A Survey of Recommender Systems: Approaches and Limitations

Meenakshi Sharma
Department of Information Technology, Banasthali University, Jaipur Campus
minaxisharma90@gmail.com

Sandeep Mann
Assistant Professor, Department of Computer Science, Government College, Gurgaon
sandeep.mann23@gmail.com

Abstract—Recommendation as a social process plays an important role in many applications as WWW has created the universe as a global village, with an explosive growth of enormous information. The paper presents an overview of the field of recommender systems along with the description of various approaches that are being used for generating recommendations. Recommendation techniques can be classified in to three major categories: Collaborative Filtering, Content Based and Hybrid Recommendations. The paper elaborates these approaches and discusses their limitations by describing the major problems suffered by recommendation methods. From basic techniques to the state-of-the-art, we attempt to present a comprehensive survey for recommendation techniques, which can be served as a roadmap for research and practice in this area.

Keywords - Cold Start, Collaborative Filtering, Content-Based Recommendation, Recommendation System, Sparsity Problem.

I. INTRODUCTION

Recommendations are a part of everyday life where people rely on external knowledge to make decisions about an artifact of interest. Recommender systems or recommendation systems are a subclass of information filtering system that seek to predict 'rating' or 'preference' that a user would give to an item (such as music, books or movies) or social element (e.g. people or group) they had not yet considered, using a model built from the characteristics of an item (content based approaches) or the user's social environment (collaborative filtering approaches). Although many different approaches to recommender systems have been developed in the past few years, the interest in this area still remains high due to growing demand on practical applications, which are able to provide personalized recommendations and deal with information overload. These growing demands pose some key challenges to recommender systems and to deal with these problems many advanced techniques are proposed like content boosted collaborative filtering, clustering based filtering, combining item based and user based similarity and many more. Despite of these advances, recommender systems still require improvement and thus becoming a rich research area.

In this paper, before discussing the major limitations of recommendation methods, the comprehensive survey of recommendation approaches is provided. The discussion of various approaches and their limitations in a proper manner thereby provides the future research possibilities in recommendation systems.

II. BACKGROUND

Recommender systems have been indispensable always as information growth has made it overly expensive for users to try every possible alternative independently and user generally relies on such systems to get individualized recommendation as output in large space of alternatives of books, news, movies, music, papers, TV programs and web pages etc.. An effective solution for reducing complexity when searching over internet has been given by recommender systems. A variety of approaches have been used to provide recommendations like collaborative filtering, content based, knowledge based, demographic or hybrid and many others [1].

Although the roots of recommender systems can be traced back to the extensive work in the cognitive science, approximation theory, information retrieval, forecasting theories, and also have links to management science, and also to the consumer choice modelling in marketing, recommender systems emerged as an independent research area in the mid 1990's when researchers started focusing on recommendation problems that explicitly rely on the ratings structure. The technique that uses rating structure is termed as "Collaborative Filtering" and it was introduced by Goldberg et al (1992) in the context of first commercial recommender system Tapestry which was designed to recommend documents drawn from newspapers to a collection of users. One important aspect of recommender system is personalization that makes it possible to provide personalised recommendation to user and it is usually supported by content information. In content based filtering, items or services are recommended on basis of user's previous actions or purchases (Basu et al,1998). Belkin & Croft [19] surveyed some of the first content-based recommendation systems and noted that they made use of technology related to information retrieval such as tf*idf and Rocchio's method.

Different algorithms and approaches are there to provide recommendations that may either use rating information or content information, however, both collaborative filtering and content based filtering faces certain limitations. Several researchers have attempted to overcome these limitations by proposing hybrid approaches that combines both rating as well as content information. For e.g., Pazzani proposed a hybrid approach that combine collaborative and content based filtering as well as demographic methods. Recommender system will remain an active research area, with dedicated ACM and IEEE conferences, including various disciplines like data mining, personalization, context awareness, information retrieval and group recommendations.

III. Recommendation System: General Concepts

Recommendation System is an intelligent system that makes suggestion about items to users that might interest them. Some of the practical applications that use such systems include recommending books, cd etc. on Amazon.com, movies by Movielens, music by last.fm and news at VERSIFI technologies.

A. *A Model of Recommendation Process*

This model illustrated by Fig. 1 is general model to cover broad range of recommendation activities. A recommender seeker may ask for recommendations or a recommender may produce recommendations without prompting. Seeker may volunteer their own preferences or recommender may ask about them. Based on the set of known preferences – his/her own, the seeker's and those of other people- the recommender recommends items that seeker probably may like. In addition the recommender may identify the people with similar interests. The seeker may use the recommendations to select items from the universe or to communicate with other like-minded people.

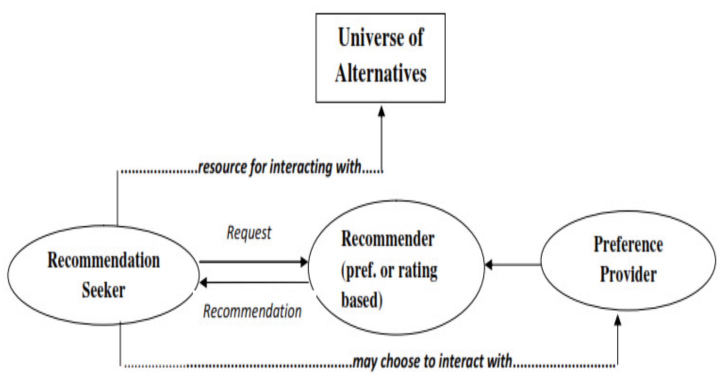

Figure 1. Model of the Recommendation Process

B. *Formal Definition of Recommender System*

In a recommendation-system application there are two classes of entities, which we shall refer to as users and items. The formal definition of recommender system is:

- *C*: The set of all users
- *S*: The set of all possible items that can be recommended, such as books, movies, or restaurants.
- U: A utility function that measures usefulness of a specific item $s \in S$ to user $c \in C$, i.e., $U: C \times S \rightarrow R$, where R is a totally ordered set.

The space *S* of possible items can be very large, ranging in hundreds of thousands or even millions of items in some applications, such as recommending books or CDs. Similarly, the user space can also be very large—millions in some cases. For each user $c \in C$, we want to choose such item $s' \in S$ that maximizes the user's utility is desired.

More formally:

$$\forall c \in C, s_c' = \arg\max_{s \in S} u(c, s) \qquad (1)$$

In recommender systems the utility of an item is usually represented by a *rating*, which indicates how a particular user liked a particular item. In general, rating is done on scale, for instance, if movies are rated on a scale of 1 to 5, then a movie rated 5 by a user means it is highly liked by user while 1 rating denotes dislike. Further, each element of user space U can be defined with a profile that includes user characteristics like userid, age, gender, occupation etc. and each element of item space S can be defined using item characteristics. For example, in a movie recommender system, a movie can be defined by its id, genre, release date, director, actors etc.

Usually, rating is not done on a complete dataset or space C×S and thus only rating on subset is available. The main aim of a recommender system is to predict ratings of the non-rated user/item combination and thus providing appropriate recommendations. A recommender system may either provide the highest estimated rating item or alternatively provide a list of top N items as recommendation to a user or set of users.

C. Recommendation System Approaches

Recommendation systems are usually classified on the basis of their approach to rating estimation:

- Collaborative Filtering System
- Content-based System
- Hybrid System

In content-based approach, similar items to the ones the user preferred in past will be recommended to the user while in collaborative filtering, items that other people with similar tastes and preferences like will be recommended. In order to overcome the limitations of both approach hybrid systems are proposed that combines both approaches in some manner.

IV. COLLABORATIVE FILTERING

Collaborative filtering (CF) systems work by collecting user feedback in the form of ratings for items in a given domain and exploiting similarities in rating behavior amongst several users in determining how to recommend an item. CF systems recommend an item to a user based on opinions of other users. For example, in a movie recommendation application, CF system tries to find other like-minded users and then recommends the movies that are most liked by them.

A. Collaborative Filtering Process

The task of traditional collaborative filtering recommender algorithm concerns the prediction of the target user's rating for the target item that the user has not given the rating, based on the users' ratings on observed items.

- CF algorithms represent the entire user-item space as a rating matrix 'R'. Each entry Rij in matrix represents the preference score (rating) if the ith user on the jth item. Each individual rating is within a numerical scale and it can be 0 as well, indicating that user has not yet rated this item.

- CF problem includes the estimation or prediction of rating for the yet unrated item. For the prediction of rating, similarities between items and users are calculated using different approaches. Thus, the two related problems consist in finding set of K users that are most similar to a given user and finding set of K items that are most similar to a given item.
- Finally using these similarities, recommendations that are produced at output interface can be of two types: Prediction and Recommendation.
- Prediction is a numerical value, Rij, expressing the predicted score of item j for the user i. The predicted value is within the same scale that is used by all users for rating.

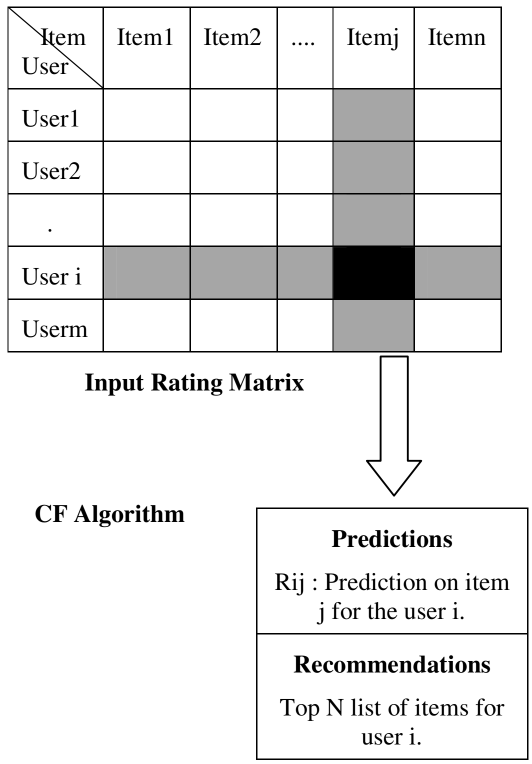

Figure 2. Collaborative Filtering Process

- Recommendation is a list of top N items that the user will like the most. This output interface is known as *Top-N recommendation.*

B. *Categories of Collaborative Filtering Techniques*

Although there are many collaborative filtering techniques,they can essentially be divided into two major categories:

- Memory-based approaches
- Model-based approaches

1) *Memory-based Approaches:* Memory-based CF uses the entire or a sample of the user-item database to generate a prediction. Every user is part of a group of people with similar interests. Such approaches has two main steps, i.e., calculation of similarities between user and items using rating information and predicting the unknown rating and thus providing either a single value or a list of top n items that the user may like.

Memory based approaches can be classified into two main types:

- User-based CF
- Item-based CF

In user based systems, the similarity between users are calculated by comparing their ratings on the same item, and then compute the predicted rating for item j by user i as a weighted average of the ratings of j by users similar to user i, where weights are the similarities of these users with i. In item-based systems, the similarity between two items is determined by comparing the rating made by same user i on the items. Then, the predicted rating of item j by user i is obtained as a weighted average of the ratings of i on items, weighted by the similarity between those items.

Two most popular similarity measures are *correlation-based* and *cosine-based.* Let S_{xy} be the set of all items co-rated by user x and y. Pearson correlation coefficient used to measure similarity is

$$sim(x,y) = \frac{\sum_{s \in S_{xy}} (r_{x,s} - \overline{r_x})(r_{y,s} - \overline{r_y})}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \overline{r_x})^2 \sum_{s \in S_{xy}} (r_{y,s} - \overline{r_y})^2}} \quad (2)$$

Where $\overline{r_x}$ is the average rating user x give to all items and $\overline{r_y}$ is the average rating user y give to all items.

However, there exists many similarity measures to compute similarity which can be used as per the context require. Some similarity measures to compute similarity: Pearson correlation, cosine vector similarity, adjusted cosine vector similarity, spearman correlation, gaussian kernel.

2) *Model-Based Approaches:* Model-based techniques provide recommendations by estimating parameters of statistical models for user ratings and do not use all the available information to make a prediction. The collection of ratings is used to learn a model of user preferences, in an offline step, which is then used to make rating predictions which are usually fast and accurate. Although they are fast and less sensitive to data sparsity, a lot of time is required to learn the model, which makes them inefficient in an online setting. The probabilistic approach is used to calculate the probability that user will give a particular rating to new item based on user's ratings of previously rated items.

Two methods to estimate this probability are:

- Cluster Models
- Bayesian Network

In cluster model, like minded users are clustered in to classes and thus clustering is used to estimate the ratings by grouping the users into different classes where the number of classes and parameter depends on the data. While in Bayesian network, items are represented as nodes and the possible rating value is determined from the state of each node. However, this approach has limitation that user can only be placed in one cluster at a time and today almost all applications require to cluster the user in several categories at a time.

Another methodology for model based recommendation includes *latent factor and matrix factorization* that assume that the similarity between users and items is simultaneously induced by some hidden lower dimensional structure in the data. Aspect Model approach is a probabilistic variant of latent Semantic Analysis, whose goal is to identify hidden semantic associations from co-occurrence data which corresponds to the ratings of items by users. Another popular model-based approach is the Personality Diagnosis method [10].

V. CONTENT-BASED RECOMMENDATION SYSTEMS

Content based recommendation systems recommend an item to a user based upon a description of the item and a profile of the user's interests. Such systems are used in recommending web pages, TV programs and news articles etc. All content based recommender systems has few things in common like means for description of items, user profiles and techniques to compare profile to items to identify what is the most suitable recommendation for a particular user.

A. Content-Based Recommendation Process

In content-based recommendation methods, the utility $u(c, s)$ of item s for user c is estimated based on the utilities $u(c, s_i)$ assigned by user c to items s_i □S that are "similar" to item s. Content-Based recommender systems make suggestions upon item features and user interest profiles. Typically, personalized profiles are created automatically through user feedback, and describe the type of items a person likes. In order to determine what items to recommend, collected user information is compared against content features of the items to examine [11]. As shown in Fig. 3:

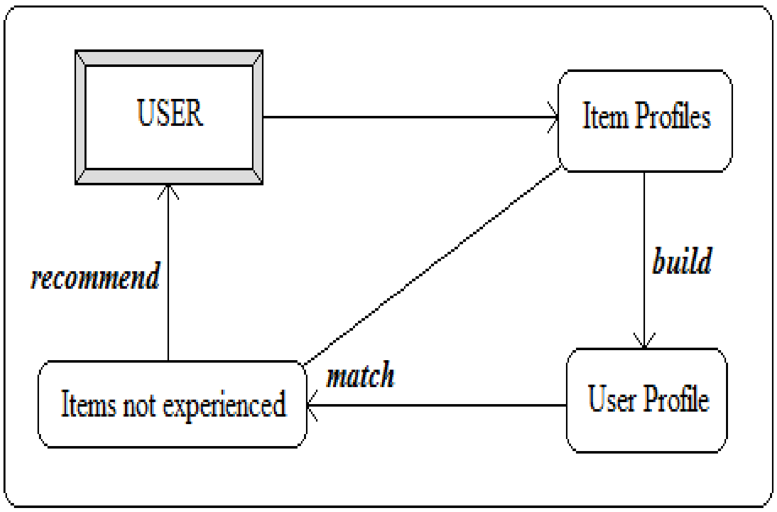

Figure 3. Content Based Recommendation

- System has a huge database consisting of the items to be recommended and the features of these items and it is termed as Item Profile.
- The users provide some sort of information about their preferences to the system. Combining the item information with the user preferences, the system builds a profile of the users.
- According to the information existing in a target user's profile, the system recommends suitable items to the user.

One can make better personalized recommendation by utilizing the features of items and users.

An item profile is defined by its important features. For instance, a movie can be described by its title, genre, language, country, actors etc. Depending upon the weighing procedure, similarity between two items can be calculated. Depending on the domain, features can be represented either by boolean values or by a set of restricted values. For example, imagine we want to analyze a set of newspaper articles about different kind of topics. While a boolean value could indicate whether a word is contained in an article or not, an integer value could express the number of times a word appears.

To build a user profile, information of a user can be used. In MovieLens dataset, users are described using demographic information that includes age, gender, occupation and zipcode. User information might be provided explicitly by the individual person or gathered implicitly by a software agent.

- Explicit user information collection basically relies on personal input by the user. A common feedback technique is the one that allows users to express their opinions by selecting a value of range. However, filling out forms or clicking checkboxes places a burden on the user. Profiles might be imprecise, because the user is not willing to spend a lot of time providing personal information or the user information is already outdated.

- Implicit feedback does not require any additional intervention by the user during the process of constructing profiles. Moreover, it automatically updates as the user interacts with the system.

Thus, systems that collect implicit feedback are more likely to be used in practice.

B. Content-Based Recommendation Algorithms

Much research related to content based recommendations has been focussed on recommending items with associated textual content such as web pages and books etc. This problem has been treated as Information Retrieval task where a query describes the user's preference and on basis of similarity with this query, unrated documents are scored.

As content in text based system is usually described with *keywords,* the "importance" of word k_i in document d_j is determined with some *weighting* measure w_{ij} that can be defined in several different ways. One of the best-known measures for specifying keyword weights in Information Retrieval is the *term frequency/inverse document frequency (TF-IDF)* measure [16]. The *tf*idf,* weight, $w(t,d)$, of a term t in a document d is a function of the frequency of t in the document (tf_t,d), the number of documents that contain the term (df_t) and the number of documents in the collection (N). The intuition behind the weight is that the terms with the highest weight occur more often in that document than in the other documents, and therefore are more central to the topic of the document.

Recommendation can be treated as classification task as an alternative to information filtering. Using classification algorithms, the probability that whether a user will like an unseen item can be estimated which can be used to sort the list of recommendations. Also, such algorithms can provide prediction of numeric value such as degree of interest.

Some techniques for content-based recommendation includes Bayesian classifiers and various machine learning techniques, including clustering, decision trees and neural networks that are different from information retrieval based approaches because they are based on a *model* learned from the underlying data using statistical learning and machine learning techniques. Nearest neighbor algorithm is another popular technique to provide recommendations on the basis of textual information stored in memory (i.e. training data). To classify a new unlabelled item, the algorithm determines the nearest neighbor or k nearest neighbors using a similarity function (Euclidean or cosine similarity according to the type of textual information) and comparing it to all stored values. The class of the unseen item can then be determined from the class labels of nearest neighbors.

VI. HYBRID RECOMMENDATION SYSTEMS

Hybrid recommenders are systems that combine multiple recommendations techniques together to achieve a synergy between them. Several researchers have attempted to combine collaborative filtering and content based approaches in order to smoothen their disadvantages and gain better performance while recommendations. Depending on domain and data characteristics, several hybridization techniques are possible to combine CF and CB techniques which may generate different outputs. Some of the techniques are weighted [6], feature augmentation, feature combination, mixed, switching, cascade etc. Different ways of hybridization are:

- Implementing CF and CB separately and combine their predictions.
- Incorporating some content based characteristics into collaborative approach.
- Incorporating some collaborative characteristics into content based approach.
- Constructing a general unifying model that incorporates both content-based and collaborative characteristics.

[5] suggests that output generated from different approaches i.e. CF and CB can be combined using linear combination of ratings. In order to add CF characteristics in CB methods, dimensionality reduction techniques [9] can be applied to the content based profiles like Latent semantic analysis. However, dimensionality reduction can lead to loss of information.

Many hybrid approaches are based on CF but CB methods are used to maintain the user profiles and such profiles are used to find similar users. Good et al. (1999) use collaborative filtering along with a number of personalized information filtering agents. Boddu Raja Sarath Kumar et al. [12] has implemented content boosted collaborative filtering algorithm to consider both content based as well as collaborative filtering approach to alleviate sparsity problem. Michael J. Pazzani discusses an approach to combine recommendations from multiple sources viz., CF, CB and demographic information. CF and CB methods can also be combined under a single unifying model. Hydra: A Hybrid Recommender System [17] discusses the combination of CF and CB approaches in the context of web-based recommendations. This hybrid approach is special in that rating data as well as content information are joined in a unified model, which leads to less parameters and more reasonable prediction results.

So far hybrid approaches are the most successful approach still these techniques consider only the direct similarity between users or items. In such approaches, two users are similar only if they have shown similar interest for a common item, either by purchasing this item or by giving it a rating. Yet, two users can have similar preferences even though they have purchased different items. Likewise, two items that have not been purchased by the same user can still be similar. Christian Desrosiers and George Karypis [14] proposed a new collaborative filtering method using a global measure of similarity (Indirect Similarity) between users and items. Although they have not considered the impact of content based information in their experimental evaluation but how content based information can be used in the approach is suggested.

VII. PROBLEMS OF RECOMMENDATION SYSTEMS

Various techniques used in a recommender system experiences some of the hurdles that may be described in terms of basic problems as:

A. Sparsity Problem

Sparsity problem is one of the major problems encountered by recommender system and data sparsity has great influence on the quality of recommendation. Generally, data of system like MovieLens is represented in form of user-item matrix populated by ratings given to movies and as no. of users and items increases the matrix dimensions and sparsity evolves. The main reason behind data sparsity is that most users do not rate most of the items and the available ratings are usually sparse. Collaborative filtering suffers from this problem because it is dependent over the rating matrix in most cases. Many researchers [14], [20], [21] have attempted to alleviate this problem; still this area demands more research.

B. Cold Start problem

Cold start problem refers to the situation when a new user or item just enters the system. Three kinds of cold start problems are: new user problem, new item problem and new system problem. In such cases, it is really very difficult to provide recommendation as in case of new user, there is very less information about user that is available and also for a new item, no ratings are usually available and thus collaborative filtering cannot make useful recommendations in case of new item as well as new user. However, content based methods can provide recommendation in case of new item as they do not depends on any previous rating information of other users to recommend the item.

C. Scalability

Scalability is the property of system indicates its ability to handle growing amount of information in a graceful manner. With enormous growth in information over internet, it is obvious that the recommender systems are having an explosion of data and thus it is a great challenge to handle with continuously growing demand. Some of the recommender system algorithms deal with the computations which increase with growing number of users and items. In CF computations grow exponentially and get expensive, sometimes leading to inaccurate results. Methods proposed for handling this scalability problem and speeding up recommendation formulation are based on approximation mechanisms. Even if they improve performance, most of the time they result in accuracy reduction [24].

D. Over Specialization Problem

Users are restricted to getting recommendations which resemble to those already known or defined in their profiles [21] in some cases and it is termed as over specialization problem. It prevents user from discovering new items and other available options. However, diversity of recommendations is a desirable feature of all recommendation system. After solving the problem using genetic algorithms, user will be provided with a set of different and a wide range of alternatives.

VIII. CONCLUSIONS

Several recommendation systems have been proposed that are based on collaborative filtering, content based filtering and hybrid recommendation methods and so far most of them have been able to solve the problems while providing better recommendations. However, due to information explosion, it is required to work on this research area to explore and provide new methods that can provide recommendation in a wide range of applications while considering the quality and privacy aspects. Thus, the current recommendation system needs improvement for present and future requirements of better recommendation qualities.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions", IEEETKDE: IEEE Transactions on Knowledge and Data Engineering, 17, 2005.

[2] Prodan Andrei-Cristian, "Implementation of a Recommender System Using Collaborative Filtering", Studia univ. Babes_bolyai, Informatica, volume lv, number 4, 2010.

[3] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms", In WWW '01: Proceedings of the 10th international conference on World Wide Web, New York, NY, USA, 2001.

[4] Michael J. Pazzani and Daniel Billsus, "Content Based Recommendation System."

[5] Claypool, M., A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper". In ACM SIGIR'99, Workshop on Recommender Systems: Algorithms and Evaluation, August 1999.

[6] Debnath, Ganguly and Mitra, "Feature Weighting In Content Based Recommendation System Using Social Network Analysis".

[7] MovieLens Dataset, USA, 2003, University Minnesota.

[8] Balabanovic, M. and Y. Shoham. Fab, "Content-based, collaborative recommendation", Communications of the ACM, 40(3):66-72, 1997.

[9] Sarwar, Badrul M., George Karypis, Joseph A. Konstan and John T. Riedl, "Application of Dimensionality Reduction in Recommender System – A Case Study", In ACM WebKDD Workshop, 2000.

[10] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste, "A constant time collaborative filtering algorithm", Information Retrieval, 4(2):133–151, 2001.

[11] Oznur Kirmemis, "OPENMORE: A Content-based Movie Recommendation System", Master Thesis, Middle East Technical University, Department of Computer Engineering, May 2008.

[12] Boddu Raja Sarath Kumarmaddali and Surendra Prasad Babuan:, "Implementation of Content Boosted Collaborative Filtering Algorithm", IJEST.

[13] J. Wang, A. P. de Vries, and M. J. T. Reinders, "Unifying User-Based and Item-Based Collaborative Filtering Approaches by Similarity Fusion". In SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, 2006.

[14] C. Desrosiers and G. Karypis, "Solving the Sparsity Problem: Collaborative Filtering via Indirect Similarities", Technical Report, Dec 2008.

[15] Terveen and Hill, "Beyond Recommendation System in HCI in the New Millennium, Addison-Wesley, 2001 p. 4 of 21.

[16] Salton, G., "Automatic Text Processing", Addison-Wesley, 1989.

[17] Stephan Spiegel, Jérôme Kunegis and Fang Li, "Hydra: A Hybrid Recommender System [Cross-Linked Rating and Content Information]".

[18] Basu, C., Hirsh, H., Cohen W., "Recommendation as classification: Using Social and Content-Based Information in Recommendation". In: Proceedings of the 15th National Conference on Artificial Intelligence, Madison, WI (1998) 714-720.

[19] Belkin, N., Croft, B., "Information Filtering and Information Retrieval: Two Sides of the Same Coin?" Communications of the ACM 35(12) (1992).

[20] Zhou and T.Luo, "A Novel Approach to Solve the Sparsity Problem in Collaborative Filtering."

[21] Yibo Chen, chanle Wu, Ming Xie and Xiaojun Guo, "Solving the Sparsity Problem in Recommender Systems Using Association Retrieval", Journal of Computers, Vol. 6, No. 9, September 2011.

[22] Yiwen Wang, Natalia Stash, Lora Aroyo, Laura Hollink, Guus Schreiber, "Semantic Relations in Content-based Recommender Systems".

[23] Scalability, http://en.wikipedia.org/wiki/Scalability.

[24] Manos Papagelis, Ioannis Rousidis, Dimitris Plexousakis, Elias Theoharopoulos, "Incremental Collaborative Filtering for Highly-Scalable Recommendation Algorithms", ISMIS 2005, LNAI 3488, pp. 553-561, 2005.