ISSN 2047-3338

Towards a Perspective of Hybrid Approaches and Methodologies in Recommender Systems

Nana Yaw Asabere
Mobile Computing Laboratory, School of Software, Dalian University of Technology, Dalian, P.R. China
yawasbere2005@yahoo.com

Abstract—**Recommender Systems apply machine learning and data mining techniques to filter undetected information and can predict whether a user of a system would like a given resource based on his/her interests and preferences. To date a number of recommendation algorithms have been proposed, where Collaborative Filtering (*CF*) and Content-Based Filtering (*CBF*) are the two most famous and adopted recommendation techniques. CF Recommender Systems recommend items by identifying other users with similar taste and use their opinions for recommendation. *CF* Recommender Systems suffer from problems and challenges such as scalability, first rater (new item), data sparsity and cold-start problems. On the other hand, *CBF* Recommender Systems recommend items based on the content information of the items and match these items with interest and preferences of a user and therefore suffer from an overspecialization problem. In order to generate accurate and good recommendations, Hybrid Recommender Systems combine *CF* and *CBF* Recommender Systems to avoid the above aforementioned problems and challenges. This paper initially discusses Recommender Systems in general, then presents an overview of the state-of-the-art research in the area of Hybrid Recommender Systems, specifically from the perspective of types, applications, architectures and algorithms and finally discusses relevant open issues of Hybrid Recommender Systems.**

Index Terms— **Collaborative Filtering (*CF*), Content Based Filtering (*CBF*), Hybrid Recommender Systems and Hybridization Methods/Techniques**

I. INTRODUCTION

THERE has been a vast increase in the volume of available digital information, electronic sources, and online services in recent years. This has resulted in information overload and has created a potential problem, which is how to filter and efficiently deliver relevant information to a user [1], [2]. Moreover, information needs to be prioritized for a user rather than just filtering the right information. Search engines such as Google and MSN help internet users by filtering pages to match explicit queries, using simple keywords [1]. The Semantic Web, also provides some help to find useful information by allowing intelligent search queries, however it depends on the extent of the annotations on the web pages [1]. Information overload problems highlight and specify a need for information extraction and data mining systems that can filter unseen information and can predict whether a user would like a given resource. Such systems are called *recommender systems* and date back as far as the 1990s [1, 3]. Recommender systems alleviate the aforementioned problems to a great extent. Given a new item, recommender systems can predict with impressive accuracy whether a user would like an item or not, based on user interests and preferences (likes- positive examples, and dislikes- negative examples), observed behaviour, and information (demographic or content information) about the item in question [1], [2], [4], [5].

Some well-known examples of video recommender services are *YouTube*[1] [6] and *MovieLens*[2] [1] movie recommender system, which recommend videos and movies based on the user's opinions. Another example of the recommender system is the *Amazon*[3] recommender engine [7], which can filter through millions of available items based on the preferences or past browsing behaviour of a user and can make personal book recommendations [1]. In recommender systems, a history of user's interactions with the system is stored thereby making the recommender system conversant with his/her interests and preferences. The history of the user can be gathered by explicit feedback, whereby the user rates some items through some scales or by implicit feedback, whereby the user's interaction with the item is observed- for instance, if a user purchases an item then this is a sign that he likes that item, his/her browsing behaviour, etc [1], [2].

Collaborative Filtering (*CF*) and Content-Based Filtering (*CBF*) Recommender Systems suffer from problems of *first rater (new item)*, *cold-start (new-user)*, *data sparsity*, *scalability* and *overspecialization* respectively. Researchers have therefore come out with and developed/designed appropriate techniques to combine *CF* and *CBF* as well as other recommender systems such as Demographic and Knowledge-Based in order to solve the challenges and problems of each of the recommender systems and also to generate accurate, efficient and trustworthy recommendations. The main contribution of this paper is to outline the state-of-the-art of research in the area of hybrid recommender systems, specifically from the perspective of types, applications, architectures and algorithms. The rest of the paper is structured and organized as follows: Section II

[1] www.youtube.com
[2] www.movielens.org
[3] www.amazon.com

presents a Background and Categorization of Recommender Systems, Section III discusses Existing Research and presents an Overview of the State-of-the-Art, Section IV elaborates on relevant Open Issues of Hybrid Recommender Systems and Section V finally concludes the paper.

II. BACKGROUND AND CATEGORIZATION OF RECOMMENDER SYSTEMS

A. General Architecture

As discussed in Section I, Recommender Systems basically suggest items of interest to users based on: their implicit and explicit preferences, the preferences/ interests of other users and user and item attributes [1], [2], [8]. Recommender Systems generally share in common a means for describing items that may be recommended, a means for creating a profile of the user that describes the types of items the user likes, and a means of comparing items to the user profile to determine what to recommend. The user profile is often created and updated automatically in response to feedback on the desirability of items that have been presented to the user [9].

(i) Learning a User Model and Information Collection Recommender Systems through explicit (by querying the user) and implicit (by observing the user's behaviour) processes/methods collect all the usable information for the prediction task including the users' attributes, behaviors, or the content of the resources the user accesses [1, 2]. Further explanation of Explicit Feedback and Implicit Feedback is given below:

1) Explicit feedback: In explicit feedback, the user of the recommender system provides data willingly. Generally, users are required to fill interface forms at the beginning of a sign up process of the system, so the system can learn a user model and form a user profile [1], [2], [9]. These forms, depending on the recommender system, usually require the filling of basic demographic information such as age, gender, education, occupation, location and user interests. An Academic Recommender (AR) system will for example require the educational interest of the learner, as well as his/her learning styles and cognitive abilities towards learning [10]. In a movie recommender system such as *MovieLens*, the user can state interests as "I like comedy films" or "I don't like action films".

2) Implicit feedback: In implicit feedback, the user is not aware of the fact that he/she is providing feedback and his/her behaviour is being observed by the recommender system. Implicit feedback can be gathered by monitoring the user activity and behaviour [1], [2], [9]. In implicit feedback, users are not disturbed, but the gathered results might not be as relevant as the results that are collected from explicit feedback [1, 2, 9]. An example could be, if a user purchases an item that is a sign that the user likes the item, while if the user purchases and returns an item that is a sign that the user doesn't like the item [9].

In general, there is a tradeoff since implicit methods can collect a large amount of data with some uncertainty as to whether the user actually likes the item. In contrast, when the user explicitly rates items, there is little or no noise in the training data, but users tend to provide explicit feedback on only a small percentage of the items they interact with [1], [2], [9].

Explicit and implicit data acquisition methods present advantages and disadvantages: explicitly acquired data are more accurate in expressing preferences or interests but implicit methods are considered unassuming to user's main goal in using a system. According to [2] in web-based recommender systems (like *queveo.tv* [2]) with rich interaction data available, users are typically asked to rate observed items in a one to five scale (e.g. amazon.com).

(ii) Learning Process: After collecting information and learning a model of a user, a recommender system then applies its learning algorithm to filter and exploit the users' features from the collected information.

(iii) Resources Prediction: After the learning process, predictions on what kind of resources the user may prefer are then made either directly based on the dataset collected in the information collection stage (memory-based predictions) or with a model learned from it (model-based predictions).

B. Categories of Recommender Systems

Recommender Systems are usually classified into the following categories [1], [2]:

(i) Collaborative or Social Recommender Systems: In this category, users are recommended items that people with similar tastes, interests and preferences liked in the past. Through collaborative filtering algorithms, collaborative recommender systems identifies users who share the same preferences (e.g., rating patterns) with the active user, and proposes items which the like-minded users. Collaborative or Social Recommender Systems, which is the most well-known type of recommender systems do not have the *overspecialization* drawbacks as the content-based recommendations, but have their own limitations such as the *cold-start* (*new user)* problem, the *new item* problem, and the rating *data sparsity* problem [1], [2], [3], [9], [11].

1) Data Sparsity: According to [12], many commercial recommender systems are used to evaluate very large product sets. The user-item matrix used for collaborative filtering will thus be extremely sparse and the performances of the predictions or recommendations of the *CF* systems are challenged. The data sparsity challenge appears in several situations, specifically, the *cold start and first rater* problems occur when a new user and new item respectively just enters the system, it is difficult to find similar users and items because there is not enough information in the system [1], [2], [12]. New items cannot be recommended until some users' rate it, and new users are unlikely given good recommendations because of the lack of their rating or purchase history [12].

According to [12], *Coverage* can be defined as the percentage of items that the algorithm could provide recommendations for. The *reduced coverage* problem occurs when the number of users' ratings may be very small compared with the large number of items in the system, and the recommender system may be unable to generate recommendations for them.

Neighbour Transitivity refers to a problem with sparse databases, in which users with similar tastes may not be identified as such if they have not both rated any of the same items. This could reduce the effectiveness of a recommendation system which relies on comparing users in pairs and therefore generating predictions [12].

To eliminate the data sparsity problem, many approaches have been proposed. Dimensionality reduction techniques, such as *Singular Value Decomposition (SVD)* [2], [13], remove unrepresentative or insignificant users or items to reduce the dimensionalities of the user-item matrix directly. Hybrid *CF* algorithms, such as the *content-boosted CF* algorithm [14], are found helpful to address the sparsity problem, in which external content information can be used to produce predictions for *new users* or *new items*. Model-based *CF* algorithms, such as Tree Augmented Naive Bayes optimized by Extended Logistic Regression (*TAN-ELR)* and Naïve Bayes Extended Logistic Regression *(NB-ELR)* [7], [12], [15], [16], address the sparsity problem by providing more accurate predictions for sparse data.

2) Scalability: When the number of existing users and items grow tremendously, traditional *CF* algorithms will suffer serious scalability problems, with computational resources going beyond practical or acceptable levels [12]. For example, with tens of millions of customers (*M*) and millions of distinct items (*N*), a *CF* algorithm with the complexity of $O(n)$ is already too large [12]. Many recommender 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 [12, 17]. Dimensionality reduction techniques such as *SVD* can deal with the scalability problem and quickly produce good quality recommendations, but they have to undergo expensive matrix factorization steps [12].

3) Synonymy: Synonymy refers to the tendency of a number of the same or very similar items to have different names or entries. Most recommender systems are unable to discover this latent association and thus treat these products differently [12]. For example, the seemingly different items "*adult movie*" and "*adult film*" are actual the same item, but memory-based *CF* systems would find no match between them to compute similarity. Indeed, the degree of variability in descriptive term usage is greater than commonly suspected [12]. The dominance of synonyms decreases the recommendation performance of *CF* systems. Previous attempts to solve the synonymy problem depended on intellectual or automatic term expansion, or the construction of a thesaurus [12]. The *SVD* techniques, particularly the *Latent Semantic Indexing* (*LSI*) method, are capable of dealing with the synonymy problems [12].

4) Gray Sheep and Black Sheep: According to [18], Gray Sheep refers to the users whose opinions do not consistently agree or disagree with any group of people and thus do not benefit from collaborative filtering. *Black sheep* are the opposite group whose idiosyncratic tastes make recommendations nearly impossible. Non-electronic recommenders also have great problems in *black sheep* cases, so *black sheep* is an acceptable failure [19]. Claypool et al. [18] provided a hybrid approach combining content-based and *CF* recommendations by basing a prediction on a weighted average of the content-based prediction and the *CF* prediction. In that approach, the weights of the content-based and *CF* predictions are determined on a per user basis, allowing the system to determine the optimal mix of content-based and *CF* recommendation for each user, helping to solve the *gray sheep* problem [18].

5) Shilling Attacks: In cases where any user can provide recommendations, people may give a great deal of positive recommendations for their own materials and negative recommendations for their competitors. It is desirable for *CF* systems to introduce precautions that discourage this kind of phenomenon [4, 12]. Recently, the *shilling attacks models* for collaborative filtering system have been identified and their effectiveness has been studied. Mobasher et al. [20] examined attack models for shilling the item-based *CF* systems using alternative *CF* systems such as hybrid *CF* systems and model-based *CF* systems believed to have the ability to provide partial solutions to the bias injection problem. O'Mahony et al. [21] contributed to solving the *shilling attacks* problem by analyzing robustness, a recommender system's resilience to potentially malicious perturbations in the customer/product rating matrix. Lam and Riedl [22] found that item-based *CF* algorithm was much less affected by the attacks than the user-based *CF* algorithm, and they suggest that new ways must be used to evaluate and detect *shilling attacks* on recommender systems.

Collaborative Filtering (*CF*) use two main approaches namely User-Based *CF* and Item-Based *CF* [1], [2], [12].
User-Based *CF* Systems usually use two steps [1, 2]:

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

Alternatively, Item-Based *CF* proceeds in an item-centric manner:

1. Build an item-item matrix determining relationships between pairs of items.
2. Using the matrix, and the data on the current user, to infer his/her tastes and interests.

The authors in [2] used both User-Based *CF* and Item-Based *CF* to develop their Hybrid Recommender system. The authors in [1] used Item-Based *CF* to improve a switching Hybrid Recommender System using Naïve Bayes Classifier and Collaborative Filtering.

(ii) Content-Based Recommender Systems: In this category, recommendations are provided by automatically matching a user's interests with items' contents. Items that are similar to ones the user preferred in the past are effectively recommended. It must be noted that recommendations are made without relying on information provided by other users, but solely on items' contents and users' profiles. In content-based filtering the features used to describe the content are of primary importance. The more descriptive they are the more accurate the prediction is. In Content-Based filtering only very similar items to previous items consumed by the user are recommended which creates a problem of *overspecialization* since there may be other items which are relevant and can be recommended but because they haven't been rated by the user before, recommendation becomes impossible [1], [2], [8], [9], [23].

(iii) Hybrid Recommender Systems and Hybridization Methods/Techniques: Both recommender systems described above present challenges, advantages and disadvantages and significant research effort has been devoted to hybrid recommendation methods that combine collaborative and content-based filtering, which results in exploiting the advantages of both methods [1], [2], [8], [12], [15], [23], [36]. Fig. 1 depicts a simple hybrid recommender system involving a mixed approach of *CF* and *CBF*. Burke [8] outlined different the Hybridization Methods for Recommender Systems, described below and further elaborated in Table I.

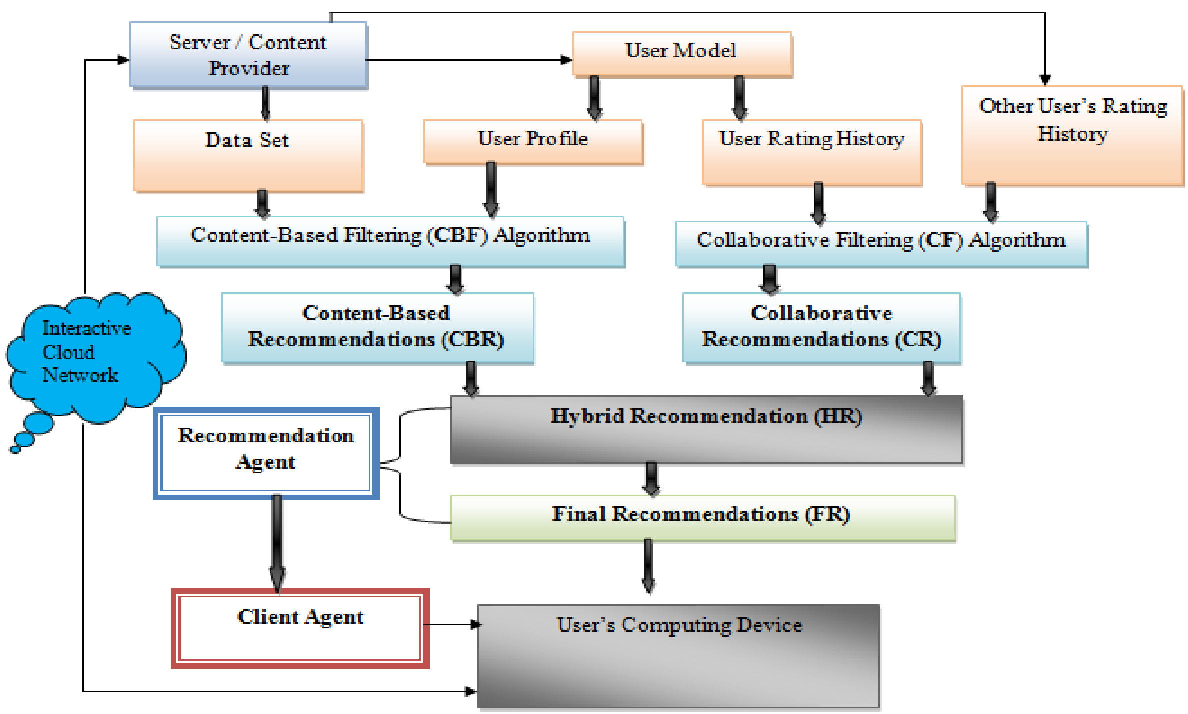

Fig. 1: CF and CBF Hybrid Recommender System

In Fig. 1, the User Model and User's Computing Device are all linked/connected to a server through an interactive cloud network. To generate *Content-Based Recommendations (CBR)* the User's Model which comprises his/her rating history and profile is matched with the data set stored in the server through a *Content-Based Filtering (CBF) Algorithm*. To generate *Collaborative Recommendations (CR)* the User is recommended information from the server based on matching between User's Rating History and Other User's Rating History with similar interests through a *Collaborative Filtering (CF) Algorithm*. In the *Recommendation Agent*, *CBR* is combined with *CR* to generate a *Hybrid Recommendation (HR)* and thus Final Recommendations. The User's Computing Device manages his/her Final Recommendations in the *Client Agent* which is connected to the *Recommendation Agent*.

1) Weighted Hybridization: A weighted hybrid recommender system is one in which the score of a recommended item is computed from the results of all of the available recommendation techniques present in the system. For example, the simplest combined hybrid would be a linear combination of recommendation scores. The *"P-Tango"* System in Claypool et al. [18] uses a weighted hybrid. It initially gives collaborative and content-based recommenders equal weight, but gradually adjusts the weighting as predictions about user ratings are confirmed or disconfirmed. The benefit of a weighted hybrid is that all of the system's capabilities are brought to bear on the recommendation process in a straightforward way and it is easy to perform post-hoc credit assignment and adjust the hybrid accordingly. However, the implicit assumption in this technique is that the relative value of the different techniques is more or less uniform across the space of possible items [8].

2) Switching Hybridization: A switching hybrid builds in item-level sensitivity to the hybridization strategy: the system uses some criterion to switch between recommendation techniques that form the hybrid. Ghazanfar and Prugel-Bennett [1] improved a Hybrid Recommender System by switching between an Item-Based *CF* Algorithm and Naïve Bayes Classifier Algorithm to generate appropriate recommendation. The "*DailyLearner"* system cited by Burke [8] uses a content/collaborative hybrid in which a content-based recommendation method is employed first. If the content-based system cannot make a recommendation with sufficient confidence, then a collaborative recommendation is attempted. Switching hybrids introduce additional complexity into the recommendation process since the switching criteria must be determined, and this introduces another level of parameterization. However, the benefit is that the system can be sensitive to the strengths and weaknesses of its constituent recommenders [8].

3) Mixed Hybridization: In a situation where it is practical to make large number of recommendations simultaneously, it is possible to use a "mixed" hybrid. Mixed Hybrid involves generation and prediction of recommendations from more than one technique. The authors Smyth and Cotter [24] discuss the PTV System which uses a mixed hybrid approach to assemble a recommended program of television viewing. The PTV System uses content-based techniques based on textual descriptions of TV shows and collaborative information about the preferences of other users. Martinez et al. [2] developed a TV Recommender System (*queveo.tv*) that also uses both Content-Based and Collaborative Filtering as a mixed approach to generate appropriate recommendation for TV users. Recommendations from the two techniques (*CF* and *CBF*) are combined together in the final suggested TV program(s). The mixed hybrid avoids the "new item" start-up problem: the content-based component can be relied on to recommend new shows on the basis of their descriptions even if they have not been rated by any of the users [1], [2],[8].

4) Feature Combination Hybridization: Another way to achieve an efficient and reliable merger of *CF* and *CBF* is to treat collaborative information as simply additional feature data associated with each example and use content-based techniques over this augmented data set [8]. For example, the authors Basu et al. [43] discuss and report on experiments in which the inductive rule learner "*Ripper*" was applied to the task of recommending movies using both user ratings and content features, and achieved significant improvements in precision over a purely collaborative approach. However, this benefit was only achieved by hand filtering content features. The authors Basu et al. [43] found out that employing all of the available content features improved recall but not precision. The feature combination hybrid allows the system to consider collaborative data without relying on it entirely, so it reduces the sensitivity of the system to the number of users who have rated an item. Equally, feature combination hybrid allows the system to have information about the inherent similarity of items that are otherwise impervious to a collaborative system.

5) Cascade Hybridization: The cascade hybrid, unlike the previous hybridization methods, involves a staged process. In this technique, one recommendation technique is employed first to produce a coarse ranking of candidates and a second technique refines the recommendation from among the candidate set [8]. The restaurant recommender "*EntreeC"*, cited and described by Burke [8] is a cascaded knowledge-based and collaborative recommender. Like Entree, it uses its knowledge of restaurants to make recommendations based on the user's stated interests. The recommendations are placed in buckets of equal preference, and the collaborative technique is employed to break ties, further ranking the suggestions in each bucket. Cascading allows the system to avoid employing the second lower-priority techniques on items that are already well-differentiated by the first or that are sufficiently poorly-rated that they will never be recommended. According to Burke [8], because the cascade's second step focuses only on those items for which additional discrimination is needed, it is more efficient than, for example, a weighted hybrid that applies all of its techniques to all items. Burke [8] further discusses that in addition, the cascade is by its nature tolerant of noise in the operation of a low-priority technique, since ratings given by the high-priority recommender system can only be refined, not overturned.

6) Feature Augmentation Hybridization: In the feature augmentation hybrid, one technique is employed to produce a rating or classification of an item and that information is then incorporated into the processing of the next recommendation technique. For instance, the "*Libra*" system developed by Mooney & Roy [25] makes content-based recommendations of books based on data found in Amazon.com, using a naïve Bayes text classifier. The text data used by the system in [25] included "related authors" and "related titles" and information that Amazon generates using its internal collaborative systems. These features were found to make a significant contribution to the quality of recommendations [8]. While both the cascade and feature augmentation techniques sequence two recommenders, with the first recommender having an influence over the second, they are fundamentally quite different. In an augmentation hybrid, the features used by the second recommender include the output of the first one, such as the ratings contributed by *GroupLens' filterbots* in Sarwar et al. [26]. In a cascaded hybrid, the second recommender does not use any output from the first recommender in producing its rankings, but the results of the two recommenders are combined in a prioritized manner [8].

7) Meta-level Hybridization: Another way that two recommendation techniques can be combined is by using the model generated by one as the input for another. This differs from feature augmentation: in a feature augmentation hybrid, a learned model is used to generate features for input to a second algorithm; in a meta-level hybrid, the entire model becomes the input. The first meta-level hybrid was the web filtering system "*Fab*" developed by Balabanovic [27], [28]. In Fab, user-specific selection agents perform content-based filtering using Rocchio's method discussed in [9] to maintain a term vector model that describes the user's area of interest. Collection agents, which garner new pages from the web, use the models from all users in their gathering operations. Therefore, documents are first collected on the basis of their interest to the community as a whole and then distributed to particular users. In addition to the way that user models were shared, *Fab* was also performing a cascade of collaborative collection and content-based recommendation, although the collaborative step only created a pool of documents and its ranking information was not used by the selection component [8].

The benefit of the meta-level method, especially for the content/collaborative hybrid is that the learned model is a compressed representation of a user's interest and a collaborative mechanism that follows can operate on this information-dense representation more easily than on raw rating data [8].

TABLE I
HYBRIDIZATION TECHNIQUES/METHODS

Hybridization Method	**Description**
Switching	The system switches between recommendation techniques depending on the current situation
Mixed	Recommendations from several different recommenders are presented at the same time
Weighted	The scores (or votes) of several recommendation techniques are combined together to produce a single recommendation.
Cascade	One recommender refines the recommendations given by another.
Feature combination	Features from different recommendation data sources are thrown together into a single recommendation algorithm.
Meta-level	The model learned by one recommender is used as input to another.
Feature augmentation	Output from one technique is used as an input feature to another.

III. HYBRID RECOMMENDER SYSTEMS

Hybridization can eliminate some of the problems associated with *CF*, *CBF* and other recommendation techniques. *CF* and *CBF* Hybrids, regardless of type, will always demonstrate the ramp-up problem (new user and new item) since both techniques need a database of ratings [8]. According to Burke [8], such hybrids with rump-up problems are still popular, because in many situations such ratings already exist or can be inferred from data. Meta-Level Hybrid techniques avoid the problem of sparsity by compressing ratings over many examples into a model, which can be more easily compared across users. Knowledge-Based and Utility-Based techniques seem to be good candidates for hybridization since they are not subject to ramp-up problems. Some exiting researches involving State-of-the-Art Hybrid Recommender Systems include the following:

Due to the notable problems and challenges of *CF* and *CBF* mentioned in this paper, the study in Ghazanfar and Prugel-Bennett [1] proposed a unique switching hybrid recommendation approach by combining a Naive Bayes classification approach with the collaborative filtering. Experimental results in [1] on two different data sets (movie lens and film trust) show that the proposed algorithm is scalable and provides better performance in terms of accuracy and coverage than other algorithms and at the same time eliminates some recorded problems with the recommender systems.

In Martínez et al. [2] discussions are presented on the fact that people are exposed to information overload, because of the expansion of digital networks and TV devices and the rapid increase of the number of channels due to the presence of several hundreds of alternative programs to watch. In this context, personalization is achieved with the employment of algorithms and data collection schemes that predict and recommend to television viewers content and match their interests and/or needs. Martínez et al. [2] introduced *queveo.tv*: a personalized TV program recommendation system. Their proposed mixed hybrid approach (combining content-filtering techniques with collaborative filtering) also provides all typical advantages of any social network as comments, tagging, ratings, etc. The web 2.0 application (*queveo.tv*) in [2] has been devised to enormously simplify the task of selecting what program to watch on TV.

Enabling transparent and augmented use of multimedia content across a wide range of networks and devices is still a challenging task within the multimedia research community. Within multimedia frameworks, content adaptation is the core concept to overcome this issue. Most media adaptation engines targeting Universal Multimedia Access (UMA) scale the content with respect to terminal capabilities and network resource constraints and do not sufficiently consider user preferences. The study in Prangl et al. [29] therefore focused on a hybrid recommender technique for configuring a cross-modal utility model that guides adaptation of multimedia content. The approach used in [29] additionally considers the user environment as well as demographic user data which lead to a personalized and increased multimedia experience. Based on a related adaptation decision technique, [29] showed how it is possible to offer a personalized adaptation for the individual user. The study in [29] presented a detailed evaluation of their approach based on results earned by subjective tests.

Mobile data communications have evolved as the number of third generation (3G) subscribers has increased to conduct mobile commerce. Multichannel companies would like to develop mobile commerce but meet difficulties because of lack of knowledge about users' consumption behaviors on the new mobile channel. Typical collaborative filtering (*CF*) recommendations may suffer from the so-called *sparsity* problem because few products are browsed on the mobile Web. The study in Liou and Liu [30] therefore proposed a hybrid multiple channel method to resolve the lack of knowledge about users' consumption behaviors on the new channel and the difficulty of finding similar users due to the sparsity problem of the typical *CF*. Products are recommended to the new mobile channel users based on their browsing behaviors on the new mobile channel as well as consumption behaviors on the existing multiple channels according to different weights. The experimental results in [30] show that the proposed method performs well compared to the other recommendation methods.

To meet more and more complex recommendation needs, it is quite important to implement hybrid recommendations for mobile commerce. In Liu et al. [31], a proposal for the design of open hybrid recommendation systems in mobile commerce, which could integrate multiple recommendation algorithms together to improve recommendation performance, is presented. First, three solutions for an open hybrid recommendation approach are discussed in detail, which are generic customer profile, weighted hybrid recommendation algorithm, and mobile device profile creation. After that, [31] present a multi-agent architecture design to make the three solutions work together. Finally a prototype system based on the proposed architecture in [31] is implemented to demonstrate the feasibility of their design and evaluation of the performance of the proposed open hybrid recommendation system.

In Ziegler et al. [32], a hybrid collaborative filtering approach was proposed to exploit bulk taxonomic information designed for exact product classification to address the data sparsity problem of *CF* recommendations, based on the generation of profiles via inference of super-topic score and topic diversification [32].

Schein et al. [33] proposed the *aspect model latent variable* method for *cold start* recommendation, which combines both collaborative and content information in model fitting [33].

Kim and Li [34] proposed a probabilistic model to address the *cold start* problem, in which items are classified into groups and predictions are made for users considering the Gaussian distribution of user ratings [34].

According to [12], the *content-boosted CF* recommender has improved prediction performance over some pure content-based recommenders and some pure memory-based *CF* algorithms. *content-boosted CF* also overcomes the *cold start* problem and tackles the *sparsity* problem of *CF* tasks.

Working on reasonably-sized subsets instead of the original rating data, Su and Khoshgoftaar [15] used *TAN-ELR* as the content-predictor and directly applied the *Pearson correlation-based CF* instead of a weighted one on the *pseudo rating matri*x to make predictions, and they achieved improved *CF* performance in terms of Mean Absolute Error (*MAE)* [15].

With the development of Mobile Internet, it is possible for people to access their daily portion of news over the Web with various types of terminals connected to it. "*EagleRadio"* proposed by Chen et al. [35] is a phonic Web news recommendation system which helps users access personalized news via mobile Internet. It reads news via a speech synthesizer to users. Chen et al. [35] proposed a hybrid news recommending strategy considering not only the user's listening history, but also taking the user's neighbors with similar tastes and preferences into account. The "*EagleRadio"* system proposed by Chen et al. [35] can track multiple threads of user's interests and the diversity of recommended news list is also guaranteed. Finally, Chen et al. [35] evaluated their proposed algorithms and quantify the effect of them from a user's perspective.

Collaborative filtering (*CF*) is one of the most successful approaches for recommendation. Su et al. [36], propose two hybrid *CF* algorithms, sequential mixture *CF* and joint mixture *CF*, each combining advice from multiple experts for effective recommendation. These proposed hybrid *CF* models work particularly well in the common situation when data are very sparse. By combining multiple experts to form a mixture *CF*, the proposed systems of Su et al. [36] are able to cope with sparse data to obtain satisfactory performance. Empirical studies in Su et al. [36] show that their algorithms outperform their peers, such as memory-based, pure model-based, pure content-based *CF* algorithms, and the *content-boosted CF* (a representative hybrid *CF* algorithm), especially when the underlying data are very sparse.

Ansari et al. [37] propose a Bayesian preference model that statistically integrates several types of information useful for making recommendations, such as user preferences, user and item features, and expert evaluations. They use Markov chain Monte Carlo (*MCMC*) methods for sampling based inference, which involve sampling parameter estimation from the full conditional distribution of parameters. Ansari et al. [37] achieved better performance than pure collaborative filtering.

In Melville et al. [14], a hybrid recommender framework for recommending movies to users is proposed. In the content-based filtering part, they get extra information about movies from the IMDB[4] web site and view each movie as a text document. A Naive Bayes classifier is used for building user and item profiles, which can handle vectors of bags-of-words [1]. The Naive Bayes classifier is used to approximate the missing entries in the user-item rating matrix, and a user-based CF is applied over this dense matrix. The problem of the approach in [14] is that it is not scalable. The hybrid recommendation approach used in Ghazanfar & Prugel-Bennett [1] combines Naive Bayes and Collaborative Filtering in a more scalable way than [14]. Ghazanfar & Prugel-Bennett [1] use synonym detection and feature selection algorithms that produce accurate profiles of users resulting in improved predictions. The off-line cost of Ghazanfar & Prugel-Bennett [1] and Melville et al. [14] are however the same, i.e. the cost to build a naive Bayes classifier. The on-line cost (in the worst case) in [1] is less than or equal to that of [14].

In Pazzani [40] a hybrid recommendation approach in which a content-based profile of each user is used to find the similar users, which are used for making predictions is proposed. Pazzani [40] used Winnow to extract features from user's home page to build the user content-based profile. The problem with the approach used in [40] is that if the content-based profile of a user is erroneous (may be due to synonyms problems or others), then it will results in poor recommendations [40].

Vozalis and Margaritis [41] and Vozalis and Margaritis [42] used demographic information about users and items for providing more accurate prediction for user-based and item-based CF. They [41, 42] proposed hybrid recommender systems (lying between cascading and feature combination hybrid recommender systems [8]) in which demographic correlation between two items (or users) is applied over the candidate neighbours found after applying the rating correlation on the user-item rating matrix. These refined sets of neighbours are used for generating predictions in [41], [42].

[4] www.imdb.com

TABLE II
PROPOSED AND POSSIBLE HYBRID RECOMMENDER SYSTEMS

HT / HC	**Weighted**	**Mixed**	**Switching**	**Feature Combination**	**Cascade**	**Feature Augmentation**	**Meta-Level**
CF/CBF	Claypool et al. [18] (*P-Tango)*	Smyth and Cotter [24], Schein et al. [33], Martinez et al. [2] (*queveo.tv)* and Chen et al. [35] (*EagleRadio)*	*DailyLearner* system cited by Burke [8] and	Basu et al. [43] (*Ripper)* and Pazzani [40]		Sarwar et al. [26] and Mooney & Roy [25] (*Libra)*	*Fab* System by Balabanovic [27] and Balabanovic [28]
CBF and CF Algorithms and Probabilistic Methods	Liu et al. [31] and Liou & Liu [30]	Ansari et al. [37] , Ziegler et al. [32], Su and Khoshgoftaar [15] & Su et al. [36]	Ghazanfar & Prugel-Bennett [1] and Melville et al. [14]		Kim and Li [34]		
KB/CF and CF/DF					*EntreeC*, cited and described by Burke [8]	Vozalis and Margaritis [41] and Vozalis and Margaritis [42]	
CF/KB/DF			Prangl et al. [29]				

(HT = Hybridization Techniques, HC = Hybrid Combination, CF = Collaborative Filtering, CBF = Content-Based Filtering, KB = Knowledge-Based and DF = Demographic Filtering)

However, in [41], [42], they completely miss the features of items for computing similarity [1], [2], [8], [9], [15], [36].

IV. DISCUSSION AND OPEN RESEARCH ISSUES

Hybridization methods and approaches in recommender systems seek to solve the problems, challenges and limitations of *CBF* and *CF* mentioned in this paper. In order to avoid the mentioned challenges, problems and limitations of these recommender system and improve recommendation performance, hybrid *CF* recommenders are combined by adding content-based characteristics to *CF* models, adding *CF* characteristics to content-based models, combining *CF* with content-based or other systems, or combining different *CF* algorithms [1], [2], [8], [12], [15], [36], [38], [39].

There are four hybridization approaches, methods and techniques that are order-insensitive: Weighted, Mixed, Switching and Feature Combination. With these hybridization methods, it does not make sense to talk about the order in which the techniques are applied: a *CBF/CF* mixed system would be no different from a *CF/CBF* one [8]. According to Burke [8], the cascade, augmentation and meta-level hybrids are inherently ordered. For instance, a feature augmentation hybrid that used a content-based recommender to contribute features to be used by a second collaborative process would be quite different from one that used collaboration first.

Burke [8] outlined and suggested some interesting types of recommenders that do not yet exist. Although collaborative filtering is the most fully explored technique, a number of its hybrids remain unexplored and need further research.

- *CBF/CF* Feature Augmentation Hybrid.
- *CF/CBF* Meta-level Hybrid, in which collaborative information is used to generate a representation of overall user ratings for an item and this representation is then used to compare across items.
- *CF*/Demographic Filtering Augmentation Hybrid in which a collaborative technique is used to place the user in a niche of like-minded users, and this information is used as a feature in a demographic rater.
- In addition, four cascade hybrid recommenders involving collaborative recommendation appear untried and yet to be researched into more detail.

The hybridization strategy must be a function of the characteristics of the recommenders being combined. With demographic, *CF* and *CBF* recommender systems, this is largely a function of the quality and quantity of data available for learning. With knowledge-based recommenders, it is a function of the available knowledge base. In accordance to this strategy, Burke [8] , discussed that, two cases can be distinguished: the uniform case, in which one recommender has better accuracy than another over the whole space of recommendation, and the non-uniform case, in which the two recommenders have different strengths in different parts of the space. If the recommenders are uniformly unequal, it may make sense to employ a hybrid in which the inaccuracies of the weaker recommender can be contained: for example, a cascade scheme with the stronger recommender given higher priority, an augmentation hybrid in which the weaker recommender acts as a “bot” contributing a small amount of information, or a meta-level combination in which the stronger technique produces a dense representation that strengthens the performance of the weaker one [8]. In the non-uniform case, the system will need to be able to employ both recommenders at different times. A switching hybrid as depicted in [1, 29, 40] is a natural choice for Hybrid Recommender Systems, but it requires that the system be able to detect when one recommender should be preferred. Feature combination and mixed hybrids can be used to allow output from both recommenders without having to implement a switching criterion. More research is needed to establish the tradeoffs between these hybridization options [8].

V. CONCLUSION

Recommender Systems since the 1990s have been very important and useful tools for filtering relevant information needs of users in domains such as education, entertainment, restaurant activities, tourism activities as well as museum activities. Recommender Systems literature however shows that individual recommender systems such as *CF* and *CBF* exhibit rump-up problems such as new user, new item constituting data sparsity as well as overspecialization and other relevant problems, challenges and limitation described in Section II of this paper. This paper, through relevant literature, delved into a perspective of hybrid recommender systems as a solution to solve problems, limitations and challenges of individual recommender systems. Hybrid recommender systems combine *CF* and *CBF* recommender systems as well as other recommender systems such as knowledge-based, utility, demographic and other *CF* algorithms to solve individual problems of the recommender systems. Research according to literature and depicted in this paper has shown that combination recommender systems using a particular type of hybridization technique has solved problems of individual recommender systems. However, as discussed in Section IV of this paper, there are still a lot of relevant issues to be tackled by researchers especially regarding the hybridization techniques to employ for a particular combination of recommender systems.

REFERENCES

[1] M.A. Ghazanfar and A. Prugel-Bennett "An Improved Switching Hybrid Recommender System Using Naive Bayes Classifier and Collaborative Filtering", *Proceedings of the International MultiConference of Engineers and Computer Scientists, IMECS,* Vol. 1, 2010.

[2] A.B.B. Matinez, J.J.P. Arias, A.F. Vilas, J.G. Duque and M.L. Nores "What's on TV Tonight? An Efficient and Effective Personalized Recommender System of TV Programs", *IEEE Transactions on Consumer Electronics,* Vol. 55, No. 1, pp. 286-294, 2009.

[3] W. Yuan, J. Zhang and J. Vassileva "Towards Effective Recommendation of Social Data across Social Networking Sites", *In Proceedings of the 14th International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA)*, pp. 61-70, 2010.

[4] P. Resnick and H. R. Varian "Recommender Systems," *Communications of ACM*, Vol. 40, No. 3, pp. 56–58, 1997.

[5] B. Mobasher, "Recommender Systems" *Kunstliche Intelligenz, Special Issue on Web Mining*, Vol. 3, pp. 41–43, 2007.

[6] S. Venkatesh, B. Adams, D. Phung, C. Dorai, R.G. Farrell, L. Agnihotri, and N. Dimitrova "You Tube and I Find" - Personalizing Multimedia Content Access", *Proceedings of IEEE* Vol. 96, No. 4.

[7] J. Y. G Linden, B Smith, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," in *IEEE, Internet Computing*, Vol. 7, 2003, pp. 76–80.

[8] R. Burke "Hybrid Recommender Systems: Survey and Experiments", *User Modeling and User-Adapted Interaction,* Vol. 12, pp. 331-37, 2002.

[9] M. Pazzani and D. Billsus "Content-Based Recommendation Systems", In: The Adaptive Web. Springer Berlin / Heidelberg, pp. 325–341, 2007.

[10] J. Zhou, T. Lou and F. Cheng "Modeling Learners and Contents in Academic-oriented Recommendation Framework", *Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing*, pp. 1017-1024, 2011.

[11] P.Y. Pan, C.H. Wang, G.J. Horng and S.T. Cheng "The Development of An Ontology-based Adaptive Personalized Recommender System", *Proceedings of the IEEE International Conference on Electronics and Information Engineering (ICEIE),* pp. 76-80, 2010.

[12] X.Su and T. M. Khoshgoftaar "A Survey of Collaborative Filtering Techniques" *Advances in Artificial Intelligence,* Volume 2009, Article ID 421425, pp. 1-19, Hindawi Publishing Corporation, 2009.

[13] D. Billsus and M. Pazzani, "Learning Collaborative Information Filters," *In Proceedings of the 15th International Conference on Machine Learning (ICML '98)*, 1998.

[14] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted Collaborative Filtering for Improved Recommendations," *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI '02)*, pp. 187–192, Edmonton, Canada, 2002.

[15] X. Su and T. M. Khoshgoftaar, "Collaborative Filtering for Multi-class Data Using belief Nets Algorithms," *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '06)*, pp. 497–504, 2006.

[16] R. Greinemr, X. Su, B. Shen, and W. Zhou, "Structural Extension to Logistic Regression: Discriminative Larameter Learning of Belief Net Classifiers," *Machine Learning*, Vol. 59, No. 3, pp. 297–322, 2005.

[17] B. Shen, X. Su, R. Greiner, P. Musilek, and C. Cheng, "Discriminative Parameter Learning of General Bayesian Network Classifiers," *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence*, pp. 296–305, Sacramento, Calif, USA, November 2003.

[18] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining Content-Based and Collaborative Filters in an Online Newspaper," *Proceedings of the SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, Berkeley, Calif, USA, 1999.

[19] J. McCrae, A. Piatek, and A. Langley "Collaborative Filtering" 15/06/2004 [Online] Available *http://wenku.baidu.com/view/5619da04e87101f69e319565.html*

[20] B. Mobasher, R. Burke, R. Bhaumik, and C. Williams, "Effective Attack Models for Shilling Item-Based Collaborative Filtering Systems," *Proceedings of the WebKDD Workshop*, August 2005.

[21] M. O'Mahony, N. Hurley, N. Kushmerick, and G. Silvestre "Collaborative recommendation: a robustness analysis," *ACM Transactions on Internet Technology*, Vol. 4, No. 4, pp. 344–377, 2004.

[22] S. K. Lam and J. Riedl, "Shilling Recommender Systems for Fun and Profit,", *Proceedings of the 13th International World Wide Web Conference (WWW '04)*, pp. 393–402, New York, NY, USA, May 2004.

[23] A. Brun, S. Castagnos, and A. Boyer "Social Recommendations: Mentor and Leader Detection to Alleviate the Cold-Start Problem in Collaborative Filtering", *Book Chapter in Social Network Mining, Analysis and Research Trends: Techniques and Applications, pp. 270-290*, 2011.

[24] B. Smyth and P. Cotter "A Personalized TV Listings Service for the Digital TV Age", *Knowledge-Based* Systems, Vol. 13, pp. 53-59, 2000.

[25] R. J. Mooney and L. Roy "Content-based Book Recommending Using Learning for Text Categorization", *In Proceedings of the Fifth ACM Conference on Digital Libraries*, 195–204, 2000.

[26] B. M. Sarwar, J. A. Konstan, A. Borchers, J. Herlocker B. Miller and J. Riedl "Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System", *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, Seattle, WA, pp. 345-354, 1998.

[27] M. Balabanovic "An Adaptive Web Page Recommendation Service", *Agents 97: Proceedings of the First International Conference on Autonomous Agents*, Marina Del Rey, CA, pp. 378-385, 1997.

[28] M. Balabanovic "Exploring versus Exploiting when Learning User Models for Text Representation", *User Modeling and User-Adapted Interaction* 8 (1-2), 71-102, 1998.

[29] M. Prangl, R. Bachlechner, and H. Hellwagner "A Hybrid Recommender Strategy for Personalized Utility-Based Cross-Modal Multimedia Adaptation", *IEEE ICME*, pp. 1707-1710, 2007.

[30] C. Liou and D. Liu "Hybrid Multiple Channel-Based Recommendations for Mobile Commerce", *IEEE Proceedings of the 43rd Hawaii International Conference on System Sciences*, pp. 1-8, 2010.

[31] C. Liu C. Sun and J. Yu "The Design of an Open Hybrid Recommendation System for Mobile Commerce", *IEEE CIMCA, IAWTIC*, and *ISE*, pp. 861-866, 2008.

[32] C.-N. Ziegler, G. Lausen, and L. Schmidt-Thieme "Taxonomy-Driven Computation of Product Recommendations," *Proceedings of the 13th International Conference on Information and Knowledge Management (CIKM '04)*, pp. 406–415, Washington, DC, USA, November 2004.

[33] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock "Methods and Metrics for Cold-Start Recommendations" *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02)*, 2002.

[34] B. M. Kim and Q. Li, "Probabilistic Model Estimation for Collaborative Filtering Based on Items Attributes," *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI '04)*, pp. 185–191, Beijing, China, September 2004.

[35] W. Chen, L. Zhang, C. Chen and J. Bu "A Hybrid Phonic Web News Recommender System for Pervasive Access", *IEEE International Conference on Communications and Mobile Computing*, pp. 122-126, 2009.

[36] X. Su, R. Greiner, T.M. Khoshgoftaar and X. Zhu "Hybrid Collaborative Filtering Algorithms Using a Mixture of Experts" , *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 645-649, 2007

[37] A. Ansari, S. Essegaier, and R. Kohli "Internet Recommendation Systems," *Journal of Marketing Research*, Vol. 37, No. 3, pp. 363–375, 2000.

[38] G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 6, pp. 734–749, 2005.

[39] M. Balabanovic and Y. Shoham, "Content-Based, Collaborative Recommendation" *Communications of the ACM*, Vol. 40, No. 3, pp. 66–72, 1997.

[40] M. J. Pazzani "A Framework for Collaborative, Content-Cased and Demographic Filtering," *Artificial Intelligence Review*, Vol. 13, No. 5 - 6, pp. 393–408, December 1999.

[41] M. Vozalis and K. Margaritis "Collaborative Filtering Enhanced by Demographic Correlation", *in AIAI Symposium on Professional Practice in AI, of the 18th World Computer Congress*, 2004.

[42] M. Vozalis and K. Margaritis "On the Enhancement of Collaborative Filtering by Demographic Data," *Web Intelligence and Agent Systems*, Vol. 4, No. 2, pp. 117–138, 2006.

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

Nana Yaw Asabere received his BSc in Computer Science from Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana in 2004 and MSc in ICT from Aalborg University, Denmark in 2010. He has eight (8) years of teaching/lecturing experience at tertiary level of education in Ghana and is currently on Lectureship Study Leave granted by Accra Polytechnic, Ghana pursuing his PhD in Computer Software Engineering at School of Software, Dalian University of Technology, Dalian, P.R. China. Nana Yaw has a number of publications to his credits in International Journals and his research interests include: Artificial Intelligence (AI), Software Engineering, Expert Systems, Mobile Learning, E-learning, ICT in Education, Information Systems, Multimedia, Recommender Systems, Social Computing, Wireless/Data/Mobile Communication and Computing Technologies.