A Hybrid Framework for Improving Diversity and Long Tail Items in Recommendations

Pragati Agarwal, Rama Syamala Sreepada, and Bidyut Kr. Patra

National Institute of Technology Rourkela, Rourkela, Odisha, India.
{217cs1089,515cs1002,patrabk}@nitrkl.ac.in

Abstract. In today's information overloaded era, recommender system is a necessity and it is widely used in most of the domains of e-commerce. Over the years, recommender system is improved to meet the main purpose of achieving better user experience, where accuracy is considered as one of the important aspects in its design. However, other aspects such as diversity, long tail item recommendation, novelty and serendipity are equally important while providing recommendations to the users. Research to improve above mentioned aspects is limited. In this paper, we propose an efficient approach to improve diversity and long tail item recommendations. The experiments are conducted on two real world movie rating datasets namely, MovieLens and Netflix. Experimental analysis shows that the proposed method outperforms the state-of-the art approaches in recommending diverse and long tail items.

Keywords: Recommender System · Diversity · Collaborative Filtering · Long Tail Items · Hybrid Reranking Framework.

1 Introduction

With the advent of the World Wide Web, information has increased exponentially and so is the need of customized techniques to filter relevant and personalized information. Recommender system is an essential tool for providing customized information about products and services. It is widely used in various domains such as movies, music, books, research articles, electronic products, apparel, *etc.* Collaborative filtering (CF) approach is one of the popular approaches in the design of recommender systems [2]. It uses rating information of the items provided by the users in the past [1, 2]. CF techniques are further classified as memory based CF and model based CF [6]. Memory based CF, also known as neighborhood based CF relies on a simple intuition that an item might be interesting to an active user if the item is appreciated by a set of similar users or if the user has appreciated similar items in the past [1, 3, 4]. On the other hand, model based CF learns the features or patterns from rating information using machine learning algorithms [2]. Once the model is trained, it is used to predict the ratings of unrated items. Examples of model based collaborative filtering are matrix factorization [5], tendency based [2], *etc.*

All the aforementioned algorithms focus on improving the accuracy of recommender system. However, accuracy oriented approaches have two major limitations. These approaches recommend items that are very similar to the items consumed or rated by the user in the past, leading to lack of diversity in recommendations [8]. Further, these approaches are biased towards recommending popular items due to which non popular items (referred as long tail items) are ignored leading to loss in business [9]. Overcoming the above mentioned shortcomings result in an improvement in overall user experience and increase in business profits.

Over the last decade, there are a few approaches proposed in the literature that focus on improving diversity and long tail item recommendation. Adomavicius and Kwon proposed a ranking-based approach that improves diversity based on the statistics of recommended items such as reverse predicted rating, item popularity ranking, item absolute likeability, item relative likeability, *etc.* [8] . There are other variants of ranking-based approach proposed in the literature such as clustering-based approach (KRCF) [10] and graph-based reranking technique [12]. Clustering-based approach known as Knowledge Reuse Framework in Collaborative Filtering (KRCF) clusters predicted items and pick items from each cluster exploiting inter cluster dissimilarity [10]. Özge and Tevfik proposed two methods to improve diversity. First method is graph-based reranking technique which is applied after predicting the ratings of unrated items and the second method incorporates a diversity factor while training the model using matrix factorization approach [12]. Pareto-efficient multi objective ranking technique is proposed to maximize accuracy and diversity in order to recommend accurate and unpopular items [11]. Valcarce et al. proposed an approach referred as item-based relevance modelling (IRM) to improve long tail item recommendations [14]. IRM utilizes a probabilistic approach to build the relevance model on long tail items.

Though the above mentioned approaches improve diversity, these approaches do not integrate different aspects of diversity such as popularity of item, finding items dissimilar to users' past ratings *etc.* We address this issue by integrating these aspects and categorized them into different item classes in order to reap the maximum benefits. We propose a <u>Hy</u>brid <u>Re</u>ranking framework in <u>C</u>ollaborative <u>F</u>iltering (HyReCF) that utilizes various statistics of the recommended items to improve diversity and long tail item recommendations. Our contributions are summarized as follow.

- The predicted ratings obtained from collaborative filtering and the item classes are fused to generate *topN* recommendations for each user.
- We identify personalized dissimilar items to improve diversity.
- The proposed technique, HyReCF is extensively tested on two datasets (MovieLens and Netflix) and exhaustive study shows that HyReCF outperforms the state-of-the-art in terms of diversity and long tail recommendations.

The rest of the paper is organized as follows. Section 2 discusses proposed methodology in detail. Experimental results and analysis are provided in Section 3. Finally, we conclude our work in Section 4.

2 HyReCF: Proposed Hybrid Reranking Framework in Collaborative Filtering

In this section, we explain the proposed approach termed as Hybrid Reranking Framework in Collaborative Filtering (HyReCF) in detail. The proposed approach is carried out in two phases. In phrase I (referred as prediction phase), a list of predictions (I_u) is generated for each user applying a traditional collaborative filtering technique. Main idea of this approach is to deploy this framework with the existing collaborative recommender system to offer more diverse and long tail items to the users. Therefore, we can apply one of the popular traditional collaborative filtering approaches such as user-based CF, item-based CF, matrix factorization, $etc.$ in the first phase to generate prediction list I_u for a user.

In phase II (referred as reranking phase), significant factors such as popularity of item, finding items dissimilar to users' past ratings and good predicted items need to be introspected to attain diverse and long tail item recommendations. The predicted items are categorized into different classes to achieve maximum diversity as well as to improve long tail item recommendations. Several aspects such as relevance, popularity, and similarity of the items are considered while defining the classes. For each user, the prediction items list (I_u) obtained from phase I is categorized into three classes: 1) Unpopular Items, 2) Personalized Dissimilar Items, 3) Good Predicted Items.

Unpopular Items: In order to improve long tail item recommendations, the prediction list I_u is ranked in the increasing order of their popularity. The popularity of an item is defined as number of users who rated that item. This approach ensures that less popular items or long tail items are prominent in the user recommendations.

Personalized Dissimilar Items: A predicted item is said to be personalized dissimilar if it is different from all the items rated by the user in the past. Let $I_R = \{i_1, i_2, \ldots, i_r\}$ be the set of items rated by user u in the past. For an unrated item i and each item j in I_R, the dissimilarity can be computed using equation 1.

$$dis(i, j) = 1 - sim(i, j) \tag{1}$$

where $sim(i, j)$ is similarity between item i and item j. Similarity measures such as Adjusted Cosine, Cosine, Pearson Correlation, $etc.$ can be used to compute $sim(i, j)$. We compute aggregate dissimilarity between the item i and rated set I_R. Any one of the popular aggregate functions such as average, maximum or minimum can be used to compute aggregate dissimilarity. Once the aggregate dissimilarities of all the unrated items are computed, these items are ranked in the descending order of aggregate dissimilarities to obtain personalized dissimilarity items class.
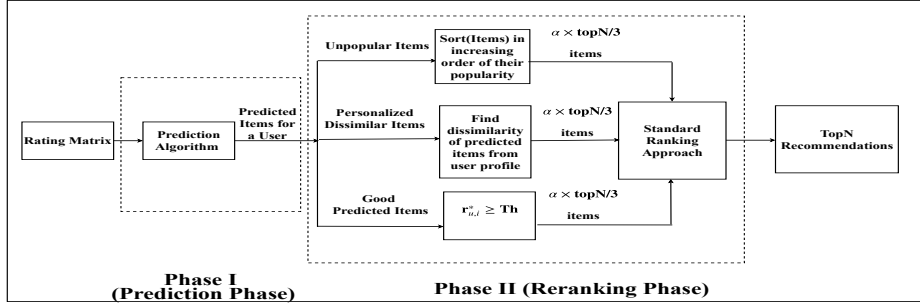
Figure 1. Hybrid Reranking Framework in Collaborative Filtering (HyReCF).

Good Predicted Items: A predicted item is considered as a good predicted item, if its predicted rating is greater than or equal to a certain threshold value (Th).

$$I_u^{Rel} = \{i \in I | \forall r_{u,i}^* \geq Th\} \tag{2}$$

where $r_{u,i}^*$ is the predicted rating of a user u for an item i and I is the set of items in the system. This class ensures accuracy in the recommendation list.

Most of the recommender systems provide $topN$ items, where $topN$ represents the number of items recommended to a user. HyReCF also recommends $topN$ items to each user. For each user, $\alpha \times topN/3$ items are selected from each of the above mentioned classes where α is an accuracy factor and holds an integer value. As the value of α increases accuracy also increases. Finally, standard ranking approach is applied on total of $\alpha \times topN$ items to generate a recommendation list I_u^{Rec} of topN items. This approach reranks the items in ascending order of their predicted rating values and picks the $topN$ items. The overview of the proposed approach is shown in Figure 1.

3 Experimental Evaluation

The details of experiments and result analysis are described in this section. In phase I of HyReCF, for predicting users' ratings of unrated items, we use matrix factorization (MF) [5] and item based collaborative filtering techniques [4]. To evaluate performance of proposed HyReCF framework, we implemented the traditional matrix factorization approach [5], item based collaborative filtering [4], ranking-based approach [8] and clustering-based approach (KRCF)[10]. All variants of ranking based technique and KRCF are executed. Out of which Item Absolute Likeability provides the best results [8]. So, results are reported with "Item Absolute Likeability" ranking approach. We used Ward's Minimum distance clustering approach in our experiments as it provides best results. Throughout all the experiments $topN$ recommendations for all algorithms is considered, where the value of N is set to 8.

Table 1. Dataset description

S.No.	Dataset	# Users ($|U|$)	# Head Items	# Tail Items	# Ratings (R)	K
1	MovieLens	5,278	2,000	836	972,165	93.51
2	Netflix	8,139	5,000	3,086	2,005,997	96.96

3.1 Dataset Description and Experimental Settings

All the implemented approaches are tested with two real world movie rating datasets namely, MovieLens and Netflix. Subsets of these datasets are created to ensure that each user has rated at least 20 items. Sparsity level of a dataset, denoted as K is the percentage of all missing ratings in a dataset. The statistics of working dataset are summarized in Table 1. In order to find long tail items, the item popularity graphs are plotted for MovieLens and Netflix datasets, as shown in Figure 2(a) and Figure 2(b), respectively. The items are sorted in descending order of their popularity. It can be noted that the items in the tail section of the plots received fewer ratings from the users, therefore considered as long tail items.

3.2 Evaluation Metrics

The following evaluation metrics are used to measure the performance of the proposed methodology.

Precision-in-TopN: It is defined as the ratio of recommended items that are actually relevant to all the recommended items. Let $Rec(u)$ be the set of $topN$ recommended items for user u, recommendation accuracy of a recommender system can be expressed as $precision\text{-}in\text{-}TopN = \sum_{u \epsilon U} |Rec(u) \cap Rel(u)| / \sum_{u \epsilon U} |Rec(u)|$, where $Rel(u)$ is the set of relevant items (ratings $\geq Th$) of user u in the test set.

Aggregate Diversity (AD): It is defined as number of distinct items that occur in the $topN$ list of all the users [8]. The AD is computed as $|\cup_{u \epsilon U} Rec(u)|$.

This metric gives an aggregate diversity achieved in the recommendations. However, it does not provide enough information regarding how many long tail items are recommended. To overcome this issue, we propose a new metric Long Tail denoted as LT.
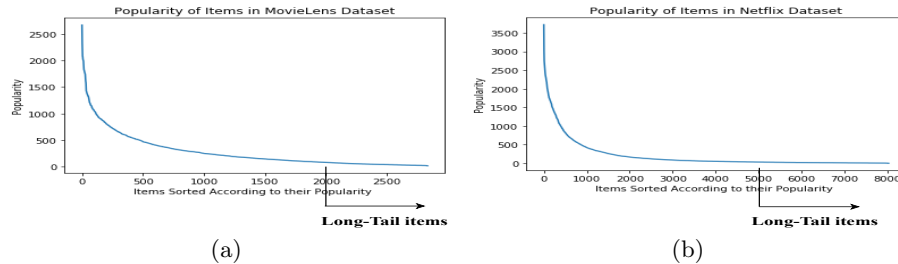

(a)   (b)

Figure 2. Item popularity graph for (a) MovieLens Dataset (b) Netflix Dataset.

Long Tail (LT): It is defined as number of times distinct long tail items occur in the *topN* list of all the users.

3.3 Experimental Results and Comparison

In order to compare improvement in diversity we keep precision of the proposed approach close to the standard approach. To achieve this we set $\alpha = 3$ and aggregate function to be maximum while creating personalized dissimilar items class. Comparative results are shown in Table 2 and 3 for MovieLens and Netflix datasets respectively. As observed from Tables 2 and 3 precision of the standard approaches is highest as their main focus is accuracy. However, HyReCF outperforms in terms of aggregate diversity and long tail item recommendations.

From Table 2, it can be noted that for prediction algorithm as matrix factorization approach, with 1% loss in precision, HyReCF approach achieves 227 items gain in aggregate diversity and 166 more long tail items are recommended with respect to traditional matrix factorization approach. In comparison to Ranking approach, with a little loss of 0.25% in precision, gain of 140 items in aggregate diversity and 122 items in long tail are obtained. Likewise, in item based CF, HyReCF earns a significant gain of 298 items in aggregate diversity and 216 in long tail with a loss of 1.1% in precision compared with traditional item based CF approach. A profit of 250 items in aggregate diversity and 196 items in long tail is attained with a loss of 0.9% in precision in comparison to Ranking approach.

Comparative results on Netflix dataset are reported in Table 3. For prediction algorithm as matrix factorization approach, we can observe that with a 1% loss in precision we attain 723 items gain in AD and 435 items in LT recommendation, while compared with standard matrix factorization. However, when compared to ranking approach, we achieve gain of 205 items in AD and 232 items in long tail recommendation with a loss of 0.41%. Similar observations can be drawn for item based CF approach.

IRM is an approach that provides long tail item recommendation [14]. In order to observe the effectiveness of long tail item recommendations, we compare HyReCF with IRM. Table 4 and 5 report the results in terms of precision loss and gain in long tail item recommendation for IRM and HyReCF approach with respect to standard matrix factorization approach.

Table 2. Experimental Results on MovieLens Dataset.

Prediction Algorithm →		Matrix Factorization			Item Based CF		
S.No.	Recommendation Algorithm ↓	Precision (TopN)	AD	LT	Precision (TopN)	AD	LT
1	Standard Approach	**0.8326**	1,967	287	**0.8325**	1,660	202
2	KRCF Approach	0.7959	2,076	344	0.8290	1,698	246
3	Ranking Approach	0.8250	2,054	331	0.8304	1,708	222
4	HyReCF Approach	0.8225	**2,194**	**453**	0.8206	**1,958**	**418**

Table 3. Experimental Results on Netflix Dataset.

Prediction Algorithm →		Matrix Factorization			Item Based CF		
S.No.	Recommendation Algorithm ↓	Precision (TopN)	AD	LT	Precision (TopN)	AD	LT
1	Standard Approach	**0.8323**	4,061	569	0.8188	4,023	730
2	KRCF Approach	0.7887	4,203	510	0.8137	3,901	595
3	Ranking Approach	0.8247	4,579	772	**0.8256**	4,173	918
4	HyReCF Approach	0.8206	**4,784**	**1,004**	0.8006	**4,919**	**1,229**

From Table 4, it can be noted that with 0.100 precision loss, IRM gains 51 items whereas 273 LT items gain is achieved in HyReCF which is a significant improvement. IRM performs adversely in terms of aggregate diversity. With the loss in precision there is a loss in diversity instead of having gain whereas HyReCF shows a significant improvement in diversity with a small drop in accuracy. Table 5 reports result on Netflix dataset. With a loss of 5%, IRM gains 387 long tail items and HyReCF gains 590 long tail items. It can be observed that HyReCF outperforms IRM in both diversity and long tail item recommendation.

The proposed approach, HyReCF is further analyzed on varying precision using accuracy factor α. Figure 3(a) and 3(b) depict the performance of HyReCF on MovieLens and Netflix datasets respectively. From Figure 3(a), in matrix factorization approach when $\alpha = 1$, precision is 0.6324 and aggregate diversity is 2817 items. At $\alpha = 2$, precision is 0.7776 and aggregate diversity is 2601. At $\alpha = 3$, precision is 0.8225 and aggregate diversity is 2194. It can be observed from the Figure 3(a) and 3(b), as precision decreases, we obtain significant improvement in diversity.

4   Conclusion

In this paper, we proposed Hybrid Reranking Framework in Collaborative Filtering (HyReCF), a technique to improve diversity and long tail item recommendations. Various statistics of the rating information and recommended items are integrated to improve diversity with a small drop in accuracy. The proposed approach provides a significant improvement in the diversity as compared to state-of-the-art. The long tail items are more prominent in the proposed framework than state-of-the-art which makes system interesting for the users as well as increases profits for the business organizations.

Table 4. Experimental Results on MovieLens Dataset.

Precision (Standard)	Precision Loss	IRM		HyReCF Approach	
		AD Gain	LT Gain	AD Gain	LT Gain
	0.001	-3	0	4	2
	0.010	-31	5	42	27
0.8326	0.025	-79	12	106	68
	0.050	-158	25	212	136
	0.100	-316	51	424	273
Standard	0.0000	AD=1,967		LT=287	

Table 5. Experimental Results on Netflix Dataset.

Precision (Standard)	Precision Loss	IRM		HyReCF Approach	
		AD Gain	LT Gain	AD Gain	LT Gain
	0.001	-3	7	19	11
	0.010	-30	77	192	118
0.8323	0.025	-77	193	481	295
	0.050	-154	387	963	590
	0.100	-308	775	1,926	1,181
Standard	0.0000	AD=4,061		LT=569	

Acknowledgement

References

1. Bobadilla, J., Ortega, F., Hernando, A. and Gutiérrez, A.: Recommender systems survey. Knowledge Based Systems **46**. 109–132 (2013).
2. Cacheda, F., Carneiro, V., Fernandez, D. and Formoso, V.: Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems. ACM Transactions on the Web **5**(1). Article 2 (2011).
3. Desrosiers, J.C. and Karypis, G.: A comprehensive survey of neighborhood-based recommendation methods. Recommender Systems Handbook. pp. 107–144 (2011).
4. Sarwar, B., Karypis, G., Konstan, J. and Riedl, J.: Item-based collaborative filtering recommendation algorithms. In : Proceedings of the 10th International Conference on World Wide Web. pp. 285–295 (2001).
5. Koren, Y., Bell, R. and Volinsky, C.: Matrix Factorization Techniques for Recommender Systems. Computer **42**(8). 30–37 (2009).
6. Koren, Y.: Factor in the neighbors: scalable and accurate collaborative filtering. ACM Trans. Knowl. Discov. Data **4**(1). 5–53 (2010).
7. Matevž, K. and Tomaž, P.: Diversity in recommender systems–A survey. Knowledge-Based Systems **123**. 154–162 (2017).
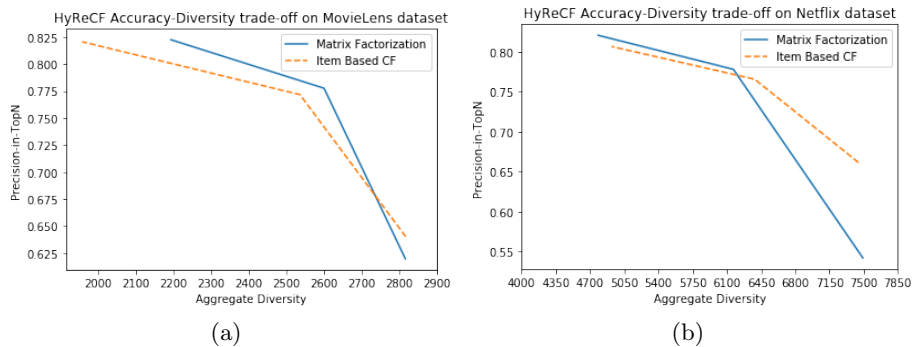
Figure 3. Performance of HyReCF on (a) MovieLens Dataset (b) Netflix Dataset.

8. Adomavicius, G. and Kwon, Y.: Improving Aggregate Recommendation Diversity Using Ranking-Based Techniques. IEEE Transactions on Knowledge and Data Engineering **24**(5). 896–911 (2012).

9. Harald, S.: Item popularity and recommendation accuracy. In: Proceedings of the fifth ACM conference on Recommender systems. pp. 125–132 (2011).

10. Pathak, A. and Patra, B. K.: A knowledge reuse framework for improving novelty and diversity in recommendations. In: Proceedings of the Second ACM IKDD Conference on Data Sciences. pp. 11–19 (2015).

11. Shanfeng, W., Maoguo, G., Haoliang, L. and Junwei, Y.: Multi-objective optimization for long tail recommendation. Knowledge-Based Systems **104**. 145–155 (2016).

12. Özge, K. M. and Tevfik, A.: Effective methods for increasing aggregate diversity in recommender systems. Knowledge and Information Systems **56**(2). 355–372 (2018).

13. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In : Proceeding of KDD Cup Workshop at 13th ACM International Conference on Knowledge Discovery and Data Mining. pp. 39–42 (2007).

14. Valcarce, D., Parapar, J. and Barreiro, A.: Item-based relevance modelling of recommendations for getting rid of long tail products. Knowledge-Based Systems **103**. 41–51 (2016).