Alleviating Unwanted Recommendations Issues in Collaborative Filtering Based Recommender Systems

Abba Almu*
Department of Computer Science, Usmanu Danfodiyo University, P.M.B 2346, Sokoto, Nigeria
Email: almu.abba@udusok.edu.ng
ORCID ID: https://orcid.org/0000-0003-2302-3232
*Corresponding Author

Abubakar Roko
Department of Computer Science, Usmanu Danfodiyo University, P.M.B 2346, Sokoto, Nigeria
Email: roko.abubakar@udusok.edu.ng

Abstract: The overabundance of information on the internet and ecommerce has resulted to the development of recommender system to discover interesting items or contents that are recommendable to the user. The recommended items might be of no interest or unwanted to the users and can make users to lose interest in the recommendations. In this work, a Collaborative Filtering (CF) based method which exploits the initial top-N recommendation lists of an item-based CF algorithm based on unwanted recommendations penalisation is presented. The method utilises a relevance feedback mechanism to solicit for users preferences on the recommendations while popularise similarity function minimises the chances of recommending unwanted items. The work explains the proposed algorithm in detail and demonstrates the improvements required on existing CF to provide some adjustments required to improve subsequent recommendations to users.

Index Terms: Recommender System, Collaborative Filtering, Unwanted Recommendations, Relevance Feedback, Popularise Similarity Function.

1. Introduction

The drastic increase of huge amount of information on daily basis on the web and the emergence of e-commerce have resulted to the problem typically known as information overload which makes it difficult to the users to find interested items. This problem led to the development of recommender system as a means of finding items of interest to the user. Recommender system serves an information filtering system that recommends relevant items to users by analyzing the users explicit and implicit identified preferences or interests [1, 2]. The system saves a lot of time and effort of users involved in searching large amount of information to find items of interest, by simply prioritising and personalising large volume of information available at its disposal to find the unknown relevant items needed by the users.

The recommender systems are categorised into three major types [3]. These include Content-Based Filtering (CBF) [4, 5], Collaborative Filtering (CF) [6, 7] and Hybrid Filtering (HF) recommender systems [8, 9]. CBF recommender system recommends items to users that are similar in content to those items the users have already liked in the past [4]. It uses users' profiles as content representation and then compares those profiles with new items features in order to provide recommendations. CF recommender system recommends items by considering users' ratings of an item to find the match of rating patterns of some items involving other users with similar interests [10, 11]. Unlike CBF, this type of recommendation deals with users ratings given to items not content of the items. HF recommender system uses combination of CBF and CF recommender techniques to provide recommendations; it uses both users' ratings and profiles to generate recommendations for other users. Among these systems, CF is the most popular and successful recommendation technique that provides personalised recommendation to users [11]. Therefore, this work focuses on

CF recommender because it can be used to recommend any items to users and also it ignores the requirements for representation user and/or item attributes.

Despite the various researches conducted for CF-based recommender system to provide good recommendations to users, it suffers from the problem of unwanted recommendations in which little attention is given in the literature. This problem can make users to lose interest in the recommendations, if most of the recommendations presented fail to satisfy the user interests. In this work, a CF-based method that exploits the initial top-N recommendation lists of item-based which utilises CF algorithm based unwanted recommendations penalisation is proposed. To alleviate this, an algorithm is developed to solicit for users' preference on the recommendations via relevance feedback mechanism and then incorporates popularise similarity function to provide penalty on unwanted recommendations. The improvement is expected to provide some adjustments required to improve subsequent recommendations to users.

This work is organised into three sections as follows: Section 1 provides background of the study on recommender systems. Section 2 investigates the related works on CF based recommender a system with a view to identify the required improvements that necessitated this research work. Section 3 presents preliminaries related to CF based algorithm construction. Section 4 proposes CF algorithm based on unwanted recommendations penalisation. Section 5 summarises the chapter and identified future directions for the work.

2. Literature Review

There are many researches based on CF idea, which are classified based on the approaches used to refine the algorithms for improving the quality and accuracy of the recommendations. These include reranking, diversification, temporal dynamics and items popularity penalisation. In this section, the related works in CF based systems are presented by describing each of the algorithms as well as highlighting its strengths and weaknesses.

2.1 Reranking Approach

Reranking is the technique used to reorder the list of recommended items for improving the relevancy of the recommendations to users. It is introduced as the new technique to enrich the initial recommendations of items to the recommender users.

Shi, Larson and Hanjalic [12] introduced a novel approach called Rerank-Collaborative Recommendation (Rerank-CR) to improve the recommendation performance to users. The approach uses an algorithm that provides reranking by making use of the information available from the original user-item matrix. The algorithm combined two techniques namely CR and reranking to successfully improved the initial top-N recommendations list by reordering the list, such that highly relevant items are moved upward during recommendations. The proposed Rerank-CR algorithm is evaluated using precision and recall metrics on two different datasets namely MovieLens and EachMovie. The results indicated the algorithm performance significantly improved across different datasets. But, the algorithm in question considered single modality when reranking the initial recommendations.

In order to address the above mentioned issue, Shi, Larson and Hanjalic [13] presented a novel reranking model for CF recommendations called Multi-Rerank algorithm. The algorithm contains a function that makes use of multiple self-contained modalities to improve the recommendations using the user-item matrix. The function provides encoding for the similarity between items with respect to three modalities (number of ratings, average rating and number of highest rating for each item) which succeeded in exploiting multiple modalities while reranking in order to improve various collaborative filtering methods by enriching the recommendation list. The performance of the proposed algorithm is tested using mean average precision metric. The experimental results indicated that the algorithm is effective for achieving improvement in exploiting multiple modalities that is not achievable when single reranking modality is used. The reranking is triggered without considering user's input about the recommendations and the rank score used may not predict the items interested to a user.

The study of Jannach, Lerche and Gdaniec [14] proposed reranking recommender system based on user short-term interests in an online store system. The system uses a protocol that combines the matching score and Bayesian Personalised Ranking (BPR) in order to re-order the recommendations. The protocol models the user behaviour of past purchases and session log information to avoid placing low ranked items on top of the list of recommended items. In this reranking approach, the experiment is performed on e-commerce site dataset with different characteristics by using recall metric. The results show that ranking based on BPR and users short-term interests increases the accuracy of the recommended items better compared with popularity or baseline algorithms. The list of items recommended to the user might appear redundant since the technique can suggest the items already seen in the current or previous user sessions.

Zhong, Xiao and Duan [3] presented a user-based CF recommendation algorithm with center distance-based reranking capability to provide effective recommendations to user. The algorithm uses a method that considers both aggregate diversity and individual diversity to rerank the top N list of the original recommended items. The method will first find the top N items; second, compute the center value of the items; third, set a threshold; fourth, rank the M above-threshold items based on the center distance; fifth, rank the N-M below-threshold items by considering the method used by the standard algorithm and finally combine the N items as candidates list of recommended items to generate more accurate recommendations. The dataset used for the experimental evaluation is MovieLens, and the evaluation metrics considered are the accuracy and coverage measures. The results show that the contents coverage of

the final recommendations has improved greatly. The algorithm does not consider the user responses to the recommendation lists presented before reranking in order to provide more diverse subsequent recommendations to capture the personalised interests of the user.

2.2 Recommendation Diversification

Diversification involves the process of mixing different types of items in the recommendation lists to improve users' satisfaction. Different literatures have noticed the importance of diversifying recommendation lists to users.

Ziegler et al. [15] presented an algorithmic framework based on topic diversification to user-based and item-based CF for increasing the diversity of the top-N list of recommended items. The algorithm takes the accuracy of suggestions and user's interest in specific topics into account. It uses a method that tends to balance and diversify the new recommendation lists of items so as to reflect the current user's tastes. The approach is evaluated using precision, recall and intra-list similarity on BookCrossing dataset. The framework generates the recommendation lists that goes beyond accuracy and involve other factors such as those related to diversity. However, it cannot provide a right mix of diverse recommended items to address the demand of multiple user interests that the recommenders suppose to provide.

Zhang [16] proposed an improved novel algorithm based on recommendation lists diversification to promote user multiple interests' satisfaction. The algorithm used a technique that considers computation of item topic vector, trusted neighbours computation and prediction preference of the user concerned. The technique considered selecting diverse recommendation neighbours to improve the recommendation lists diversity. The algorithm is evaluated using epinions.com dataset and the results show that diversifying recommendations can improve the quality of the recommended items. The recommendation diversity focus on the similarity between items neighbours only without considering the users who are not similar to the active user.

Yang, Ai and Li [17] proposed neighbour diversification approach based on CF for the purpose of improving recommendation lists. The approach comprises of an algorithm that selects k diversity of neighbours instead of top-k most similar neighbours. The algorithm uses a diverse set of neighbours to provide recommendations to users based on this neighbour set with a view to improve the recommendation lists. The algorithm is evaluated on MovieLens dataset using six different evaluation metrics namely precision, recall, individual diversity, novelty, aggregate diversity and coverage. The results of the experiment proved that the efficiency of the proposed algorithm to improve the diversity of the recommendation lists. The study does not consider the different behaviour patterns of the individual users to the recommendation lists personalisation expectations.

Harper et al. [18] presented a recommender system that gives some control to users for improving the diversity of the resulting recommendations. The system has a method that uses a linear weighted combination of personalisation variable and blending variable. This recommendation method incorporated user-tuned popularity and recency modifiers to provide such control to users, therefore, indicating that users evaluated the fine-tuned recommendations much more positively than the ones without control. The experimental evaluation is conducted by inviting some existing MovieLens dataset users to participate in the study. The users were given some controls to evaluate the resulting recommendations and provide feedback. The results revealed that users are happier with the system, if they are given some controls over the recommendations. Though, the method is based on static users' preferences, and do not capture the dynamic nature of the recommendation processes.

2.3 Temporal Dynamics

Temporal dynamics is the technique used to predict the changes of user preferences on the items to be recommended over time. Existing literatures have investigated the benefits of incorporating temporal dynamics in the user's preferences on the recommended items.

Koren [19] modeled temporal dynamics in a collaborative filtering recommender system to demonstrate that user preferences change from time to time. The model uses a method that tracks the time changing behaviour of the items life span in order to separate transient factors from lasting ones. The method shows the customers' preferences to products or items changed over time and the result indicated the items rated by an individual customer decays over time. The evaluation is performed using a large movie ratings dataset by Netflix on two temporal effects (top and bottom ratings) emerging within the dataset. The quality of the approach is measured using Root Mean Squared Error (RMSE), and the experiment show the best results than those reported in the previous literature on the same dataset. The study shows that users to items interest decreased from time to time thereby affecting their rating patterns.

Lathia et al. [20] studied the importance of temporal diversity approach in CF recommender systems as it relates to dynamic nature of user preferences. The approach explores the changes data items undergo over time and how users' survey responded to the recommendations. This resulted in users given higher ratings to diversified recommendations and the lower ratings undiversified ones. The work evaluated three CF algorithms (Baseline, kNN and SVD) using diversity metric in the sequence of recommendations they produced over time. The results demonstrated how a number of features of user rating patterns affect the diversity. This necessitated the need to deal with repeated recommended items over a period of time so as to improve the recommendations diversity.

Yin, Wang and Yu [21] took a different dimension and studied the ways of improving temporal Euclidean embedding model in CF recommender systems. The model incorporates   temporal factors of rating behaviour to determine the relationship between users and items biases for better prediction accuracy. This model technique used increases the accuracy and the efficiency of generating recommendations to users. The proposed recommendation approach is evaluated using precision, recall and time metrics. The results of the experiment on MovieLens dataset indicated that, the fast strategy adopted reduces the recommendation time drastically. The work focused on improving accuracy and efficiency of the recommendations without taking the effect of the interaction between users and recommended items into account.

There was another study that focused on improving CF method [22], in which a new temporal CF algorithm is introduced to reflect the dynamic nature of user interest changes on the recommended items. The algorithm incorporates a method that uses items in relation to the predicted one to produce the similarity scores needed to capture the reality of user preferences. The method consists of a simple weighting function that computes each score with a weight for the different items based on user's interest changes according to time. The quality of the proposed temporal recommendation algorithm was evaluated on MovieLens dataset using MAE metric. The results demonstrated that the proposed algorithm has lower MAE values than the traditional CF algorithm. However, the recommendations provided by weakening the score value of the older items may not satisfy the user's dynamic interests.

2.4 Items Popularity Penalisation

Popular items are the items recommended with high rating values given by most of the users. Recent literatures have attempted to minimise the effect of popular items as a way of dealing with uninterested recommendations to users.

Nakatsuji *et al.* [23] studied the novel recommendations system to recommend the highly novel items in order to expand user interests. The system comprises of a method that modelled user interest based on items rated by the individual users and the taxonomy of items to measure similarity among users. This method uses an algorithm to adjust the similarity between users and incorporated it into CF-based methods, thereby; the recommendation of items is done to the active user by considering items with high novelty. The proposed method is evaluated on MovieLens and non-Japanese music artists' datasets, by using MAE and coverage metrics to determine the effectiveness of the approach. Results show that, this method can recommend items with high novelty and a better accuracy than the existing methods used in the study. However, the study do not consider user's interest on popular items while looking for items with high novelty, which may affect the accuracy of the recommendations provided to the active user.

Oh *et al.* [24] presented a novel recommendation system based on Personal Popularity Tendency (PPT) matching which recommends items to users. The system uses a method that recommends items similar to PPT of each individual user while improving the recommendation accuracy of the active user interest. The method proposed uses an algorithm to penalise popular items while balancing novelty and preference of the active user during prediction process. The efficiency of the proposed method is evaluated on MovieLens dataset using popularity, diversity and accuracy metrics. The results indicated that, PPT method performs better in terms of novelty and accuracy of the recommendations than the existing methods considered in this study. This work focus on the popularity distribution of the recommended items only, without considering the effect of popularity bias on the recommendations.

Zhao, Niu and Chen [25] proposed an opinion-based approach to deal with popularity bias in user-based CF recommender system. The system introduces a weighting function that replaces the original weighting function used in measuring the similarity between users. The function reduces the influence of popular items by considering the opinion of users on the target item, which is also included in the similarity function. The proposed approach is evaluated on MovieLens dataset using normalised discounted cumulative gain, coverage and coverage in long tail metrics. The results demonstrated that, the Opinion-based Weighting UserCF (OWUserCF) approach outperforms the comparative approaches in dealing with popular items by decreasing items with similar opinions and increasing items with dissimilar ones based on high accuracy of the recommendations obtained. This proposed approach considers the user-based CF algorithm only and it does not allow the user to specify the items that are not of interest.

Yang and Wang [26] studied the user-based and item-based CF system that recommends a list of items to a user. The system consists of a method that uses user active index and item popularity index to find neighbour of users (or items), and predictive score. The method used algorithm that reduces the influence of highly active users and highly popular items. The proposed system was evaluated on MovieLens and GOMO datasets using three evaluation metrics namely, recall, coverage and average popularity. The experimental results indicated that, the system provides recommendations with better quality while decreasing average popularity than the traditional methods. However, there is a tendency that the user may be interested in some of the popular items penalised before recommendations.

Fan *et al.* [27] proposed an improved CF-based punishment similarity function to alleviate the effect of popular items on similarity calculation. The function uses punishment function and Jaccard function to the adjusted cosine similarity to regulate the influence of popular items and the ratings conflict on common items during similarity computation. It penalise popular items contribution during similarity calculation in order to enable less popular items to be included in finding the real similarity between the users or items. The similarity function increases the accuracy of the users rating prediction as well as the efficiency of the prediction algorithm. However, it may not predict interested items to the active users due to the punishment function employed that highly penalise popular items to be recommended.

Almu *et al.* [28] presented a popularised similarity function (pop_sim) to provide effective recommendations to users. The pop_sim function introduces a modified punishment function to minimise the penalty on high popular items. The function also employs a popularity constraint which uses the items ratings threshold to increase the chances of choosing less popular items so as to obtain more similar items neighbours. Experimental studies conducted indicate that compared to the existing functions, the proposed pop_sim performs better in terms of reducing MAE and RMSE. Thus, improve the accuracy of the rating prediction. However, the function relies on a popularity constraint from the dataset which uses minimum rating threshold to identify the level of item popularity to be penalised instead of soliciting user input for the uninterested items. In turn, it leads to unwanted items recommendation to users.

2.5 Research Justification

Most of the existing CF methods [12, 3, 18, 21, 22, 23, 24, 26, 27, 28] investigated focused on the recommended items that are of interest to a user, without taking into consideration those items that the user did not evaluate because they are not of interest at all. These categories of recommended items that the user may not like can block the chances of recommending more items that are of interest to user by the recommender systems. Although there is a large body of work, addressing the problem of recommendations of interest, we noticed the following issues with the existing approaches:

(i)   The opinion of the user on the recommended items have not been modelled and utilised for reranking purposes.
(ii) There is no mechanism to appropriately deal with unwanted recommended items that the users may not like in the lists.

Based on the investigated literature the afore-mentioned issues have not been incorporated properly into the existing CF recommendation framework. This forms the basis and motivation of our research work.

3.   Preliminaries of CF Based Algorithm Construction

In real life setting, people tend to find out or get recommendation from other people with similar opinions about what they intend to acquire before making a choice. Collaborative filtering is built based on this idea to provide recommendations to individuals that share common interests. The basic assumption of CF algorithm is that, if there exist a list of *n* users represented as *{u₁, u₂, u₃, …, uₙ}* and also a list of *m* items represented as *{i₁, i₂, i₃, …, iₘ}*, then there is possibility that each user has a list of rated items that can be used for prediction purposes. To achieve that, those rated items are used as a basis to provide recommendations to some users who have not rated or seen those items in question before. Most of these algorithms collect users' explicit or implicit ratings to determine the predicted items for other users with common preferences. Consider Table 1 showing six students with their respective courses interests about the five computing science books.

Table 1. Students-Courses Preferences

Courses ⟋ Students	Programming	Networks	Compiler	Architecture	Security
Abba	1	0	0	1	1
Bashar	1	0	1	1	0
Chika	0	1	0	0	1
Dauda	1	1	1	0	1
Ebere	0	1	0	1	0
Fatima	0	0	1	0	0

From Table 1, assume that the number of student closest neighbours based on item of interest similarity is 2. Then we can observe that Abba likes "Programming", so Abba interest neighbours are Bashar and Dauda. For "Architecture" Abba interest neighbours are Bashar and Ebere. Likewise, for "Security" Abba interest neighbours are student Chika and Dauda. So, courses recommendation to Abba that he has not read can simply be achieved by considering courses maximum common preferences of Bashar and Dauda with Abba. Therefore, based on maximum common interest of the two students the course "Compiler" can be recommended to Abba. The recommendation procedure is also the same for the remaining students with similar interest neighbours.

The following are the steps involved in building Collaborative Filtering Algorithm as identified by Zhang [29]:

3.1 Rating Matrix Construction

The data which serves as basic source of recommendation can be expressed as m users and n items that is normally represented in the form of mxn matrix typically called user-item rating matrix as shown in Equation (1).

$$R_{mxn} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ r_{2,1} & r_{2,2} & \ldots & r_{2,n} \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ r_{m,1} & r_{m,2} & \cdots & r_{m,n} \end{bmatrix} \tag{1}$$

In equation (1) m means the number of items, n means the number of users and r_{ij} represents the rating of item i by user j. If the user j fails to rate item i, then $r_{ij} = 0$ and the algorithm tries to predict those unrated items.

3.2 Similarity Computation and Neighbours Selection

The existing CF methods any of the following similarity measure to find the similar neighbours for the target user:

(i) One of the measure used is Cosine Similarity [38] which uses the user-item rating matrix in Equation (1) to computes the similarity between two items, where each item is represented by a vector of user ratings as in Equation (2). The measure is good at computing the degree of closeness of the angle between these rating vectors. However, it is unable to cater for users' preferences with different rating scales.

$$sim(i,j) = \cos(i,j) = \frac{i.j}{|i|.|j|} \tag{2}$$

Where, i is represents the vector for i item = {$r_{1,i}$, $r_{2,i}$, $r_{3,i}$ ... $r_{m,i}$}, j is represents the vector for j item = {$r_{1,j}$, $r_{2,j}$, $r_{3,j}$ ... $r_{m,j}$} and the '.' represents dot product of the vector's components.

(ii) The Modified Cosine Similarity measure [39] is also used to take care with the problems of different users rating preferences that have been ignored in Equation (1). This measure represented as in Equation (3) considers subtracting users' average rating from the individual user rating during similarity computations. The measure is able to handle users' preferences with different rating scales. However, it cannot handle the common preference of co-rated users' ratings.

$$sim(i,j) = \frac{\sum_{c \in U}(R_{ci} - \overline{R_i})(R_{cj} - \overline{R_j})}{\sqrt{\sum_{c \varepsilon U}(R_{ci} - R_i)^2}\sqrt{\sum_{c \varepsilon U}(R_{cj} - R_j)^2}} \tag{3}$$

Where, R_{ci} represents the rating of user c on item i and $\overline{R_i}$ represents the average rating of all items rated by the user c. U is the set of users that rated both items i and j.

The Similar Neighbours Selection is performed after the similarity computations to select the item neighbours for rating score prediction purposes. The active item neighbours can be selected using the top-N approach [30] which finds a neighbour by defining the number of item neighbours. This approach is chosen because it is one of the effective methods of neighbours' selection. It identifies top-N neighbours of item by comparing the computed similarity values obtained using Equation (3) with that of the active user.

3.3 Prediction and Recommendation

The top neighbours selected are used for predicting item rating of the target user based on the result obtained in either Equation (2) or Equation (3). The prediction formula is shown in Equation (4).

$$P_{ui} = R_u + \frac{\sum_{a=1}^{n}(Rai - Ra) \text{ x sim(i,j)}}{\sum_{a=1}^{n}|sim(i,j)|} \tag{4}$$

Where, P_{ui} represents the predicted rating on item i for user u and $\overline{R_u}$ represents the user u average rating. The items with the highest predicted values are recommended to users.

4. Proposed CF Based Algorithm Construction

4.1 Problem Formulation

Much of the work done in the area of collaborative filtering recommenders [19, 23, 21, 31, 32, 18, 22] focused on the recommendation of items that are of user's interest. However, they still recommend items that are not of user's interest causing the user to lose interest in the recommender system. Thus, an attempt to deal with inaccurate recommendations is made by reducing the influence of popular items [24, 25, 26, 27, 28]. Despite the success of the existing approaches, they suffer from the following problems: First, the existing approaches do not solicit for user opinion of the items on the recommendation list; thereby causing the items that are not of interest to the user to be recommended. Second, those unwanted recommended items tend to be ignored by the existing approaches thereby causing inaccurate recommendations [33]. These pose significant challenges in providing good and acceptable recommendations to the active users. Therefore, this research work seeks to address the highlighted problems that affect the usefulness of CF based recommender systems.

4.2 CF Algorithm Based on Unwanted Recommendations Penalisation

Remember, both user-based and item-based CF methods find the similarities between items or users. Each of the algorithms also generates recommendations of items to users. These algorithms and their existing improvements are based on the CF architecture that normally ends the stage of top-N recommended list of items for the active user as depicted in Fig. 1 [34]. Similarly, the users of the recommender systems may or may not be interested in some of the items recommended. To alleviate the effect of unwanted recommendations the proposed architecture is extended as depicted in Fig. 1. This proposed CF based architecture incorporates two additional mechanisms consisting of relevance feedback to solicit for user preference and popularise similarity function to penalise the unwanted recommendations in order to improve subsequent recommendations (revised top-N of recommended items).

4.3 Exploiting Relevance Feedback

In information retrieval, Relevance Feedback (RF) enables the user to provide input on the initial retrieved results in order to identify its relevancy and irrelevancy. It has the following pseudo code [35]:

(i) The user issues a simple query.
(ii) The system returns an initial set of retrieval results.
(iii) The user marks some returned documents as relevant or non relevant.
(iv) The system computes a better representation of the information need based on the user feedback.
(v) The system displays a revised set of retrieval results.

In our proposed method, the following little modifications were done on above pseudo code as follows:

(i) The user will not provide a query as in the case of information retrieval but instead the user previous ratings on the items of interest are to be used as an implicit input to the system. Similarly, in (ii) the first top-N recommendations provided by the recommender system will be considered as the initial set of retrieval results. Therefore, (iii) – (v) will be incorporated in the existing recommender algorithm. In (iii) the user will mark any of the items recommended that are not of interest; (iv) the algorithm will capture the user feedback provided on the recommended item so as to computes a better recommendations that utilises user preference and (v) display a revised version of the top-N recommendations.

4.4 Incorporating Popularise Similarity Function

The popularised similarity function (pop_sim) introduces a modified punishment function to minimise the penalty on high popular items and also it employs a popularity constraint which uses the items ratings threshold to increase the chances of choosing less popular items so as to obtain more similar items neighbours for providing interesting recommendations to users [28]. It is presented as shown in Equation (5).

$$pop\_sim(i,j) = \begin{cases} sim(i,j) - log(I_{pop}), & if\ nR_j > T_{min} \\ sim(i,j), & otherwise \end{cases} \tag{5}$$

Where, $sim(i,j)$ is the modified cosine similarity function, I_{pop} is the modified popularity punishment function, $log(I_{pop})$ is subtracted from $sim(i,j)$ to moderately penalise high popular items and $(nR_j > T_{min})$ is the popularity constraint incorporated to increase the chances of less popular items to be predicted during recommendation process.

Therefore, the pop_sim is adapted with little modification and incorporated into the proposed method to deal with the unwanted items identified by the user. In this CF representation, the unwanted items represented by nUI_j will be obtained from the user input on the recommendations. This will be treated as new ratings given to the concerned items for the purpose of recalculating the similarity by using the pop_sim. The introduction of this function serves as a mechanism to help in minimising predicting unwanted items in the subsequent recommendations.

4.5 Unwanted Recommendations Penalisation Algorithm

The Pseudo code of the Unwanted Recommendation Penalisation Algorithm (URPA) that utilises the RF mechanism and pop_sim function to accept new set of ratings as input and output the predicted item rating is described as follows:

Algorithm 1: URPA

Input: Top-N List of recommended items - - containing new set of rating given by the user
Output: New set of recommended items
Begin:
1. for each item mark by the user in the *Top-N Lists* do
2.   determine the user evaluation_score of items, nUI_j
3.   capture all items nUI_j as input via RF
4.   if $nUI_j < UA_{limit}$ // test for the user evaluation score against the average limit
5.   **begin:**
6. $I_{pop} = 1 + \frac{1}{nUI_j}$ // *punish uninterested items*
7.   compute item similarity *mpop_sim(i, j)*   // *compute modified popularise similarity*
8. $mpop\_sim(i,j) \longleftarrow pop\_sim(i,j) - log(I_{pop})$
9.   **end:**
10.   else
11. $mpop\_sim(i,j) \longleftarrow pop\_sim(i,j)$
12.   compute new predicted ratings for each item neighbours as follows:

$$r_{t,i} = \overline{r_i} + \frac{\sum_{i \in neighbors}(R_{(t,i)} - \overline{R_t}) \times mpop\_sim(i,j)}{\sum_{i \in neighbors}|mpop\_sim(i,j)|}$$

13.   return new predicted ratings
14.   return items with highest ratings as the recommended lists
End:

The incorporated mechanisms in the CF based URPA Algorithm with penalty on unwanted recommendations will be evaluated based on user interest and interaction with the recommendations lists provided based on the following evaluation metrics:

(i) **Mean Absolute Error (MAE)**: This measure is used to evaluate the prediction quality of the proposed and the existing CF methods [36]. This accuracy metric is defined as in Equation (6). It is used in this work to measure the accuracy of the predicted rating scores of the modified method against actual ratings in the system.

$$MAE = \frac{\sum_{i=1}^{N}|(p_i - q_i)|}{N} \tag{6}$$

Where *N* is the number of the items recommended to the user in both the initial and subsequent recommendations.

(ii) **Precision**: This measure is used to evaluate capacity of a recommender system to show only useful and relevant items to users while trying to minimise the mixture of them with useless ones [37]. This metric is described mathematically as in Equation (7).

$$Precision = \frac{\#(Relevant\ Interested\ Recommended\ Items)}{\#(Total\ Recommended\ Items)} \tag{7}$$

This measure is used to capture the relevance of the recommended items to the user's interest for both the proposed and the existing CF methods.
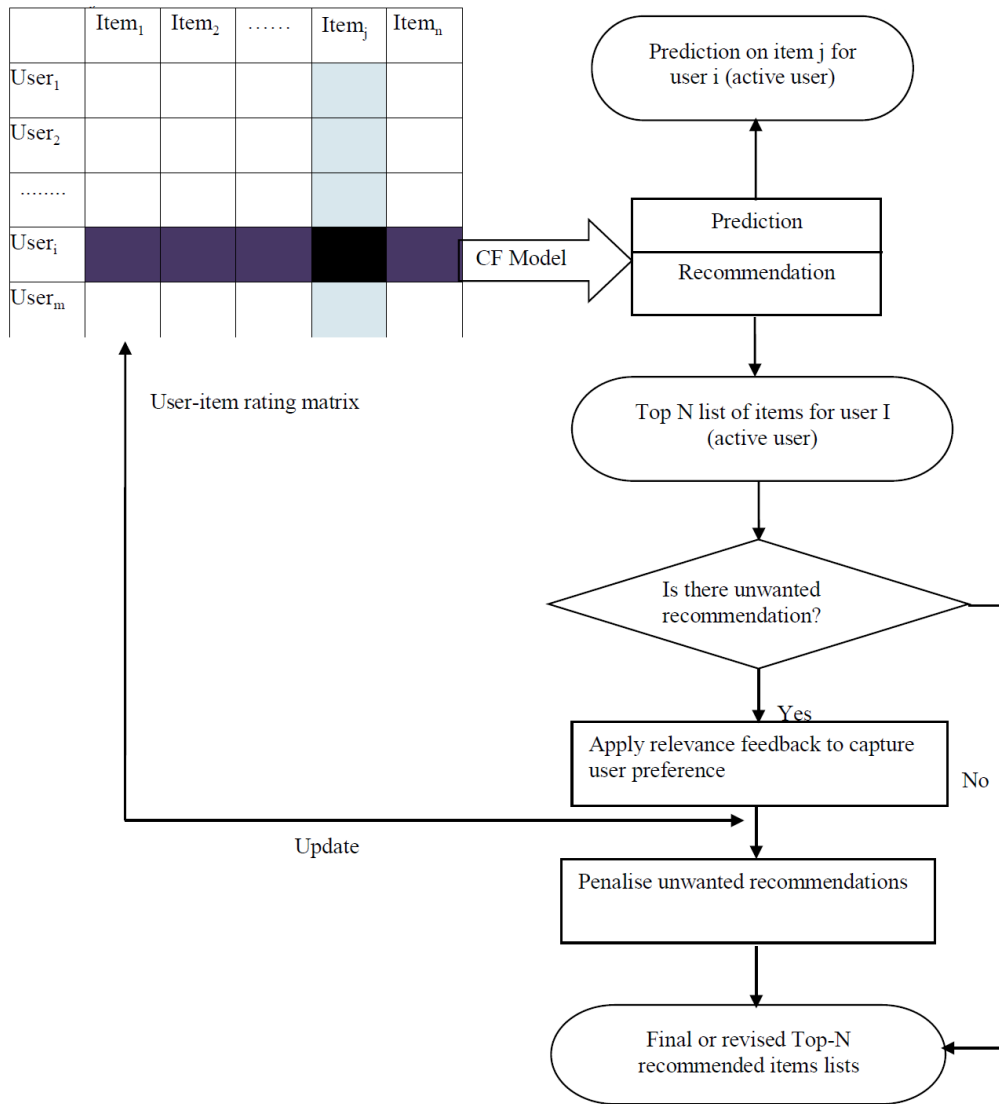
Fig. 1. Schematic Diagram of the CF Based Method

5. Conclusion and Future Work

Recommender systems identify and recommend items or contents of interest to the users from large collections of information. These systems fail to take the preference of the users on the recommendations provided since it can contain unwanted recommended lists. To alleviate these limitations, an improved CF-based method is presented that exploits the initial top-N recommendations for improving subsequent recommendations. The method uses algorithm that captures users' interest on the recommendations using relevance feedback to identify unwanted recommended items with a view to penalising those items by employing pop_sim function, thereby increasing the tendency of recommending useful interested items to the target users. This way, the recommendations are expected to become more relevant and accurate compared to the initial CF recommendations. This proposed unwanted recommendations penalisation algorithm is described in detail with the demonstrations of the incorporated improvements.

In the future work, an empirical study of the proposed CF method and incorporated mechanisms will be conducted to verify its accuracy and effectiveness in relation to some state-of-the-art method.

References

[1]   Resnick P., & Varian H. Recommender Systems. *Communications of the ACM*, 1997, 40(3):56-58.

[2]   Khusro, S., Ali, Z., and Ullah, I. Recommender Systems: Issues, Challenges and Research Opportunities. *Lecture Notes in Electrical Engineering* 376: Springer Science+Business Media Singapore 2016, pp. 1179-1189.

[3]   Zhong, Z., Xiao, B., and Duan, Y. Recommendation Algorithm with Center Distance-based Reranking. *Journal of Computational Information Systems*, 2014,10(23): 9957-9965.

[4]   Adomavicius, G., and Tuzhilin, A. Toward the Next Generation Recommender Systems: A Survey of the State-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 2005, 17(6): 734-749.

[5]   Pazzani, M., J., and Billsus, D. Content-Based Recommendation Systems. *The Adaptive Web-Springer*,2007, 4321:325-341.

[6]   Goldberg, D., Nichols, D., Oki, B., M., and Douglas, T. Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM*, 1992, 35(12):61-70.

[7]   Ekstrand, M. D., Riedl, J. T., and Konstan J. A. Collaborative Filtering Recommender Systems. *Foundations and trends in Human –Computer Interaction*, 2010, 4(2): 81-173.

[8]   Burke, R. Hybrid Recommender Systems. *User Modeling and User-Adapted Interaction*, 2002, 12(4):331-370.

[9]   Chen, W., Niu, Z., Zhao, X., and Li, Y. A Hybrid Recommendation Algorithm Adapted in e-Learning Environment. *Journal of World Wide Web*, 2012, 1-14.

[10]   Montaner, M., Lopez, B., De La Rosa, J. L. ATaxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review*, 2003, 19: 285-330.

[11]   Chen, Y., Wu, C., Xie, M., and Guo, X. Solving the Sparsity Problem in Recommender Systems using Association Retrieval. *Journal of Computers*, 2011, 6(9): 1896-1902.

[12]   Shi, Y., Larson, M., and Hanjalic, A. Connecting with the Collective: Self-contained Reranking for Collaborative Recommendation. *In Proceeding of ACM CMM'10*, 2010, Firenze, Italy.

[13]   Shi, Y., Larson, M., and Hanjalic, A. Reranking Collaborative Filtering with Multiple Self-contained Modalities. *In ECIR 2011*, LNCS, 2010, 6611: 699-703. Springer-Verlag.

[14]   Jannach, D., Lerche, L., and Gdaniec, M. Re-ranking Recommendations Based on Predicted Short-Term Interests- A Protocol and First Experiment. *In Proceeding of AAAI 2013 Workshop*.2013, Pp. 31-37.

[15]   Ziegler, C., McNee, S., M., Konstan, J., A., and Lausen, G. Improving Recommendation Lists Through Topic Diversification. *In Proceeding of International World Wide Web Conference Committee (IW3C)*, Chiba, Japan. 2005, Pp. 22-32.

[16]   Zhang, F. Improving Recommendation Lists Through Neighbor Diversification. *IEEE International Conference on Inteligent Computing and Intelligent Systems*. 2009, Pp. 222-225.

[17]   Yang, C., Ai, C., C., and Li, R., F. Neighbor Diversification-Based Collaborative Filtering for Improving Recommendation Lists. *IEEE International Conference on High Performance Computing and Communications (HPCC) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, 2013.

[18]   Harper, F., M., Xu, F., Kaur, H., Condiff, K., Chang, S., and Terveen, L. Putting Users in Control of their Recommendations. *In Proceeding of ACM RecSys'15*, 2015, Vienna, Austria.

[19]   Koren, Y. Collaborative Filtering with Temporal Dynamics. *In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France. 2009, Pp. 447-456.

[20]   Lathia, N. Temporal Diversity in Recommender Systems. *In Proceeding of ACM SIGIR'10*, Geneva, Switzerland, 2010.

[21]   Yin, L., Wang, Y., and Yu, Y. Collaborative Filtering via Temporal Euclidean Embedding. *In APWeb 2012*, LNCS, 2012, 7235: 413-520. Springer-Verlag.

[22]   Gasmi, I., Seridi-Bouchelaghem, H., Hocine, L., and Abdelkarim, B. Collaborative Filtering Recommendation Based on Dynamic Changes of User Interest. *Intelligent Decision Technologies*, 2015, 9: 271-281.

[23]   Nakatsuji, M., Fujiwara, Y., Fujimura, K., Tanaka, A., Ishida, T., and Uchiyama, T. Classical Music for Rock Fans: Novel Recommendations for Expanding User Interests. *In: Proceedings of the 19th ACM international conference on Information and knowledge management*, Canada, 2010, 949-958.

[24]   Oh, J., Park, S., Yu, H., Song, M., and Park, S. Novel Recommendation Based on Personal Popularity Tendency. *In: Proceedings of the 11th IEEE International Conference on Data Mining (ICDM)*, Vancouver,BC, Canada, 2011, 507-516.

[25]   Zhao, X., Niu, Z., and Chen, W. Opinion-Based Collaborative Filtering to Solve Popularity Bias in Recommender System. *International Conference on Database and Expert Systems Applications*, Springer-Heidelberg, 2013, 8056: 426-433.

[26]   Yang J. and Wang Z. An Improved Top-N Recommendation for Collaborative Filtering. In: Li Y., Xiang G., Lin H., Wang M. (eds) Social Media Processing, SMP 2016. *Communications in Computer and Information Science*, 2016, 669: 233-244. Springer, Singapore.

[27]   Fan, S., Yu, H., & Huang, H. An Improved Collaborative Filtering Recommendation Algorithm Based on Reliability. *Proceedings of the 3rd IEEE International Conference on Cloud Computing and Big Data Analysis* (pp. 45-51). Chengdu, China, 2018.

[28]   Almu, A., Roko, A., Mohammed, A., and Saidu, I. Popularised Similarity Function for Effective Collaborative Filtering Recommendations. *International Journal of Information Retrieval Research*, 2020, 10 (4): 34-47.

[29]   Zhang, D. Collaborative Filtering Recommendation Algorithm Based on User Interest Evolution. *Advances in MSEC*, 2011, 2(AISC 129): 279-283. Springer-Verlag.

[30]   Falk, K. Practical Recommender Systems. MEAP edition. Manning Publications Co, 2017.

[31]   Liu, Z. Collaborative Filtering Recommendation Algorithm Based on User Interests. *International Journal of u-and-e Service, Science and Technology*, 2015, 8(4): 311-320.

[32]   Toledo, R., Y., Mota, Y., C., and Martinez, L. Correcting Noisy Ratings in Collaborative Recommender Systems. *Knowledge-Based Systems*, 2015, 76: 96-108.

[33]   Xie, F., Chen, Z., Xu, H., Feng, X., and Hou, Q. TST: Threshold Based Similarity Transitivity Method in Collaborative Filtering with Cloud Computing. *Tsinghua Science and Technology*, 2013, 18(3): 318-327.

[34]   Isinkaye, F. O., Folajimi., Y. O., and Ojokoh, B. A. Recommendation Systems: Principles, Methods and Evaluation. *Egyptian Informatics Journal*, 2015, 16: 261-273.

[35]   Manning, C. D., Raghavan, P., and Schütze, H. An Introduction to Information Retrieval. England: Cambridge University Press, 2009. [online] Available from: http://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf [Accessed March 26, 2023]

[36]   Chen, Z., Jiang, Y., and Zhao, Y. A Collaborative Filtering Recommendation Algorithm Based on User Interest Change and Trust Evaluation. *International Journal of Digital Content Technology and its Applications*, 2010, 4(9):106-113.

[37]   Olmo, F., H., D., and Gaudioso, E. Evaluation of Recommender Systems: A New Approach. *Expert Systems with Applications*, 2008, 35:790–804.

[38]   Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. Analysis of Recommendation Algorithms for E-Commerce. *In Proceedings of the 2nd ACM conference on Electronic commerce*, 2000, pp. 158-167.

[39] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. Item-based Collaborative Filtering Recommendation Algorithms. *In Proceeding of the 10th international conference on World Wide Web, Hong Kong*, 2001, pp.285-295.

Authors' Profiles

Abba Almu received his PhD in Computer Science from Usmanu Danfodiyo University, Sokoto. Masters Degree in Computing: Information Engineering from Robert Gordon University, Aberdeen United Kingdom. He also received his B.Sc. (Hons) in Computer Science from Usmanu Danfodiyo University, Sokoto. He is currently a Senior Lecturer at the Computer Science Department of Usmanu Danfodiyo University, Sokoto Nigeria. His current research area is Intelligent Information Retrieval consisting of Recommender Systems, Question and Answering, Information Filtering, Social Network Analysis, etc.

Abubakar Roko is a Reader at the Computer Science Department, Faculty of Physical and Computing Sciences, Usmanu Danfodiyo Univesity Sokoto, Nigeria. He received his PhD from Univeristy Putra Malaysia in 2016, specializing in the field of XML Retrieval. Currently, his research interest is in the area of text data management and analysis, focusing in particular on query processing in Information retrieval, Recommender Systems, and Sentiment Analysis.