Comparative Study on filtering techniques for Recommendation Systems

Sanjana De
Computer Science dept
St. Xavier's College
Kolkata, India
sanjanade12345@gmail.com

***Abstract*—The ongoing rapid expansion of the Internet greatly increases the necessity of effective recommendation systems (RS) for filtering the abundant information. These systems search through large volumes of dynamically generated information to provide users with personalized content and services. In this paper, a comparative study of the different techniques used for the filter process is discussed, mainly the commonly used collaborative filtering, content-based filtering and hybrid filtering are included.**

Keywords—filtering, collaborative, content-based, hybrid, recommendation systems

1. INTRODUCTION

The explosive growth in the amount of available digital information and the number of visitors to the Internet have created a potential challenge of information overload which hinders timely access to items of interest on the Internet. This increases the demand for recommendation systems more than ever. Recommendation systems are information filtering systems that deal with the problem of information overload by filtering vital information fragments out of a large amount of dynamically generated information according to the user's preferences, interest or observed behavior about the item. It also predicts whether a user would prefer an item or not based on the user's profile[1]. The development of recommendation systems is a multi-disciplinary effort which involves experts from various fields like Artificial Intelligence, Human Computer Interaction, Information Technology, Data Mining, Statistics, Adaptive User Interfaces, Decision Support Systems, Marketing or Consumer Behavior[2]. Apart from collaborative, content-based and hybrid filtering techniques, the recommendation systems may also adopt techniques like knowledge-based, demographic or community-based filtering techniques.

2. RELATED WORK

The recommendation system was defined as a means of assisting and augmenting the social process of using recommendations of others to make choices when there is no sufficient personal knowledge or experience of the alternatives. Various approaches of building recommendation systems have been developed, which can utilize either collaborative filtering, content-based filtering or hybrid filtering. Collaborative filtering(CF) recommends items by identifying other users with similar taste. It uses their opinion to recommend items to the active user. Amazon uses topic diversification algorithms to improve its recommendation[3]. Their system uses collaborative filtering to overcome scalability issue by generating a table of similar items offline through the use of item-to-item matrix. The system then recommends other products which are similar online according to the users' purchase history. GroupLens[4] is another CF system that is based on client/server architecture. The developers of one of the first recommender systems, Tapestry[5] coined the phrase "collaborative filtering (CF)," which has been widely adopted regardless of the facts that recommenders may not explicitly collaborate with recipients and recommendations may suggest particularly interesting items, in addition to indicating those that should be filtered out. Content based filtering methods are based on a description of the item and a profile of the user's preferences. These methods are best suited to situations where there is known data on an item but not on the user. Pazzani et al. [6] designed an intelligent agent using content based approach that attempts to predict which web pages will interest a user by using naive Bayesian classifier. The agent allows a user to provide training instances by rating different pages as either hot or cold. Jennings and Higuchi [7] describe a neural network that models the interests of a user in a Usenet news environment. A hybrid recommender algorithm is employed by many applications as a result of new user problem of content-based filtering techniques and average user problem of collaborative filtering [8]. A simple and straightforward method for combining content-based and collaborative filtering was proposed by Cunningham et al. [9]. A music recommendation system which combined tagging information, play counts and social relations was proposed in Konstas et al. [10]. In order to determine the number of neighbors that can be automatically connected on a social platform, Lee and Brusilovsky[11] embedded social information into collaborative filtering algorithm.

3. FILTERING TECHNIQUES

3.1 Collaborative filtering

Collaborative filtering technique is the simplest and original implementation that recommends to the active user the items that other users with similar tastes liked in the past. The similarity in taste of two users is calculated based on the similarity in the rating history of the users. Collaborative filtering methods are classified as memory-based and model-based. The memory-based method can be further divided into item-based and user-based methods whereas the model-based technique includes clustering techniques, association techniques, neural networks, Bayesian networks etc. the breakdown is shown in Figure (1) below[1]. The fundamental assumption of Collaborative Filtering(CF) method is that if users X and Y rate *n* items similarly, or have similar behaviours (e.g., buying, watching, listening), and hence will rate or act on other items similarly. CF techniques use a database of preferences for items by users to predict additional topics or products a new user might like. In a typical CF scenario, there is a list of *m* users U= $\{u_1, u_2, \ldots,$

u_m} and a list of *n* items I={i_1, i_2, ... , i_n}, and each user, u_i, has a list of items, I_{ui}, which the user has rated, or about which their preferences have been inferred through their behaviours. The ratings can either be explicit indications, and so forth, on a 1–5 scale, or implicit indications, such as purchases or click-throughs. For example, we can convert the list of people and the movies they like or dislike to a user-item ratings matrix. There might be missing values in the matrix where users did not give their preferences for certain items. There exists a distinguished user $u_a \in U$ called the active user for whom the task of a collaborative filtering algorithm is to find an item likeliness that can be of two forms :

- Prediction: a numerical value, $P_{a,j}$, expressing the predicted likeliness of item $i_j \notin I_{ua}$ for the active user u_a. This predicted value is within the same scale (e.g., from 1 to 5) as the opinion values provided by u_a.
- Recommendation: a list of N items, $I_r \subset I$, that the active user will like the most. Note that the recommended list must be on items not already purchased by the active user, i.e., $I_r \cap I_{ua} = \phi$. This interface of CF algorithms is also known as Top-N recommendation[13,19].

CF algorithms represent the entire *m x n* user-item data as a ratings matrix, *A*. Each entry $a_{i,j}$ in *A* represents the preference score (ratings) of the *i*th user on the *j*th item. Each individual rating is within a numerical scale and it can as well be 0 indicating that the user has not yet rated that item. CF algorithms get further divided into: Memory-based (or neighborhood-based) and Model-based algorithms[14].

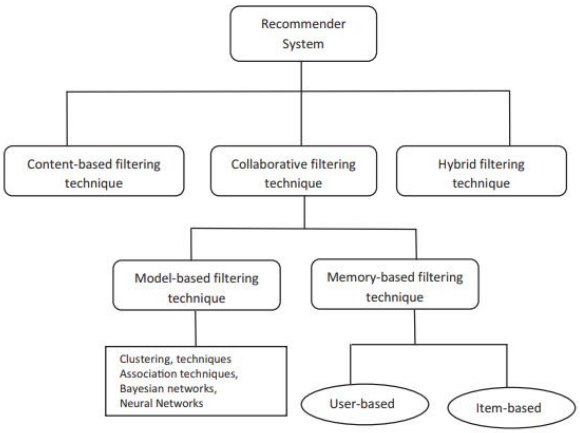

Figure :– (1)

1) Memory-based CF techniques

The items that were already rated by the user before play a relevant role in searching for a neighbor that shares appreciation with him. Once a neighbor of a user is found, different algorithms can be used to combine the preferences of neighbors to generate recommendations. Due to the effectiveness of these techniques, they have achieved widespread success in real life applications. Memory-based CF can be achieved in two ways through user-based and item-based techniques.

a) User-based CF techniques

User-based systems[24], such as GroupLens, Bellcore video, and Ringo, evaluate the interest of a user u for an item *i* using the ratings for this item by other users, called neighbors, that have similar rating patterns. The neighbors of user *u* are typically the users *v* whose ratings on the items rated by both *u* and *v*, i.e. I_{uv}, are most correlated to those of u. User-based neighborhood recommendation methods predict the rating r_{ui} of a user u for a new item *i* using the ratings given to *i* by users most similar to *u*, called nearest-neighbors[15]. Suppose we have for each user $v \neq u$ a value w_{uv} representing the preference similarity between *u* and *v*. The *k*-nearest-neighbors (*k-NN*) of *u*, denoted by *N(u)*, are the *k* users, *v* with the highest similarity w_{uv} to *u*. However, only the users who have rated item *i* can be used in the prediction of r_{ui}, and we instead consider the *k* users most similar to u that have rated *i*. We write this set of neighbors as $N_i(u)$. The rating r_{ui} can be estimated as the average rating given to *i* by these neighbors:

$$\hat{r}_{ui} = \frac{1}{|\mathcal{N}_i(u)|} \sum_{v \in \mathcal{N}_i(u)} r_{vi}. \quad (1)$$

But it does not take into account the fact that the neighbors can have different levels of similarity. A common solution to this problem is to weigh the contribution of each neighbor by its similarity to *u*. The prediction approach just described, where the predicted ratings are computed as a weighted average of the neighbors' ratings, essentially solves a regression problem.[20]

b) Item-based CF Technique

Item-based approaches predict the rating of a user u for an item *i* based on the ratings of *u* for items similar to *i*. In such approaches, two items are similar if several users of the system have rated these items in a similar fashion. While user-based methods rely on the opinion of like-minded users to predict a rating, item-based approaches look at ratings given to similar items[16]. This idea can be formalized as follows. Denote by $N_u(i)$ the items rated by user *u* most similar to item *i*. The predicted rating of *u* for *i* is obtained as a weighted average of the ratings given by *u* to the items of $N_u(i)$:

$$\hat{r}_{ui} = \frac{\sum_{j \in \mathcal{N}_u(i)} w_{ij} r_{uj}}{\sum_{j \in \mathcal{N}_u(i)} |w_{ij}|}. \quad (3)$$

Item-based filtering techniques compute predictions using the similarity between items and not the similarity between users. It builds a model of item similarities by retrieving all items rated by an active user from the user-item matrix, it determines how similar the retrieved items are to the target item, then it selects the k most similar items and their corresponding similarities are also determined[16]. Prediction is made by taking a weighted average of the active users rating on the similar items *k*. Several types of similarity measures are used to compute similarity between item/user.

The two most popular similarity measures are correlation-based and cosine-based.

- Pearson correlation coefficient is used to measure the extent to which two variables linearly relate with each other and is defined as :

$$s(a,u) = \frac{\sum_{i=1}^{n}(r_{a,i} - \overline{r_a})(r_{u,i} - \overline{r_u})}{\sqrt{\sum_{i=1}^{n}(r_{a,i} - \overline{r_a})^2}\sqrt{\sum_{i=1}^{n}(r_{u,i} - \overline{r_u})^2}} \quad (4)$$

From the above equation, *s(a,u)* denotes the similarity between two users *a* and *u*, $r_{a,i}$ is the rating given to item *i* by user *a*, r_a is the mean rating given by user *a* while *n* is the total number of items in the user-item space. Also, prediction for an item is made from the weighted combination of the selected neighbors' ratings, which is computed as the weighted deviation from the neighbors' mean[17]. The general prediction formula is :

$$p(a,i) = \overline{r_a} + \frac{\sum_{i=1}^{n}(r_{u,i} - \overline{r_u}) \times s(a,u)}{\sum_{i-1}^{n} s(a,u)} \quad (5)$$

Other correlation-based similarities include: constrained Pearson correlation, a variation of Pearson correlation that uses midpoint instead of mean rate; Spearman rank correlation, similar to Pearson correlation, except that the ratings are ranks; and Kendall's τ correlation, similar to the Spearman rank correlation, but instead of using ranks themselves, only the relative ranks are used to calculate the correlation.

- Cosine similarity is different from Pearson-based measure in that it is a vector-space model which is based on linear algebra rather than statistical approach. It measures the similarity between two n-dimensional vectors based on the angle between them[18]. Cosine-based measure is widely used in the fields of information retrieval and text mining to compare two text documents, in this case, documents are represented as vectors of terms. The similarity between two items u and v can be defined as follows:

$$s(\vec{u},\vec{v}) = \frac{\vec{u}\cdot\vec{v}}{|\vec{u}| * |\vec{v}|} = \frac{\sum_i r_{u,i} r_{v,i}}{\sqrt{\sum_i r_{u,i}^2} \times \sqrt{\sum_i r_{v,i}^2}} \quad (6)$$

Similarity measures are also referred to as similarity metric, and they are methods used to calculate the scores that express how similar users or items are to each other. These scores can then be used as the foundation of user- or item-based recommendation generation. Depending on the context of use, similarity metrics can also be referred to as correlation metrics or distance metrics[22][23].

2) Model-based CF techniques

This technique employs the previous ratings to learn a model in order to improve the performance of Collaborative filtering Technique. The model building process can be done using machine learning or data mining techniques. These techniques can quickly recommend a set of items for the fact that they use pre-computed model and they have proved to produce recommendation results that are similar to neighborhood-based recommender techniques. Examples of these techniques include Dimensionality Reduction technique such as Singular Value Decomposition (SVD), Matrix Completion Technique[25], Latent Semantic methods, and Regression and Clustering. Model-based techniques analyze the user-item matrix to identify relations between items; they use these relations to compare the list of top-N recommendations. Model based techniques resolve the sparsity problems associated with recommendation systems.

a) Clustering

Clustering algorithm tries to partition a set of data into a set of sub-clusters in order to discover meaningful groups that exist within them [27]. Once clusters have been formed, the opinions of other users in a cluster can be averaged and used to make recommendations for individual users. A good clustering method will produce high quality clusters in which the intra-cluster similarity is high, while the inter-cluster similarity is low. The measurement of the similarity between objects is determined using metrics such as Minkowski distance and Pearson correlation. For two data objects, $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$, the popular Minkowski distance is defined as :

$$d(X,Y) = \sqrt[q]{\sum_{i=1}^{n} |x_i - y_i|^q}, \quad (7)$$

where n is the dimension number of the object and x_i, y_i are the values of the *i*th dimension of object X and Y respectively, and *q* is a positive integer. When *q = 1*, *d* is Manhattan distance; when *q = 2*, *d* is Euclidian distance[28]. Clustering methods can be classified into three categories: partitioning methods, density-based methods, and hierarchical methods[29]. In most situations, clustering is an intermediate step and the resulting clusters are used for further analysis or processing to conduct classification or other tasks. Clustering CF models can be applied in different ways. Sarwar et al. [29] and O'Connor and Herlocker [30] use clustering techniques to partition the data into clusters and use a memory-based CF algorithm such as a Pearson correlation-based algorithm to make predictions for CF tasks within each cluster. Clustering models have better scalability than typical collaborative filtering methods because they make predictions within much smaller clusters rather than the entire customer base. The complex and expensive clustering computation is run offline. However, its recommendation quality is generally low. It is possible to improve quality by using numerous fine-grained segments, but then online user-segment classification becomes almost as expensive as finding similar customers using memory based collaborative filtering. As optimal clustering over large data sets is impractical, most applications use various forms of greedy cluster generation techniques[31].

b) Regression

Regression analysis is used when two or more variables are thought to be systematically connected by a linear relationship. For memory-based CF algorithms, in some cases, two rating vectors may be distant in terms of Euclidean distances but they have very high similarity using vector cosine or Pearson correlation measures, where memory-based CF algorithms do not fit well and need better solutions. Also, numerical ratings are common in real-life recommender systems. Regression methods that are good at making predictions for numerical values are helpful to address these problems. A regression method uses an approximation of the ratings to make predictions based on a regression model. Let $X = (X_1, X_2, \ldots, X_n)$ be a random variable representing a user's preferences on different items. The linear regression model can be expressed as :

$$Y = \Lambda X + N , \quad (8)$$

where Λ is a $n \times k$ matrix. $N = (N_1, \ldots, N_n)$ is a random variable representing noise in user choices, Y is an $n \times m$ matrix with Y_{ij} is the rating of user i on item j, and X is a $k \times m$ matrix with each column as an estimate of the value of the random variable X (user's ratings in the k-dimensional rating space) for one user. Typically, the matrix Y is very sparse[34].

It is a powerful and diversity process to analyse associative relationships between dependent variable and one or more independent variables. Uses of regression contain curve fitting, prediction, and testing systematic hypotheses about relationships between variables. The curve can be useful to identify a trend within dataset, whether it is linear, parabolic, or of some other forms.

c) Baesyian Classifiers

They are a probabilistic framework for solving classification problems which is based on the definition of conditional probability and Bayes theorem. Bayesian classifiers [35] consider each attribute and class label as random variables. Given a record of N features ($A_1, A_2, \ldots, A_N$), the goal of the classifier is to predict class C_k by finding the value of C_k that maximizes the posterior probability of the class given the data $P(C_k | A_1, A_2, \ldots, A_N)$ by applying Bayes' theorem, $P(C_k|A_1, A_2, \ldots, A_N)\ P(A_1, A_2, \ldots, A_N|C_k)P(C_k)$. The most commonly used Bayesian classifier is known as the Naive Bayes Classifier. In order to estimate the conditional probability, $P(A_1, A_2, \ldots, A_N|C_k)$, a Naive Bayes Classifier assumes the probabilistic independence of the attributes that is, the presence or absence of a particular attribute is unrelated to the presence or absence of any other. This assumption leads to $P(A_1, A_2, \ldots, A_N|C_k) = P(A_1|C_k)P(A_2|C_k)\ldots P(A_N|C_k)$. The main benefits of Naive Bayes classifiers are that they are robust to isolated noise points and irrelevant attributes, and they handle missing values by ignoring the instance during probability estimate calculations[36]. However, the independence assumption may not hold for some attributes as they might be correlated. In this case, the usual approach is to use Bayesian Networks.

d) Matrix completion techniques

The essence of matrix completion technique is to predict the unknown values within the user-item matrices. Correlation based K-nearest neighbor is one of the major techniques employed in collaborative filtering recommendation systems[37]. They depend largely on the historical rating data of users on items. Most of the time, the rating matrix is always very big and sparse due to the fact that users do not rate most of the items represented within the matrix[38]. This problem always leads to the inability of the system to give reliable and accurate recommendations to users. Different variations of low rank models have been used in practice for matrix completion especially toward application in collaborative filtering[39]. Formally, the task of matrix completion technique is to estimate the entries of a matrix, $M \in R^{m \times n}$, when a subset, $\Omega\ C\{(i, j) : 1 \leq i \leq m,\ 1 \leq j \leq n\}$ of the new entries is observed, a particular set of low rank matrices, $M_b = UV^T$, where $U \in R^{m \times k}$ and $V \in R^{m \times k}$ and $k \ll \min(m, n)$. The most widely used algorithm in practice for recovering M from partially observed matrix using low rank assumption is Alternating Least Square (ALS) minimization which involves optimizing over U and V in an alternating manner to minimize the square error over observed entries while keeping other factors fixed.

e) Latent semantic CF models

A Latent semantic CF technique relies on a statistical modeling technique that introduces latent class variables in a mixture model setting to discover user communities and prototypical interest profiles. Conceptionally, it decomposes user preferences using overlapping user communities. The main advantages of this technique over standard memory-based methods are its higher accuracy and scalability[40,41].

3.1.1 Advantages of CF techniques

Collaborative Filtering has some major advantages over CBF in that it can perform in domains where there is not much content associated with items and where content is difficult for a computer system to analyze (such as opinions and ideal). Also, CF technique has the ability to provide serendipitous recommendations, which means that it can recommend items that are relevant to the user even without the content being in the user's profile [42].

A key advantage of this technique is that as it doesn't rely on machine analyzable content, it therefore is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself.

3.1.2 Disadvantages of CF filtering

- *Cold start problem*: This refers to a situation where a recommender does not have adequate information about a user or an item in order to make relevant predictions [43]. This is one of the major problems

that reduce the performance of recommendation system. The profile of such new user or item will be empty since he has not rated any item; hence, his taste is not known to the system.

- *Data sparsity:* This is the problem that occurs as a result of lack of enough information, that is, when only a few of the total number of items available in a database are rated by users. This always leads to a sparse user-item matrix, inability to locate successful neighbors and finally, the generation of weak recommendations[45]. Also, data sparsity always leads to coverage problems, which is the percentage of items in the system that recommendations can be made for.
- *Scalability:* When numbers of existing users and items grow tremendously, traditional CF algorithms will suffer serious scalability problems, with computational resources going beyond practical or acceptable levels. For example, with tens of millions of customers (M) and millions of distinct catalog items (N), a CF algorithm with the complexity of O(n) is already too large. As well, many systems need to react immediately to online requirements and make recommendations for all users regardless of their purchases and ratings history, which demands a high scalability of a CF system[45].

3.2 Content Based Filtering Technique

Content-based recommendation systems try to recommend items similar to those a given user has liked in the past. The basic process performed by a content-based recommender consists in matching up the attributes of a user profile in which preferences and interests are stored, with the attributes of a content object (item), in order to recommend to the user new interesting items.

Systems implementing a content-based recommendation approach analyze a set of documents and/or descriptions of items previously rated by a user, and build a model or profile of user interests based on the features of the objects rated by that user [47]. The profile is a structured representation of user interests, adopted to recommend new interesting items.

Content-based filtering exploits the content of data items to predict its relevance based on the user's profile. Research on content-based recommender systems takes place at the intersection of many computer science topics, especially Information Retrieval [48] and Artificial Intelligence.

In most content-based filtering systems, item descriptions are textual features extracted from Web pages, emails, news articles or product descriptions. Unlike structured data, there are no attributes with well-defined values. Textual features create a number of complications when learning a user profile, due to the natural language ambiguity.

Semantic analysis and its integration in personalization models is an approach proposed in literature to solve those problems. The key idea is the adoption of knowledge bases, such as lexicons or ontologies , for annotating items and representing profiles in order to obtain a "semantic" interpretation of the user information needs.

It could use Vector Space Model such as Term Frequency Inverse Document Frequency (TF/IDF) or Probabilistic models such as Naı¨ve Bayes Classifier [35], Decision Trees [50] or Neural Networks [49] to model the relationship between different documents within a corpus. These techniques make recommendations by learning the underlying model with either statistical analysis or machine learning techniques.

1) Keyword-based Vector Spaced Model(VSM)

VSM is a spatial representation of text documents. In that model, each document is represented by a vector in a n-dimensional space, where each dimension corresponds to a term from the overall vocabulary of a given document collection.

Formally, every document is represented as a vector of term weights, where each weight indicates the degree of association between the document and the term. Let D = $\{d_1,d_2,...,d_N\}$ denote a set of documents or corpus, and T = $\{t_1,t_2,...,t_n\}$ be the dictionary, that is to say the set of words in the corpus. T is obtained by applying some standard natural language processing operations, such as tokenization, stopwords removal, and stemming [52]. Each document d_j is represented as a vector in a n-dimensional vector space, so $d_j = \{w_{1\,j}, w_{2\,j}, ..., d_{n\,j}\}$, where $w_{k\,j}$ is the weight for term t_k in document dj .

Document representation in the VSM raises two issues: weighting the terms and measuring the feature vector similarity. The most commonly used term weighting scheme, TF-IDF (Term Frequency-Inverse Document Frequency) weighting, is based on empirical observations regarding text [51]:

- rare terms are not less relevant than frequent terms (IDF assumption);
- multiple occurrences of a term in a document are not less relevant than single occurrences (TF assumption);
- long documents are not preferred to short documents (normalization assumption).

These assumptions are well exemplified by the TF-IDF function:

$$\text{TF-IDF}(t_k, d_j) = \underbrace{\text{TF}(t_k, d_j)}_{\text{TF}} \cdot \underbrace{log\frac{N}{n_k}}_{\text{IDF}} \tag{11}$$

where N denotes the number of documents in the corpus, and n_k denotes the number of documents in the collection in which the term tk occurs at least once.

$$\text{TF}(t_k, d_j) = \frac{f_{k,j}}{max_z f_{z,j}} \tag{12}$$

where the maximum is computed over the frequencies f_z, j of all terms t_z that occur in document d_j. In order for the weights to fall in the [0,1] interval and for the documents to be represented by vectors of equal length, weights obtained by Equation (11) are usually normalized by cosine normalization:

$$w_{k,j} = \frac{\text{TF-IDF}(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|} \text{TF-IDF}(t_s, d_j)^2}} \quad (13)$$

In content-based recommender systems relying on VSM, both user profiles and items are represented as weighted term vectors. Predictions of a user's interest in a particular item can be derived by computing the cosine similarity.

In the vector space model user profiles can be represented just like documents by one or more profile vectors. The degree of similarity between a profile vector P, where P =(u_1 ,..., u_k) can be determined by using the cosine measure:

$$sim(D, P) = \frac{D \cdot P}{\|D\| \cdot \|P\|} = \frac{\sum_k u_k \cdot w_k}{\sqrt{\sum_k u_k^2 \cdot \sum_k w_k^2}} \quad (14)$$

3.2.1 Advantages of CBF techniques

CB filtering techniques overcome the challenges of CF.

- *New item:* They have the ability to recommend new items even if there are no ratings provided by users. So even if the database does not contain user preferences, recommendation accuracy is not affected.
- *User independence:* if the user preferences change, it has the capacity to adjust its recommendations in a short span of time. They can manage situations where different users do not share the same items, but only identical items according to their intrinsic features. Users can get recommendations without sharing their profile, and this ensures privacy.
- *Transparency:* Explanations on how the recommender system works can be provided by explicitly listing content features or descriptions that caused an item to occur in the list of recommendations. Those features are indicators to consult in order to decide whether to trust a recommendation. Conversely, collaborative systems are black boxes since the only explanation for an item recommendation is that unknown users with similar tastes liked that item[1].

3.2.2 Disadvantages of CBF techniques

The problems associated with content-based filtering techniques are limited content analysis, overspecialization and sparsity of data.

- *Limited content analysis*: Content based filtering techniques are dependent on items' metadata. That is, they require rich description of items and very well-organized user profile before recommendation can be made to users. So, the effectiveness of CBF depends on the availability of descriptive data.
- *Content overspecialization*: Content-based recommenders have no inherent method for finding something unexpected. The system suggests items whose scores are high when matched against the user profile, hence the user is going to be recommended items similar to those already rated. This drawback is also called serendipity problem to highlight the tendency of the content-based systems to produce recommendations with a limited degree of novelty.
- *New user:* Enough ratings have to be collected before a content-based recommender system can really understand user preferences and provide accurate recommendations. Therefore, when few ratings are available, as for a new user, the system will not be able to provide reliable recommendations[1,55].

3.3 Hybrid Filtering Techniques

Hybrid filtering technique combines different recommendation techniques in order to gain better system optimization to avoid some limitations and problems of pure recommendation systems. The idea behind hybrid techniques is that a combination of algorithms will provide more accurate and effective recommendations than a single algorithm as the disadvantages of one algorithm can be overcome by another algorithm. Using multiple recommendation techniques can suppress the weaknesses of an individual technique in a combined model. The combination of approaches can be done in any of the following ways: separate implementation of algorithms and combining the result, utilizing some content-based filtering in collaborative approach, utilizing some collaborative filtering in content-based approach, creating a unified recommendation system that brings together both approaches.

This vision for hybridization was refined by Burke[59] and then by Adomavicius and Tuzhilin. Burke made a list of the following seven hybridization techniques:

- *Weighted:* the recommendation value of an item is based on the sum of available methods. For example, the system P-Tango gives an equal value to both collaborative filtering and content-based filtering. This value is then weighted by a confirmation of the users.
- *Switching:* the system chooses to apply either a data-based method or social filtering depending on the search context of the user.
- *Mixed:* this technology facilitates the proposal of recommendations from traditional methods with the aim of limiting the drawbacks of each classic method.
- *Features Combination:* this method offers the possibility of enriching data which has been integrated a priori into the system with the ratings of users, which enriches the database a posteriori . The computation of the recommendation is carried out over all of the data.
- *Cascade:* this process consists of a double analysis of user profiles. The first is used to highlight potential candidates, the second to refine the selection of users.
- *Features Augmentation:* this is a technique which is similar to the previous one for the first pass-through. If the number of candidates is too high on the first pass-through, then a second will carry out a

secondary discrimination by integrating the data of recommended items.

- *Meta level:* as for the first two methods, it involves filtering users twice in order to determine similarities. The difference is that the first pass-through makes possible the generation of a model or profile type of the user.[59]

II. CONCLUSION

Most recommender systems nowadays use a hybrid approach, combining collaborative filtering, content-based filtering, and other approaches. The paper discusses the strengths and weaknesses of collaborative filtering and content based filtering, which can be overcomed by the implementation of mix and match of these techniques giving rise to hybrid filtering techniques. Also, this knowledge of various algorithms can serve as a road map to either improve on the existing algorithms or to build a recommender system for researchers. From an e-commerce perspective, RS is used as a tool that helps users search through records of knowledge which is related to users' interest and preference.

ACKNOWLEDGMENT

This work was supported by the post-graduate department of Computer Science at St. Xavier's College, Kolkata. As an author, I am grateful to professors of the department for giving me the chance to go ahead with this topic for the paper.

REFERENCES

[1] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, Recommendation systems: Principles, methods and evaluation, Egyptian Informatics Journal, Volume 16, Issue 3,2015,Pages 261-273,ISSN 1110-8665,https://doi.org/10.1016/j.eij.2015.06.005.

[2] Francesco Ricci and Lior Rokach and Bracha Shapira, Introduction to Recommender Systems Handbook, Recommender Systems Handbook, Springer, 2011, pp. 1-35

[3] Ziegler CN, McNee SM, Konstan JA, Lausen G. Improving recommendation lists through topic diversification. In: Proceedings of the 14th international conference on World Wide Web; 2005. p. 22–32.

[4] Resnick, Paul, Neophytos Iacovou, Mitesh Suchak, Peter Bergström, and John Riedl. "GroupLens: an open architecture for collaborative filtering of netnews." In Proceedings of the 1994 ACM conference on Computer supported cooperative work, pp. 175-186. ACM, 1994.

[5] Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM. December.

[6] Pazzani MJ. A framework for collaborative, content-based and demographic filtering. Artific Intell Rev 1999;13:393–408, No. 5(6).

[7] Jennings A, Higuchi H. A personal news service based on a user model neural network. IEICE Trans Inform Syst 1992;E75-D(2):198–209.

[8] Burke R. Hybrid web recommender systems. In: Brusilovsky P, Kobsa A, Nejdl W, editors. The adaptive web, LNCS 4321. Berlin Heidelberg, Germany: Springer; 2007. p. 377–408. http:// dx.doi.org/10.1007/978-3-540-72079-9_12.

[9] Cunningham P, Bergmann R, Schmitt S, Traphoner R, Breen S, Smyth B. ''WebSell: Intelligent sales assistants for the World Wide Web. In: Proceedings CBR in ECommerce, Vancouver BC; 2001. p. 104–9.

[10] Konstan I, Stathopoulos V, Jose JM. On social networks and collaborative recommendation. In: The proceedings of the 32nd international ACM conference (SIGIR'09), ACM. New York, NY, USA; 2009. P.195–202.

[11] Lee DH, Brusilovsky P. Social networks and interest similarity: the case of CiteULike. In: Proceedings of the 21st ACM conference on Hypertext and Hypermedia (HT'10). ACM. New York, NY, USA; 2010. p. 151–6.

[12] Herlocker JL, Konstan JA, Terveen LG, Riedl JT. Evaluating collaborative filtering recommender systems. ACM Trans Inform Syst 2004;22(1):5–53.

[13] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-Based Collaborative Filtering Recommendation Algorithms. GroupLens Research Group/Army HPC Research Center Department of Computer Science and Engineering University of Minnesota, Minneapolis, MN 55455

[14] Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43-52.

[15] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," in IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 734-749, June 2005, doi: 10.1109/TKDE.2005.99.

[16] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item based collaborative filtering recommendation algorithms," in Proceedings of the 10th International Conference on World Wide Web (WWW '01), pp. 285–295, May 2001.

[17] M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and evaluation metric that accurately model the user experience," in Proceedings of 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04), pp. 329–336, Sheffield, UK, 2004.

[18] Aggarwal, C. C., Wolf, J. L., Wu, K., and Yu, P. S. (1999). Horting Hatches an Egg: A New Graph-theoretic Approach to Collaborative Filtering. In Proceedings of the ACM KDD'99 Conference. San Diego, CA. pp. 201-212.

[19] G. Salton and M. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, New York, NY, USA, 1983.

[20] G. Karypis, "Evaluation of item-based top-N recommendation algorithms," in Proceedings of the International Conference on Information and Knowledge Management (CIKM '01), pp. 247–254, Atlanta, Ga, USA, November 2001.

[21] K. Yu, X. Xu, J. Tao, M. Ester, and H.-P. Kriegel, "Instance selection techniques for memory-based collaborative filtering," in Proceedings of the SIAM International Conference on Data Mining (SDM '02), April 2002.

[22] Acilar AM, Arslan A. A collaborative filtering method based on Artificial Immune Network. Exp Syst Appl 2009;36(4):8324–32.

[23] Adomavicius G, Tuzhilin A. Toward the next generation of recommender system. A survey of the state-of-the-art and possible extensions. IEEE Trans Knowl Data Eng 2005;17(6):734–49.

[24] Herlocker JL, Konstan JA, Terveen LG, Riedl JT. Evaluating collaborative filtering recommender systems. ACM Trans Inform Syst 2004;22(1):5–53.

[25] Zhao ZD, Shang MS. User-based collaborative filtering recommendation algorithms on Hadoop. In: Proceedings of 3rd international conference on knowledge discovering and data mining, (WKDD 2010), IEEE Computer Society, Washington DC, USA; 2010. p. 478–81

[26] Keshavan RH, Montanari A, Sewoong O. Matrix completion from a few entries. IEEE Trans Inform Theor 2010;56(6):2980–98.

[27] Bojnordi E, Moradi P. A novel collaborative filtering model based on combination of correlation method with matrix completion technique. In: 16th CSI international symposium on artificial intelligence and signal processing (AISP), IEEE; 2012.

[28] Kuzelewska U. Advantages of information granulation in clus- _ tering al-gorithms. In: Agents and artificial intelligence. NY: Springer; 2013. p. 131–45.

[29] X. Su, M. Kubat, M. A. Tapia, and C. Hu, "Query size estimation using clustering techniques," in Proceedings of the 17th International

Conference on Tools with Artificial Intelligence (ICTAI '05), pp. 185–189, Hong Kong, November 2005.

[30] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Recommender systems for large-scale E-commerce: scalable neighborhood formation using clustering," in Proceedings of the 5th International Conference on Computer and Information Technology (ICCIT '02), December 2002.

[31] M. O'Connor and J. Herlocker, "Clustering items for collaborative filtering," in Proceedings of the ACM SIGIR Workshop on Recommender Systems (SIGIR '99), 1999.

[32] J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, Calif, USA, 2001. .

[33] D. Billsus and M. Pazzani, "Learning collaborative information filters," in Proceedings of the 15th International Conference on Machine Learning (ICML '98), 1998.

[34] J. Canny, "Collaborative filtering with privacy via factor analysis," in Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 238–245, Tampere, Finland, August 2002

[35] S. Vucetic and Z. Obradovic, "Collaborative filtering using a regression-based approach," Knowledge and Information Systems, vol. 7, no. 1, pp. 1–22, 2005.

[36] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn 1997;29(2–3):131–63

[37] Zhang T, Vijay SI. Recommender systems using linear classifiers. J Mach Learn Res 2002;2:313–34.

[38] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. IEEE Comput 2009;8:30–7.

[39] Bojnordi E, Moradi P. A novel collaborative filtering model based on combination of correlation method with matrix completion technique. In: 16th CSI international symposium on artificial intelligence and signal processing (AISP), IEEE; 2012.

[40] Taka´cs G, Istva´n P, Bottya´n N, Tikk D. Investigation of various matrix factorization methods for large recommender systems. In: IEEE international conference on data mining workshops. ICDMW'08. IEEE; 2008. p. 553–62.

[41] T. Hofmann, "Latent semantic models for collaborative filtering," ACM Transactions on Information Systems, vol. 22, no. 1, pp. 89–115, 2004.

[42] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," Machine Learning, vol. 42, no. 1-2, pp. 177–196, 2001.

[43] Schafer JB, Frankowski D, Herlocker J, Sen S. Collaborative filtering recommender systems. In: Brusilovsky P, Kobsa A, Nejdl W, editors. The Adaptive Web, LNCS 4321. Berlin Heidelberg (Germany): Springer; 2007. p. 291–324.

[44] Burke R. Web recommender systems. In: Brusilovsky P, Kobsa A, Nejdl W, editors. The Adaptive Web, LNCS 4321. Berlin Heidelberg (Germany): Springer; 2007. p. 377–408. http:// dx.doi.org/10.1007/978-3-540-72079-9_12.

[45] Burke R. Hybrid recommender systems: survey and experiments. User Model User-adapted Interact 2002;12(4):331–70.

[46] Park DH, Kim HK, Choi IY, Kim JK. A literature review and classification of recommender systems research. Expert Syst Appl 2012;39(11):10059–72.

[47] Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. (2000). Application of Dimensionality Reduction in Recommender System{A Case Study. In ACM WebKDD 2000 Workshop.

[48] Marko Balabanovic and Yoav Shoham. Fab: Content-based, collaborative recommendation. Communications of the Association for Computing Machinery, 40(3):66–72, 1997.

[49] C. Basu, H. Hirsh, and W. Cohen. Recommendation as classification: Using social and content-based information in recommendation. In Proceedings 14 of the Fifteenth National Conference on Artificial Intelligence (AAAI-98), pages 714–720, July 1998.

[50] Bishop CM. Pattern recognition and machine learning, vol. 4, no. 4. Springer, New York; 2006.

[51] Duda RO, Hart PE, Stork DG. Pattern classification. John Wiley & Sons; 2012.

[52] [Schwab & Pohl 1999] Schwab, I. and Pohl, W. (1999). Learning User Profiles from Positive Examples. In Proceedings of the International Conference on Machine Learning & Applications, pp. 15-20. Chania, Greece.

[53] [Yan & Garcia-Molina 1994] Yan, T. W. and Garcia-Molina, H. (1994) Index Structures for Information Filtering Under the Vector Space Model. In Proceedings of the International Conference on Data Engineering.

[54] McSherry D. Explaining the pros and cons of conclusions in CBR. In: Calero PAG, Funk P, editors. Proceedings of the European conference on case-based reasoning (ECCBR-04). Madrid (Spain): Springer; 2004. p. 317–30.

[55] Magnini, B., Strapparava, C.: Improving User Modelling with Content-based Techniques. In: Proceedings of the 8th International Conference of User Modeling, pp. 74–83. Springer (2001)

[56] Melville, P., Mooney, R.J., Nagarajan, R.: Content-Boosted Collaborative Filtering for Improved Recommendations. In: Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI-02), pp. 187–192. AAAI Press, Menlo Parc, CA, USA (2002)

[57] Billsus D, Pazzani MJ. A hybrid user model for news story classification. In: Kay J, editor. In: Proceedings of the seventh international conference on user modeling, Banff, Canada. Springer-Verlag, New York; 1999. p. 99–108.

[58] B. Smyth and P. Cotter, "A personalized TV listings service for the digital TV age," in Proceedings of the 19th International Conference on Knowledge-Based Systems and Applied Artificial Intelligence (ES '00), vol. 13, pp. 53–59, Cambridge, UK, December 2000.

[59] Z. Huang, W. Chung, and H. Chen, "A graph model for Ecommerce recommender systems," Journal of the American Society for Information Science and Technology, vol. 55, no. 3, pp. 259–274, 2004.