International Journal of Engineering Trends and Technology Volume 69 Issue 12, 13-20, December, 2021
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V69I12P203

Original Article

An Improved Personalized Collaborative Filtering Recommendation Based on Items Diversity

Masoud Ghorbanian[1], Noor Hafizah Binti Hassan[2], Mohd Shahidan Bin Abdullah[3]

[1,2,3]*Razak Faculty of Technology and Informatics, UTM, Malaysia*

[1]gmasoud4@graduate.utm.my, [2]noorhafzah.kl@utm.my ,[3]mshahidan@utm.my

***Abstract** - With the rapid information growth and the flood of data, users usually face a huge number of items. Therefore, finding desirable items is a challenge for users. On the other hand, users' behavior is changing, and these changes are creating pressure on the profits of the popular items, while at the same time, the new or previously unknown items are being ignored due to lack of coverage in recommendations. In other words, the existing algorithms have not fully covered the diversity and coverage of the items in the recommendation lists. In this paper, a personalized recommendation algorithm is proposed to improve the diversity and coverage of the items. The contributions are as follows: a method is proposed to calculate the user's diversity level in order to improve items' diversity and coverage in recommendations by performing a collaborative filtering method and k-means cluster based on genre. Also, intra-list diversity, coverage, and accuracy results are investigated between the proposed recommendation algorithm and other similar recommendation algorithms. The results of the experiment show that our proposed recommendation algorithm improves the diversity and coverage of the items significantly while still preserving the users' interest.*

Keywords - *Accuracy, Collaborative filtering, Coverage, Diversity, the Recommendation algorithm*

I. INTRODUCTION

Over the past few years, 90% of the world's data has been generated, and with this huge load of information, users have more difficulties finding the items they need [1]. Many researchers have discussed and introduced search engines and recommendation systems as a way to address this problem. By using search engines, users should know some keywords related to what they are looking for. While in Recommendation systems, the solution is different. Based on each individual user, it can recommend the items that are relevant to the user's interest. Therefore, it assists users in finding their desired items more effectively. Generally, it must be able to recommend different items for each user based on the user's interest. By filtering and generating a list of items that have the highest probability to meet user's needs instead of a huge number of items offered by an online store, not only recommendation systems assist users in handling information overload but also minimize the time that each user needs to spend in order to find the desired items and improve the process of users' decision making. Moreover, recommendation systems can help merchants to increase their sales by changing explorers to buyers and creating consumer loyalty [2]. Therefore, further studies on a user-oriented approach in recommendation algorithms are required. The focus of this type of approach is on users' satisfaction.

Generally, every recommender system has three steps: collecting users' preferences as input, calculating the recommendation by appropriate techniques, and eventually giving the recommendation outcomes to the users [3]. When the goal of the recommendation algorithm is only to improve accuracy, it usually focuses on the second step, and the third step is where the usefulness and quality of recommendations such as diversity can be improved in a recommendation algorithm.

This research contributes towards improving the item's diversity and coverage in recommendations via collaborative filtering method and k-means cluster based on the genre to achieve higher diversity in the recommended list while still relevant to users' preferences. The proposed algorithm in this research is implemented, and its result is evaluated based on Intra-List diversity and coverage by using the MovieLens dataset. Furthermore, the accuracy of the proposed recommendation algorithm is investigated and analyzed in comparison to other related works. Thus, this paper is organized as follows; First, related works are discussed in the next section. Then in section III, the details of the proposed personalized recommendation algorithm are described, followed by the analysis of the experimental results in the fourth section. Finally, the conclusion and future works are presented in section V.

II. RELATED WORKS

The studies on personalized recommendation algorithms toward diversity have been getting a lot of attention [15], [16], [24]. Many researchers focused on diversity in recommendation to provide more diverse results [9], [10], [17], [18], [19], [20], [21]. In this section, some of the recent studies that are related to this research are discussed. A study by Eskandanian, Mobasher, and Burke (2017) proposed a personalized approach to diversify items by using collaborative filtering on a different group of users according to the diversity level of users' profiles. First, they clustered users which had similar interests based on movie genres that users had rated. Then, they used standard collaborative filtering on each cluster to generate the recommendation lists. Although their results proved satisfactory diversity and accuracy on Yelp and MovieLens datasets, it is not effective in generating a useful recommendation for the new items. Furthermore, their user's clustering approach is not efficient for the new users since it cannot determine the diversity level of the new users' profiles.

Another study by Mehrjoo and Hajipour (2019) proposed an optimal diversity approach for recommendations based on the users' desirable diversity. First, they determined users' desirable diversity by calculating the difference between the diversity of what users rated in the past and intra-list diversity of the recommended list. Then, they used a similarity matrix and standard collaborative filtering to predict items. Finally, they selected the items with the highest prediction value to recommend to the user. The results of this study showed that the proposed approach improved the personal diversity in recommendation lists on the MovieLens dataset, but it did not address the cold-start problem.

In comparison to the related works, the goal of this paper is not only to improve diversity in recommendations based on users' desirable diversity but also to tackle the cold-start problem.

III. THE PROPOSED ALGORITHM

The recommendation algorithm is responsible for generating a recommendation list when users search or select an item. Then, the generated recommendation list can be personalized based on the user's profile. In the first step of the proposed algorithm, item-based collaborating filtering is used to calculate items similarities. Next, the output of the previous step is processed to generate a top-n list for each user. In the next step, items are clustered based on genre using k-means cluster to rerank the items based on the top-n list and user's interests with its diversity level. To rank the items in the top-n list, it populates those items which are not in the same cluster according to the user's diversity level to assure the variety of items based on user interest. The proposed algorithm considers factors such as item popularity, user profile, and diversity level when ranking the items to considerably increase diversity in the recommendation list. The higher diversity level brings the less popular items on top, where popularity is determined by the number of ratings that each item has. The framework of the proposed algorithm is illustrated in Figure 1.

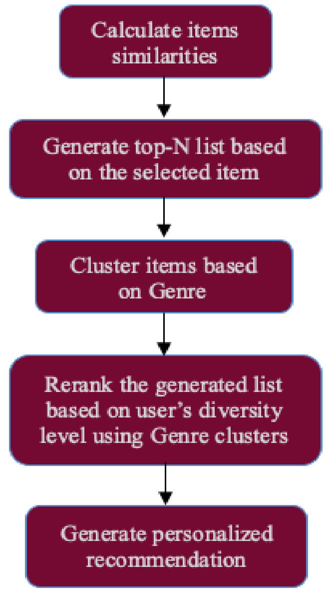

Fig. 1 Proposed Algorithm Framework

***A. Similarities Between Items**:* For any item in the dataset, the similarity is determined by a similarity measure. The values of similarities are processed to predict user-item pair ratings that are not presented in the dataset. There are a wide variety of mathematical formulas to measure the similarity for any two items [4]. In this paper, one of the most commonly utilized similarity measures named Pearson correlation is used to examine the effect of user's diversity level on the recommendation list. Using this similarity measure improves the score accuracy, especially when data is not normalized. It measures similarity between two non-zero variables, X and Y. Its value ranges from +1 to -1, with 1 indicating the highest positive linear correlation, 0 indicating no linear correlation, and -1 indicating maximum negative linear correlation. The mathematical equation is presented as follows [8]:

$$P(x, y) = \frac{n(\sum xy) - (\sum x \sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \quad (1)$$

Where:
N = the number of pairs of scores
Σxy = the sum of the products of paired scores
Σx = the sum of x scores
Σy = the sum of y scores
Σx^2 = the sum of squared x scores
Σy^2 = the sum of squared y scores

For example, if the result is closer to 1 for two items, these items have directly corresponded with each other. If it is closer to -1, these items are inversely proportional to each other. If it is lower or closer to 0, those items don't have a strong dependency with respect to each other.

Figure 2 represents the high-level pseudocode that analyses the items rated by users and processes them to discover the similarity of other items to the selected item. By analyzing the similarities between the items, it can find out what user u might be interested in. This analysis is based on the high rated similar items, which will be a candidate list of related items for user u, who has a high probability to like these items.

```
1.   for all item i which user u has selected, do
2.          for all item j which user u has no preference, do
3.                Compute similarities s between item i and
     item j
          end for
     end for
4. order all items by similarity s descending
5. return top-n items
```

Fig. 2 Pseudocode to Generate top-n Items

B. Cluster Items based on Genre*:* In this section, a clustering approach segregates items that have the same level of diversity based on genre. To segment the items based on their genre, k-means clustering is applied, which is a partition-based clustering to determine the K data groups. Therefore, the elbow method is used to specify the right number of clusters. The idea of this method is to run k-means clustering on the dataset for a range of values of k and then calculate the sum of squared errors (SSE) for each value of k. For each point, the error is the distance to the nearest cluster [25]. To get SSE, these errors are squared and then sum them as shown in the following equation [5].

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(c_i, x) \tag{2}$$

Where dist is the distance between the data and the Cluster center, the clustering process can be depicted using pseudo-code, which requires two inputs called k (number of clusters) and a set of points (x_1, x_2, …., x_n) as shown in Figure 3 [14].

```
1.   Select k value, which is the number of clusters you
     want to identify.
2.   Select k data points as centroids (c1, c2,…,ck)
     randomly from the vector space.
3.   Repeat until convergence :
          3.1  for each data point xi:
                 3.1.1 find the nearest centroid cj using
                   distance.
                 3.1.2 assign xi to that nearest cluster j.
          3.2  for each cluster, find the new centroid, which
          is the mean of all the xi points assigned to that
          cluster j.
4.   Terminate when none of the cluster assignments
     change.
```

Fig. 3 Pseudocode for K-Means Clustering

3.3 Rerank Items based on User's Diversity Level and Genre Clusters: There are several ranking methods that affect how various items are recommended which can improve the diversity in a recommended list. The overall idea of ordering information is not something new, but it's significant when it comes to information retrieval as it can minimize duplication and increase the variety of recommendation results by reranking them. Some of the well-known ranking methods are the top-n ranking method, reverse ranking method, and random-based ranking method.

The recommendation list provided by applying one of the ranking methods mentioned has several different and diverse items (which may not be in most highly predicted necessarily) to recommend to the users. In this way, users can get recommended more distinguished, long-tail, less frequently recommended items that may not be as widely popular but can still be very relevant to the user. Therefore, reranking the candidate items can improve the recommendation diversity significantly. For instance, Table 1 shows the result of a recommendation using the standard top-n ranking method.

Table 1: Recommendation List Using Top-n Ranking Method

Standard recommendation			Proposed recommendation		
ID	**Rank**	**Cluster**	**ID**	**Rank**	**Cluster**
121	1	13	121	1	13
546	2	18	546	2	18
117	3	17	117	3	17
597	4	18	25	4	12
118	5	17	125	5	10

As it is shown in Table 1, the proposed recommendation replaced those items that have repeated genre clusters based on the user's diversity level. For example, when a user's diversity level has a maximum value of 100, it means the repeated genre is not allowed. Therefore, the item with an ID of 597 in the standard recommendation is replaced by another item with an ID of 25, which has a different genre cluster. The value of diversity level is asked from users explicitly, which is a percentage value from 0 to 100. Then, the number of genres that can be repeated is calculated by Equation 3.

$$\text{RepeatedGenre} = n - \frac{n * dl}{100} \tag{3}$$

Where n is the number of items in the recommendation list and dl is the user's diversity level. Figure 4 shows the pseudocode for the proposed recommendation.

1. Get dl value which is the user's diversity level.
2. Get n value which shows a number of items in the recommendation list.
3. Generate candidate lists using item-based collaborative filtering as described in section 3.1.
4. Calculate RepeatedGenre based on dl and n.
5. Repeat until the recommendation list reach n items
   - 5.1. find all the items that are not covered by Candidate list.
   - 5.2. select the recommended item.
   - 5.3. if it is in the genre cluster that already used, then select a random item from the List at step 5.1 to recommend.
   - 5.4. If RepeatedGenrevalue is greater than zero then ignore step 5.3 and select the item at Step 5.2 to recommend (RepeatedGenre--).
6. Terminate when n items are selected.

Fig. 4 Pseudocode for the Proposed Recommendation

IV. RESULTS AND DISCUSSION

A set of experiments that demonstrate the effectiveness of the proposed algorithm is described in this section. Our recommendation algorithm is implemented using Apache Mahout [22] and MongoDB on a 2.40 GHz Intel(R) Core i3 processor with 4 GB RAM. HTML is used to provide information to users, and Java is used to communicate with the database and implement the recommendation algorithm.

A. Dataset: To carry out experiments in this research, a dataset named MovieLens1m is selected, which is publicly available. This dataset was put together by the Group lens research group at Minnesota University [6]. More details about this dataset are provided in this section. Since the experiment in this research relies on items' categories, the Movie genres have been utilized. The MovieLens1m data set consists of 3,706 movies, and the total number of ratings is 1,000,209, which are all rated by 6,040 users. The summary statistics of this dataset are represented in Table 2.

Table 2: Details of MovieLens Dataset

	ML 1M
Date	4/2000 - 2/2003
Rating Scale	1–5 stars
Users	6,040
Movies	3,706
Genres	18
Ratings	1,000,209

Each rate represented in <userId::movieId::rating::timestamp> form. The scale of ratings is numerical, which is from 1 to 5. All the rates in this dataset belong to those who registered in MovieLens and had more than 20 rate entries. Furthermore, some users' information such as gender, age and occupation are captured in this dataset in <userId::gender::age::occupation::zip-code> form. The information about movies are provided in form of <movieId::title::genres> as presented in Table 3.

Table 3: An Example of Movies Information

6::Heat (1995)::Action|Crime|Thriller
7::Sabrina (1995)::Comedy|Romance
8::Tom and Huck (1995)::Adventure|Childrens

As Table 3 shown, the format of genres is not numeric. However, the k-means cluster requires numeric attributes. Thus, there is a need to change the format into a set of binary attributes where only 1 means that the movie has that attribute. An example of this format is shown in Table 4.

The reason for choosing this dataset as the main dataset for this research is that it is well known in the literature for testing algorithms in recommender systems. In other words, the strength of this dataset is its widespread use, which enables researchers to compare their results with other algorithms.

Table 4: An Example of Movies Information with Numeric Attributes

Id	**Title**	**Action**	**Adventure**	**Animation**	**Children**	**Comedy**	**Crime**	**Documentary**	**Drama**	**Fantasy**	**Film-Noir**	**Horror**	**Musical**	**Mystery**	**Romance**	**Sci-Fi**	**Thriller**	**War**	**Western**
6	Heat	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0
7	Sabrina	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
8	Tom and Huck	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

***B. Genre Clusters**:* To group the items based on genre, a k-means cluster is used on the selected dataset. The sum of squared errors (SSE) should be calculated for each value of k to find the best number of clusters. The result of calculating SSE is presented in Table 5.

Table 5:SSE for Different Values of k (Number of Clusters)

Number of the cluster(k)	SSE	Number of the cluster(k)	SSE
5	1664077.12	13	692278.01
6	1255502.03	14	676047.96
7	983365.31	15	703437.20
8	1078734.85	16	597685.86
9	789228.51	17	603331.58
10	801675.17	18	580813.96
11	663857.32	19	458436.30
12	743042.32	20	328043.92

Using these values to make a line chart helps to determine the best k. Usually, this line chart appears like an arm. There is a point on the arm called the "elbow" that has the best k value. As shown in Figure 5, k=18 is the correct number of clusters.

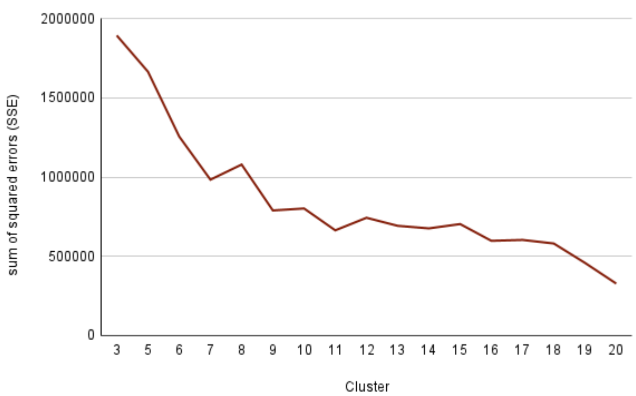

Fig. 5SSE of Items Clusters in MovieLens based on Genre

Figure 6 shows the number of movies per cluster in the MovieLens dataset. There are 3,706 movies in total in these clusters.

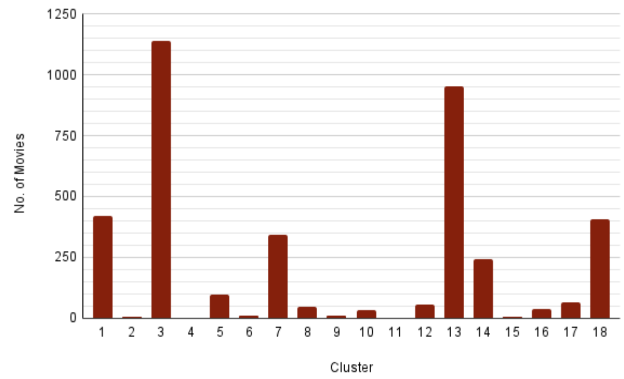

Fig. 6Items Clusters in Movie Lens based on Genre

4.3 Evaluation: To carry on with the proposed algorithm, item-based collaborative filtering is used to generate a candidate list. Then, the k-means cluster based on genre, which is calculated in the previous step, has been applied to rank the items and replaced those items that have duplicated genre clusters based on the user's diversity level. The results are presented in Table 6 based on the measure of similarity between recommended items which is called Intra-List Similarity (ILS). The ILS equation can calculate the similarity between any two items (i_j, i_k) where n is the number of items in the recommended list, as it is presented in Equation 4 [7], [13].

$$\mathrm{ILS_{user}} = \frac{1}{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \mathrm{sim}(\mathrm{i_j}, \mathrm{i_k}) \quad (4)$$

Table 6 demonstrates intra-list similarity for a recommended list of 20 items. As the percentage of the diversity level increases, the Intra-List similarity decreases, which shows higher diversity.

Table 6: Detection of Intra-List Similarity

Diversity Level (%)	Intra-List Similarity
0	137
10	137
20	137
30	137
40	137
50	113
60	97
70	83
80	72
90	68
100	57

According to the intra-list similarity, which is calculated in Table 6, intra-list diversity can be calculated by Equation 5 [13].

$$\text{ILD} = 1 - \text{ILS}/\text{N} \tag{5}$$

Figure 7 represents the experiment results for different simulated algorithms such as item-based collaborative filtering (IBCF) [11], diverse clustering (Div-Clust) [9], personal, collaborative filtering (PCF) [10], and the proposed algorithm. The results show that the proposed algorithm has significantly improved the intra-list diversity in the recommended list where maximum diversity is 1, and the minimum is 0. It has improved diversity by 12% in comparison to PCF and 20% compared to the baseline algorithm. The improvement is due to reranking items based on the user's diversity level and genre clusters which are introduced in the proposed recommendation algorithm.

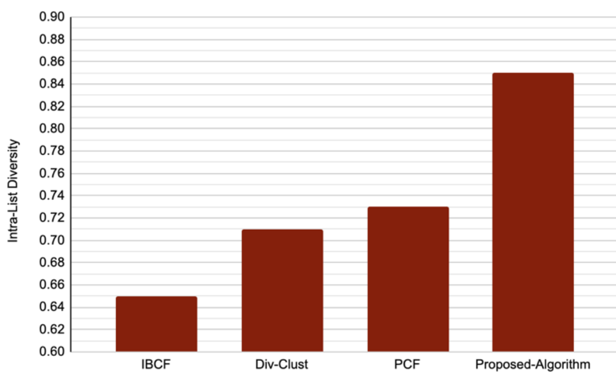

Fig. 7Intra-List Diversity Results

To evaluate the effectiveness of our proposed algorithm on the cold-start problem, one of the well-known metrics called coverage is used to carry on the experiment. It calculates the number of distinct items in the recommendation list (n) over the total number of items (N) [23], as is shown below in Equation 6.

$$\text{Coverage}_{\text{item}} = \frac{\text{n}}{\text{N}} \tag{6}$$

The coverage result is represented in Figure 8. The maximum coverage value is 1, which means all items in the dataset were recommended for at least one time. A higher coverage indicates that the recommendation algorithm can recommend a greater number of items. The results show that the proposed algorithm has the highest coverage compared to the related works. Therefore, it proves that it has a better performance to recommend the new items.

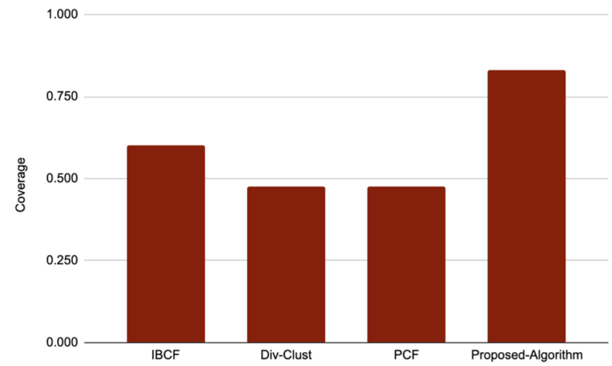

Fig. 8Coverage Results

As it is mentioned in Section 1, diversity plays an important role in improving users' satisfaction. However, accuracy is still the main target of any recommendation algorithm. Therefore, the mean absolute error (MAE) metric is used to measure the accuracy of recommendations, and it is calculated by Equation 7 [12].

$$\text{MAE} = \frac{\sum_{i=1}^{N} |r_{ui} - \bar{r}_{ui}|}{N} \tag{7}$$

Where rui is the actual rate value of the user for item i andrui is the predicted rate value. The lower value of MAE is better, which means the predictions are closer to the actual values. Figure 9 indicates the MAE results.

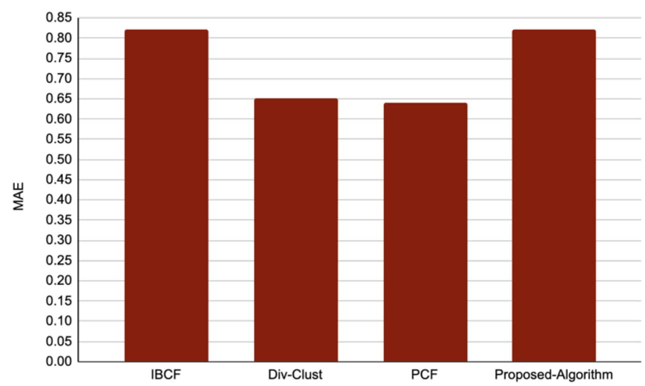

Fig. 9MAE Results

Another common evaluation metric is the root mean square error (RMSE), which is used to evaluate the accuracy of the recommendations. Unlike MAE, RMSE penalizes large errors, and it is calculated by using Equation 8 [12].

$$\text{RMSE} = \frac{\sum_{i=1}^{N} \sqrt{(r_{ui} - \bar{r}_{ui})^2}}{N} \tag{8}$$

The lower value RMSE and MAE have, the better accuracy of a recommendation algorithm has. Figure 10 shows the root mean square error (RMSE) results.

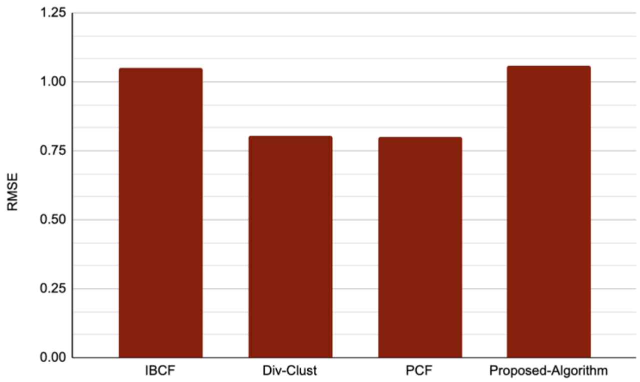

Fig. 10 RMSE Results

According to the results in Figures 7, 8, 9, and 10, which represent the experiment results for intra-list diversity, coverage, MAE, and RMSE, respectively, we noticed that the proposed algorithm has acceptable accuracy with the highest intra-list diversity and coverage among the investigated algorithms. Thus, the results prove that the proposed personalized recommendation algorithm increases the diversity of items and coverage in the recommended list.

V. CONCLUSION

In this work, a novel approach for items diversity in a collaborative filtering recommendation algorithm is proposed to improve diversity and coverage in recommendations which leads to more opportunities to create new offerings. To achieve this goal, diversity level and a number of repeated genres are introduced, which evaluate items' diversity and coverage in an item-based recommendation algorithm by proposing a k-means cluster based on the genre to rank the items. Considering the intra-list similarity evaluation, the effect of diversity level in the proposed algorithm is discussed, and the results are analyzed and compared with other similar algorithms. It is discovered that with a higher diversity level, the similarity of items reduces, and more diverse items are recommended. Furthermore, the coverage improved significantly, which proved that the proposed algorithm has better performance to recommend the new items. In future works, further research and investigations can be done in this area of study. In particular, studies on users can be explored to add more validations and insights to the proposed algorithm. Additionally, the proposed algorithm can be extended further based on other domains or fields.

REFERENCES

[1] F. Li, B. Wu, L. Xu, C. Shi, and J. Shi, A Fast-Distributed Stochastic Gradient Descent Algorithm for Matrix Factorization, JMLR: Workshop and Conference Proceedings 36(7787) (2014).

[2] Abdullah, Noraswaliza, Integrating collaborative filtering and matching-based search for product recommendation, Ph.D. thesis, Queensland University of Technology, Brisbane, Australia, (2012).

[3] MaizatulAkmar Ismail, Zahra Moghaddasi, Samira Sarvari, Ali Ebrahimi, Masoud Ghorbanian, Recommender Systems for Research Materials: An Overview, The 4th International Conference on Computer Science and Computational Mathematics, ICCSCM 2015, (2015) 349-354.

[4] SaikatBagchi, Performance and Quality Assessment of Similarity Measures in Collaborative Filtering Using Mahout, 2nd International Symposium on Big Data and Cloud Computing (ISBCC'15), 50 (2015) 229-234.

[5] Rena Nainggolan, ResiantaPerangin-angin, Emma Simarmata, and AstutiFerianiTarigan, Improved the Performance of the K-Means Cluster Using the Sum of Squared Error (SSE) optimized by using the Elbow Method8, 1st International Conference of SNIKOM, (2018) 23–24.

[6] F. Maxwell Harper and Joseph A. Konstan, The MovieLens Datasets: History and Context, ACM Trans. Interact. Intell. Syst. 5, 4, Article 19 (January 2016), (2016) 19.

[7] Marius Kaminskas and Derek Bridge, diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems, ACM Transactions on Interactive Intelligent Systems, 7(1) 2016.

[8] Rodrigues, C. M., Rathi, S., & Patil, G., An efficient system using item & user-based CF techniques to improve recommendation, 2016 2nd International Conference on Next Generation Computing Technologies, (2016).

[9] Eskandanian, F., Mobasher, B., & Burke, R., A clustering approach for personalizing diversity in collaborative recommender systems, Paper presented at the 25th Conference on User Modeling, Adaptation, and Personalization, Bratislava, Slovakia, (2017).

[10] Mehrjoo, S., Mehrjoo, M., &Hajipour, F., Optimal Diversity of Recommendation List for Recommender Systems based on the Users' Desire Diversity, Journal of Information Science Theory and Practice, 7(3) (2019) 31–39.

[11] Ruipeng Li and Luiz Fernando Capretz, Assessing the Performance of Recommender Systems with MovieTweetings and MovieLens Datasets, International Journal of Innovation, Management, and Technology , 10(6) (2019) 229-234.

[12] Adzemovic H, Sandor A., Comparison of user and item-based collaborative filtering on sparse data, Stockholm, Sverige: Royal Institute of Technology, (2017).

[13] Castells, P., Hurley, N. J., & Vargas, S., Novelty and diversity in recommender systems, In F. Ricci, L. Rokach, & B. Shapira (Eds.), Recommender systems handbook,. New York: Springer, 881-918 (2015).

[14] S. Koras, M. Venkatarao Rao and V. Bhuvaneswari, Bi-directional Methodology for Literature Extraction from PubMed Abstracts using Web Scrapper and Web Crawler, International Conference on Communication and Electronics Systems (ICCES), (2019) 1978-1983.

[15] Eric K. Clemons Paul F. Nunes, Carrying your long-tail: Delighting your consumers and managing your operations, Elsevier Science Publishers B. V. Amsterdam, The Netherlands, (2011).

[16] Tommaso Di Noia, Vito Claudio Ostuni, Jessica Rosati, Paolo Tomeo, and Eugenio Di Sciascio, An analysis of users' propensity toward diversity in recommendations, In Proceedings of the 8th ACM Conference on Recommender Systems (RecSys '14). Association for Computing Machinery, New York, NY, USA, (2014) 285–288.

[17] Zuo, Y., Gong, M., Zeng, J., Ma, L., & Jiao, L., Personalized recommendation based on evolutionary multi-objective optimization, IEEE Computational Intelligence Magazine, 10 (2015) 52–62.

[18] Gogna, A., & Majumdar, A., Diablo: Optimization-based design for improving diversity in recommender system, Information Sciences, 378 (2017) 59–74.

[19] Zhou, T., Kuscsik, Z., Liu, J.-G., Medo, M., Wakeling, J. R., & Zhang, Y.-C., Solving the apparent diversity-accuracy dilemma of recommender systems, Proceedings of the National Academy of Sciences, 107 (2010) 4511– 4515.

[20] Kunaver, M., &Pozrl, T., Diversity in recommender systems–a survey, Knowledge-Based Systems, 123 (2017) 154–162.

[21] Shanfeng Wang, L. M. Q. C. L. J., Maoguo Gong, Decomposition based multiobjective evolutionary algorithm for collaborative filtering recommender systems, In Proceedings of the IEEE Congress on Evolutionary Computation (CEC), (2014) 672–679.

[22] Owen, S., Anil, R., Dunning, T., & Friedman, E., Mahout in Action, Greenwich, CT, USA: Manning Publications Co, (2011).

[23] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl, Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems , 22(1) (2004) 5–53.

[24] Sean M. McNee, John Riedl, and Joseph A. Konstan, Being accurate are not enough: how accuracy metrics have hurt recommender systems, CHI '06 Extended Abstracts on Human Factors in Computing Systems. Association for Computing Machinery, New York, NY, USA, (2006) 1097–1101.

[25] Asst.Prof.Johnsymol Joy, Overview of Different Data Clustering Algorithms for Static and Dynamic Data Sets, International Journal of Computer Science and Engineering (SSRG-IJCSE), (2018).