendations.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 4, Issue 12, December 2016

- ❖ Sparsity: The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

B. CONTENT-BASED FILTERING: Another important common approach when designing recommender systems is content-based filtering. Content-based filtering methods are based on a description of the item and a profile of the user's preference. Content-based recommendations: the user is recommended items similar to the ones the user preferred in the past; In a content-based recommender system, keywords are used to describe the items and a user profile is built to indicate the type of item this user likes. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). Content based filtering uses the assumption that items with similar features will be rated similarly. It depends on the availability of item description and user profile. The profile is based on items user has liked in the past or explicit interests. A content-based recommender system matches the profile of the item to the user profile to decide on its relevancy to the user. The recommender system uses additional data about the context of item consumption. For example, in the case of a restaurant the time or the location may be used to improve the recommendation compared to what could be performed without this additional source of information. In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. This approach has its roots in information retrieval and information filtering research. A content-based filtering also generates recommendations using the content from objects intended for recommendation; therefore, certain content can be analyzed, like text, images and sound. From this analysis, a similarity can be established between objects as the basis for recommending items similar to items that a user has bought, visited, heard, viewed and ranked positively. To create a user profile, the system mostly focuses on two types of information:

1. A model of the user's preference.
2. A history of the user's interaction with the recommender system.

Basically, these methods use an item profile (i.e. a set of discrete attributes and features) characterizing the item within the system. The system creates a content-based profile of users based on a weighted vector of item features. The weights denote the importance of each feature to the user and can be computed from individually rated content vectors using a variety of techniques such as Bayesian Classifiers, cluster analysis, decision trees, and artificial neural networks in order to estimate the probability that the user is going to like the item.

A key issue with content-based filtering is whether the system is able to learn user preferences from users' actions regarding one content source and use them across other content types. When the system is limited to recommending content of the same type as the user is already using, the value from the recommendation system is significantly less than when other content types from other services can be recommended. For example, recommending news articles based on browsing of news is useful, but would be much more useful when music, videos, products, discussions etc. from different services can be recommended based on news browsing.

C. HYBRID RECOMMENDER SYSTEMS: Hybrid recommender approach methods combine collaborative and content-based methods. Recent research has demonstrated that a hybrid approach, combining collaborative filtering and content-based filtering could be more effective in some cases. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model for a complete review of recommender systems). Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 4, Issue 12, December 2016

Netflix is a good example of the use of hybrid recommender systems. They make recommendations by comparing the watching and searching habits of similar users (i.e. collaborative filtering) as well as by offering movies that share characteristics with films that a user has rated highly (content-based filtering).

A hybrid recommender system is one that combines multiple techniques together to achieve some synergy between them.

- **Collaborative:** The system generates recommendations using only information about rating profiles for different users. Collaborative systems locate peer users with a rating history similar to the current user and generate recommendations using this neighborhood.
- **Content-based:** The system generates recommendations from two sources: the features associated with products and the ratings that a user has given them. Content-based recommenders treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on product features.
- **Demographic:** A demographic recommender provides recommendations based on a demographic profile of the user. Recommended products can be produced for different demographic niches, by combining the ratings of users in those niches. The term hybrid recommender system is used here to describe any recommender system that combines multiple recommendation techniques together to produce its output.

D. HYBRIDIZATION TECHNIQUES: The mainly seven Hybridization Techniques as follows

- **Weighted:** The score of different recommendation components are combined numerically.
- **Switching:** The system chooses among recommendation components and applies the selected one.
- **Mixed:** Recommendations from different recommenders are presented together.
- **Feature Combination:** Features derived from different knowledge sources are combined together and given to a single recommendation algorithm.
- **Feature Augmentation:** One recommendation technique is used to compute a feature or set of features, which is then part of the input to the next technique.
- **Cascade:** Recommenders are given strict priority, with the lower priority ones breaking ties in the scoring of the higher ones.
- **Meta-level:** One recommendation technique is applied and produces some sort of model, which is then the input used by the next technique.

IV.EXISTING SYSTEM

The neighbourhood methods can be user-oriented or item-oriented. They try to find likeminded users or similar items on the basis of co-ratings, and predict based on ratings of the nearest neighbors. While latent factor models involve most ratings to capture the general taste of users, they still have difficulties in catching up with the drifting signal in dynamic recommendation because of sparse, and it is hard to physically explain the reason of the involving. In our experiences, the interest cycle differs from user to user, and the pattern how user preferences changes cannot be precisely described by several simple decay functions. Moreover, CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high.

Limitations of Existing System

1. Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsity problem where more information were mined than just in each of them.
2. The principle of utilization of rating data in these algorithms some approaches emphasize utilization of time information to deal with the dynamic nature.
3. The involved ratings can reflect similar users' preferences and provide useful information for recommendation.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 4, Issue 12, December 2016

V .PROPOSED SYSTEM

Use only historical data but not future data for current prediction in real applications. In traditional evaluations training and testing data are randomly sampled and the train and test split is not based on time. This would produce current prediction based on future data. The data in different phases of interest have different training ratios. It is clear that the proposed algorithm is quite robust in the phases, and we found it is not true that the more recent ratings should have heavier weights across the whole time, which illustrates the advantages of the features – light computation, flexibility and high accuracy.

The Proposed system is to make use of profiles to extend the co-rating relation, and then we propose a set of dynamic features to reflect user's preferences or item's reputations in different phases of interest, and after that we recommend an adaptive algorithm for dynamic personalized recommendation.

1. Relation Mining of Rating Data: The main complexity of capturing user's dynamic preferences is the lack of useful information, which may come from three sources - user profiles, item profiles and historical rating records during the sparsity of recommendation data. Existing algorithms mainly rely on the co-rate relation. But this will not efficient in calculation while the data is sparse as it limits the amount of data during prediction. So, to overcome this we introduce a semi co-rate relation for finding useful ratings for dynamic personalized recommendation.

2. Dynamic Feature Extraction: To compute better recommendation algorithm, three kinds of methods were proposed such as instance selection, time- window (usually time decay function) and ensemble learning. This technique contains a set of dynamic features to describe users' multi-phase preferences in consideration of computation, accuracy and flexibility.

3. Adaptive Weighting Algorithm: The parameters are quantified in the feature extraction as per the previous step, so now it's easy to organize them for accurate rating estimation by using adaptive weighting. Sizes of all the relevant subsets are also computed in MPD (Multiple Phase Division) and could reflect on data density.

- 1) To evaluate the accuracies of above mentioned dynamic recommendation algorithms as follows:
- 2) Sort the complete dataset in natural time order; use a certain training ratio to determine its corresponding splitting.
- 3) Use the previous part as the training set to adjust all parameters.
- 4) Run algorithm on this testing set and generate estimated rating for each user-item pair.
- 5) Compare each estimated ratings and real ratings within the testing set and calculate for them.
- 6) Use variety of ratios and cycle through the last four steps.

VI. COMPARING THE TRADITIONAL CLASSIFICATION OF RECOMMENDER SYSTEMS

The comparing the traditional classification of recommender systems, as summarized in Table figure 2, can be extended in numerous ways that include improving the understanding of users and items, incorporating the related information into the recommendation process, supporting multi-criteria ratings, and providing more flexible and less interfering types of recommendations. Such more widespread models of recommender systems can provide better recommendation capabilities.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirccce.com

Vol. 4, Issue 12, December 2016

Recommender System Approaches(RSA)	Recommender System Techniques(RST)	
	Heuristic-based or Memory based Techniques	Model-based Techniques
Content-based	Commonly used techniques:	Commonly used techniques:
	<ul style="list-style-type: none"> TF-IDF -Term Frequency-Inverse Document Frequency (Information Retrieval and Information Filtering) 	<ul style="list-style-type: none"> Bayesian classifiers and clustering analysis
	<ul style="list-style-type: none"> Clustering 	<ul style="list-style-type: none"> Latent features and Matrix Factorization. Fuzzy system and genetic algorithm
		<ul style="list-style-type: none"> Decision trees and Artificial Neural Networks
Collaborative based	Commonly used techniques:	Commonly used techniques:
	<ul style="list-style-type: none"> k-nearest neighbor (k-NN) approach 	<ul style="list-style-type: none"> Bayesian networks and Clustering
	<ul style="list-style-type: none"> Clustering 	<ul style="list-style-type: none"> Artificial neural networks
	<ul style="list-style-type: none"> Pearson Correlation Collaborative filtering and Graph Theory 	<ul style="list-style-type: none"> Linear regression and probabilistic models
Hybrid based	Combining content-based and collaborative components using	Combining content-based and collaborative components by:
	<ul style="list-style-type: none"> Linear combination of predicted ratings and various voting schemes. 	<ul style="list-style-type: none"> Incorporating one component as a part of the model for the other.
	<ul style="list-style-type: none"> Incorporating one component as a part of the heuristic for the other. 	<ul style="list-style-type: none"> Building one unifying model.
	<ul style="list-style-type: none"> Bioinspired or probabilistic methods such as genetic algorithms, fuzzy genetic neural networks, Bayesian networks , clustering and latent features. 	<ul style="list-style-type: none"> A widely accepted taxonomy divides recommendation methods into memory-based techniques.
	<ul style="list-style-type: none"> Memory-based methods usually use similarity metrics to obtain the distance between two users, or two items, based on each of their ratios. 	<ul style="list-style-type: none"> Bayesian classifiers, neural networks, fuzzy systems, genetic algorithms, latent features and matrix factorization.

Figure 2. Comparing the traditional classification of recommender systems research

VII. CONCLUSION AND FUTURE DIRECTIONS

In this paper, Recommender systems made a significant progress over the last decade when various content based, collaborative and hybrid methods were proposed and several industrialized strength systems have been developed. However, despite all these advances, the contemporary generation of recommender systems surveyed in this paper still requires further improvements to make recommendation methods more effective in a broader range of applications. In this paper, we reviewed various boundaries of the current recommendation methods and discussed possible extensions that can provide better recommendation capabilities. These extensions include, among others, the improved modelling of users and items, incorporation of the contextual information into the recommendation process, support for multi-criteria ratings, and provision of a more flexible and less intrusive recommendation process. We hope that the issues



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 4, Issue 12, December 2016

presented in this paper would advance the discussion in the recommender systems community about the next generation of recommendation technologies. We proposed a novel dynamic recommender system for sparse data, in which more rating data is utilized in one prediction by involving more neighbouring ratings through each attribute in user and item profiles. A set of dynamic features are designed to describe the preference information based on TSA technique, and finally a recommendation is made by adaptively weighting the features using information in different phases of interest. The proposed algorithm is highly effective, and its computational cost is much acceptable.

REFERENCES

- [1] B. M. Sarwar, G. Karypis, J. A. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: WWW, 2001, pp. 285–295.
- [2] P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web, Methods and Strategies of Web Personalization, Lecture Notes in Computer Science, Springer, 2007.
- [3] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
- [4] Y. Koren, Collaborative filtering with temporal dynamics, Communications of the ACM 53 (4) (2010) 89–97.
- [5] L. Candillier, F. Meyer, M. Boullé, Comparing state-of-the-art collaborative filtering systems, in: P. Perner (Ed.), MLDM, Vol. 4571 of Lecture Notes in Computer Science, Springer, 2007, pp. 548–562.
- [6] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, H. Kriegel, Probabilistic memory-based collaborative filtering, IEEE Transactions on Knowledge and Data Engineering 16 (1) (2004) 56–69.
- [7] F. Fouss, A. Pirotte, J. Renders, M. Saerens, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, IEEE TKDE 19 (3) (2007) 355–369.
- [8] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.
- [9] Adomavicius, G., R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach. ACM Transactions on Information Systems, 23(1), January 2005.
- [10] S. Boutemedjet, D. Ziou, Long-term relevance feedback and feature selection for adaptive content based image suggestion, Pattern Recognition 43 (12) (2010) 3925–3937.