Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

JURNAL TEKNIK INFORMATIKA
Homepage : http://journal.uinjkt.ac.id/index.php/ti

Diversity Balancing in Two-Stage Collaborative Filtering for Book Recommendation Systems

Rifqi Fauzia Muttaqien[1], Dade Nurjanah[2], Hani Nurrahmi[3]

[1,2,3]Informatics, School of Informatics, Telkom University
[1,2,3] Jl. Telekomunikasi No. 1, Bandung, West Java

ABSTRACT

The large amount of information on the internet makes the role of recommendation systems very important in helping users choose the desired product in various fields, one of which is online book sales. The recommendation system helps users find the books that are most relevant to them. One approach that is often used in recommender systems is Collaborative Filtering (CF). However, CF has several shortcomings, one of which is that the system only recommends items that are popular and most relevant to users. This causes the item recommendations given to users to be less diverse. Therefore, we propose a book recommendation system based on Two-stages CF using the Diversity Balancing method to balance diversity in the recommendation results. System accuracy is measured using precision and recall, while diversity is measured using personal diversity and aggregate diversity. The test results indicate that as the number of recommended items increases, the proposed system's accuracy improves, but the diversity of recommended items decreases. In consideration of the trade-off between accuracy and diversity, our system achieves a recall score of 0.301, a precision score of 0.282, a PD score of 0.048, and an AD score of 0.095 with recommendation list size of 8 items.

Keywords: *recommender system, collaborative filtering, diversity balancing*

Article:
Accepted: September 15, 2023
Revised: June 12, 2023
Issued: October 28, 2023

***Correspondence Address:**
rifqifauzia@student.telkomuniversity.ac.id

1. INTRODUCTION

In this modern era, there is so much information available on the internet that it makes it difficult for internet users to choose the available products or services. To overcome this problem, researchers developed a recommender system to provide relevant recommendations to make it easier for users to choose the product or service [1]. Recommender systems have been widely used in various fields. One of the biggest contributions of recommender systems is in book recommendations [2]. The book recommender system functions to provide a number of book recommendations based on user preferences [3]. With the help of these recommendations, users are able to access many books with less effort [4].

Collaborative Filtering (CF) is one of the methods in recommender systems that is most commonly used today [5][6]. CF works by predicting item ratings for certain users based on items that have been previously rated by other similar users [7]. Several researchers have developed recommender systems using CF in various fields. In book recommendations, researchers [3] developed a CF-based book recommender system. Meanwhile, researchers [4] developed an improved CF-based system. Researchers [8] developed a user-based system using KNN. In e-commerce field, researcher [9] designed a product recommender system using combination of PCA K-Means with user-based CF.

However, a CF-based recommender system has several weaknesses, one of which is that the system is too focused on providing relevant recommendations for its users. Generally, recommender systems ignore less relevant popular items or new items with few ratings and only focus on items that have enough ratings to be recommended [10][11]. This causes the item recommendations given by system to be less diverse and makes the scope of recommended items very narrow. So there needs to be a recommender system that can provide diverse recommendations with high accuracy [7].

Several studies have been conducted in addressing the lack of item diversity in recommender system. Research [7] used the diversity balancing method and succeeded balancing recommendation item diversity. In Research [12], a movie recommender system was developed using Two-stage Collaborative Filtering, successfully increasing item diversity while maintaining accuracy, precision, recall, personal diversity, and aggregate diversity. In research [10], researchers successfully increased the Personal Diversity of the recommendation results in the recommender system by combining Content-based and Collaborative Filtering methods.

In this research, the authors are interested in implementing diversity balancing in a book recommendation system. Therefore, we propose a book recommendation system based on Two-stage Collaborative Filtering with the Diversity Balancing method. This approach was chosen because it can increase diversity while maintaining good accuracy of CF results. Two-stages Collaborative Filtering is a CF method that has two main stages. The first stage is the generating a list of candidate items recommended to users. In this research, we use K-Means CF because it works well in sparse data. The second stage focuses on diversity balancing of the recommendation results. Hence, the recommendations provided by the system can contain less popular items but are still relevant to user preferences.

There are 3 datasets used in this research, i.e. ratings, users, and books datasets obtained from the Book-Crossing Community. System accuracy is measured based on precision and recall. While diversity of system is measured based on Personal Diversity and Aggregate Diversity.

2. METHODS

2.1. Preprocessing Data

In this research, we use the Book Recommendation Dataset created by Cai-Nicolas Ziegler. This dataset consists of three datasets, that is the ratings dataset which contains book ratings from each user, the user dataset which contains user demographic data, and the books dataset which contains detailed data related to books.

Due to limited processing time and computing capabilities, this research only uses partial data under certain conditions. The conditions applied are that the rating values used only come from books that have been rated by at least 200 users, while the user data that will be used is users who have rated at least 7 books. The ratings dataset contains 33,251 rows of data with 3 feature columns. The rating value range for an item is 1.0 – 10.0. The user dataset contains 1792 rows of data with 3 feature

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

columns. Meanwhile, the books dataset consists of 195 data with 5 feature columns. Tables 1, 2, and 3 show examples of a user dataset, a rating dataset, and a book dataset, respectively.

Table 1.User dataset

User-ID	ISBN	Book-Rating
276725	034545104X	0
276726	0155061224	5
276729	052165615X	3
276727	0446520802	0
276729	0521795028	6

Table 2. Rating dataset

User-ID	Location	Age
1	nyc, new york, usa	Nan
2	stockton, california, usa	18
3	porto, v.n gaia, Portugal	17
4	moscow, yukon territory, russia	Nan
5	farnborough, hants, united kingdom	Nan

Next, the data will be cleaned from empty and invalid values. The user and rating datasets are then combined and aggregated. Thus, a user rating dataset is formed which is ready to be used in the process of the proposed system.

2.2. The Proposed Method

After collecting and preprocessing data is completed, in this research we propose a book recommendation system based on Two-stage Collaborative Filtering with the Diversity Balancing method.

a. User clustering

The K-Means method is used to form cluster of users so that a list of recommended items can be generated from a collection of users with similar preferences. K-Means will be trained using data features from the user dataset, that is the user's age and location. K-Means works by calculating the similarity between data objects by calculating the distance between the data. The distance between data is measured using Euclidean Distance with Equation 1 [9]:

$$d = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2} \quad (1)$$

Where x and y are the two data whose distance will be measured, n is the number of features in the data, x_k and y_k are the k_{th} features in the data x and y. Then K-Means will group the data based on its similarity or closeness to other data. The K-Means algorithm is explained in detail in Figure 1.

The optimal number of clusters that will be formed by K-Means will be measured using the Silhouette Coefficient in Equation 2 [9].

$$g_i = \frac{m_i - n_i}{max\ (m_i, n_i)} \quad (2)$$

Where m_i is the average minimum distance between clusters, n_i is the average distance from a point in a cluster to all data in the same cluster.

Table 3. Book dataset

ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
0195153448	Classical Mythology	Mark P. O. Morford	2002	Oxford University Press
0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada
0393045218	Decision in Normandy	Carlo D'Este	1991	HarperPerennial
0393045218	Flu: The Story of the Great …	Gina Bari Kolata	1999	Farrar Straus Giroux
0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton & Company

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

Algorithm 1 : User Clustering with K-Means

Input : User data
Output: Prediction user cluster

Begin.
1. Determine number of clusters to be formed
2. Determine the cluster center (centroid). For initial conditions, one data will be randomly selected as the centroid
3. Calculate the distance of each data to the centroid using equation (1)
4. Grouping data based on the proximity of the data to the centroid. The smaller the value, the closer the data is to the cluster centroid
5. With the same number of clusters, find the new centroid of each cluster by calculating the average distance of each data in the cluster
6. Repeat steps 2 to 5, the loop stops when the position of all data or clusters has not changed any more

End.

Figure 1.K-Means clustering algorithm

b. Generate recommendation

User-ID Location Age

At this stage, a list of candidate items will be created using a User-based Collaborative Filtering (User-based CF) recommendation system. To overcome the sparsity problem in the User-based CF algorithm, empty values in the user-item matrix will be filled with the average user value in each cluster so that the recommendation results can be better [9]. The process of predicting ratings and creating a list of recommendation items for users using CF begins with calculating user similarity. The user similarity value between users u and v is obtained from the Cosine similarity Equation 3.

$$sim(u,v) = cos(\theta) = \frac{u.v}{\|u\|.\|v\|} \quad (3)$$

Where u and v are the dot products between two users. The similarity value range is between -1 to 1, if the value is closer to 1, the more similar the two users are. If it is close to 0 then the two users are not similar, and if it is close to -1 then the two users are opposites.

After calculating user similarity, the top five users who have the greatest similarity are selected as bases who participate in predicting item ratings. Item ratings are predicted by the Equation 4 [7].

$$P_{u,i} = \overline{R_u} + \frac{\sum_{v \in Uu} sim(u,v) \times (R_{v,i} - \overline{R_v})}{\sum_{v \in Uu} sim(u,v)} \quad (4)$$

Where $P_{u,i}$ is the predicted rating from user u on item i and $sim(u,v)$ is the similarity value between users u and v. Based on active user rating predictions, a number of items with high ratings will be included in the list of candidates for recommendation. The User-based CF algorithm is explained in detail in Figure 2.

Algorithm 2 : User-based Collaborative Filtering

Input : User data U, rating data R, num of neighbor N
Output: List of recommendation Items

Begin.
1. Define the neighbors of the active user. Neighbors are users who are close / similar to active users. The number of neighbors determines the number of users that must be considered in calculating the rating prediction
2. Calculate the similarity of active users with their neighbors using the Cosine equation in equation (3)
3. Predict ranking values for products that have never been rated by active users. The predicted value is obtained by equation (4)
4. Sort the ranking predictions from highest to lowest. Then take a number of items as a list of candidate recommendation items

End.

Figure 2. User-based collaborative filtering

c. Diversity balancing

Conventional recommendations generally only use user ratings to rank items and the aim is only to increase accuracy [13]. Therefore, it is necessary to determine the optimal trade-off between accuracy and diversity of the system [7]. In this study, diversity is added to the considerations in recommending items. Diversity balancing is carried out by replacing the item that has the highest rating with an item outside the list of candidates that has the lowest similarity to that item.

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

In the item replacement process, the system compares a number of top items from each candidate list with a number of items with the highest rating outside the candidate list (neglected list). The item that has the lowest similarity is selected as a replacement item for the popular item. Comparison of item similarity is carried out with the Equation 5 [5][13]:

$$\frac{R_i - R_j}{P_i - P_j} \quad (5)$$

Where R_i and R_j are the rating values given by user u to items i and j, P_i and P_j are the popularity of items i and j. The popularity of an item is obtained by calculating the number of ratings from users for the item. After the item is replaced, you will get a top-n list of book recommendations for the user. Diversity balancing in CF algorithm is explained in Figure 3.

Algorithm 3 : Diversity Balancing in Two-stage Collaborative Filtering (DBTS)
Input : Candidate list, number of replacement **Output**: Top-n recommendation list
Begin. 1. Select the item with the highest predicted rating as many as the number of replacement items 2. Calculating replacement items with equation (5) 3. Enter the item with the lowest rating into the recommendation list 4. Make a top-n recommendation list **End.**

Figure 3.Diversity balancing algorithm

2.3. Evaluate System Performance

There are two things that are used to evaluate the performance of this recommender system, that is the accuracy of recommendations and the diversity of results from the recommender system. System accuracy is measured based on precision and recall. Precision is the comparison between true positives from all positives. In recommender systems, precision will usually decrease if the number of recommendation items increases. Precision is calculated using Equation 6.

$$precision = \frac{t_p}{t_p + f_p} \quad (6)$$

Meanwhile, recall is a comparison between true positives from all those that should be positive. In other words, recall is how much the algorithm can detect true positives. In recommender systems, recall is the opposite of precision where the recall value will increase if there are more recommended items. Recall is calculated using Equation 7.

$$recall = \frac{t_p}{t_p + f_n} \quad (7)$$

Diversity in the system is measured based on Personal Diversity (PD) and Aggregate Diversity (AD). PD defines dissimilarities between items in a recommendation list. PD is calculated with Equation 8 [5][13].

$$PD = \frac{1}{|U|}\sum_{u \in U}\left(1 - \frac{\sum_{i,j \in L(u), i \neq j}(1 - sim(i,j))}{\frac{1}{2}|L(u)|(|L(u)| - 1)}\right) \quad (8)$$

Where $|U|$ is the number of users, $sim(i,j)$ is the similarity of users i and j. $L(u)$ is a list of recommendations and $|L(u)|$ is the number of items in the recommendation list $L(u)$. Next, Aggregate Diversity is the total unique items recommended to all users. AD is calculated with Equation 9 [5][13].

$$AD = \frac{\sum_{u \in U} |L(u)|}{|I|} \quad (9)$$

Where $|I|$ is the number of all items recommended to users.

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

3. RESULTS AND DISCUSSION

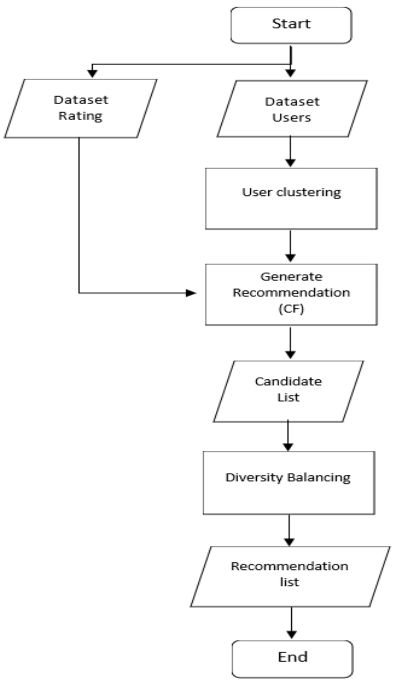

Figure 4.Proposed system workflow

In this research, we implemented a CF-based book recommendation system. Unlike existing research which only focuses on the accuracy of recommendations, we try to balance the diversity of recommendation results given to users. Hence, we propose a book recommendation system based on Two-stage CF with diversity balancing.

The proposed system has 2 main stages, that is creating a recommendation list and balancing the diversity of items within that list. In the first stage, we begin by grouping users based on their similarity to each other by forming user clusters using the K-Means algorithm. The user clusters that have been formed are then used in CF to create a recommendation list based on items liked by other users within the same cluster.

Moving on to the second stage, we apply diversity balancing. This involves replacing some of the highest-rated items with items outside the list that have the lowest similarity. Hence, the final result of the system is a list of recommendations whose diversity of items has been balanced. Figure 4 shows the workflow of our system.

3.1. Testing

All datasets used will undergo a data preprocessing stage where the data will be cleaned and prepared so that the data can be used in system training and testing. The dataset will then be divided into training data consisting of 1433 data, and test data consisting of 359 data. The training data will be used to train the K-Means model on the CF algorithm, while the test data will be used to evaluate the recommender system.

a. Clustering Test Result

The optimum number of clusters in K-Means is determined by measuring clustering performance on test data with different numbers of clusters. In this test, the number of clusters has a value range between 2 and 19. The results of the K-Means performance test are shown in Figure 5.

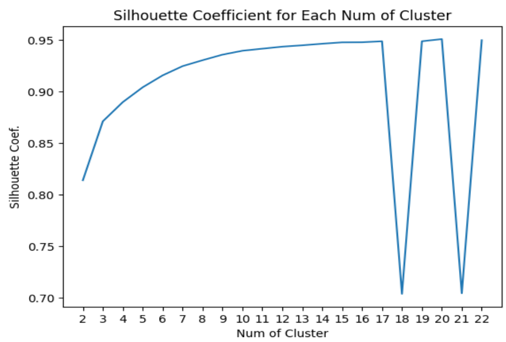

Figure 5.Silhouette coefficient score

Based on Figure 5 it can be seen that the highest Silhouette coefficient is achieved when the number of clusters is 17 with a value of 0.95. Silhouette coefficient value continues to increase until the number of clusters is 17. However, the Silhouette coefficient drops significantly and become unstable when the number of clusters is more than 17. Therefore, for the next testing stage the optimum number of user clusters can be determined as 17 clusters. The results of K-Means on user data are shown in Table 4.

Table 4.K-means result on user dataset

User-ID	Cluster
0399501487	5
0061009059	13
0345361792	1
0425180638	14
0385508042	5

b. Two-stage CF Test Result

Then the next test is carried out by running the recommender system model against the user test in test data. Testing was carried out to obtain a list of recommended items for all user tests that would be used in the next stage. The accuracy and diversity of the recommender

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

system are evaluated using evaluation metrics. Several system parameters will be tested to see how the parameters affect the accuracy and diversity of the system.

The first parameter tested was the influence of the number of recommendation items (k) on system precision and recall. Then test the effect of the threshold rating on precision and recall to find out the lower limit of the optimum rating to be included in the calculation. As well as testing the effect of the number k on the diversity of the system using Personal Diversity (PD) and Aggregate Diversity (AD).

The first test was carried out by calculating the effect of the number k, where k is the number of items in the recommendation list that will be included in the calculation. Suppose value of k has a value range of 6, 8, 10, 12, and 14, the threshold rating used is 5, then the results obtained are as shown in Table 5.

Table 5.System accuracy for each k

k (*recommendation list size*)	*Metrics*	
	precision	*recall*
6	0.2211	0.2552
8	0.2349	0.2965
10	0.2296	0.3209
12	0.2268	0.3468
14	0.2231	0.3679

Based on Table 5, it can be observed that the number of recommendation items (k) influences the precision and recall values of the system. The recall value tends to increase when the k value becomes larger. Meanwhile, the precision value tends to increase when the k value is less than 10, then tends to decrease if the k value is more than 10. This shows that when the number of recommended items increases, the system is able to provide many relevant item recommendations to the user, but the system is less able to provide items. according to user preferences.

Then the effect of the threshold rating on the accuracy value is tested to find the optimal threshold rating. This test is carried out using the same number of k for each threshold rating. Suppose the k value is 10 and the threshold rating values tested are 1, 3, 5, and 7, then the model performance is obtained based on the threshold ratings in Table 6.

Table 6.System accuracy by threshold rating

Threshold Rating	Metrics	
	Precision	Recall
1	0.2094	0.3158
3	0.2100	0.3156
5	0.2296	0.3209
7	0.2877	0.3259

The results in Table 6 show that the threshold rating affect the precision and recall of the system. The higher the threshold rating, the system precision increases. Recall also tends to increase when threshold rating getting higher, although it is not significant. This is because the items included in the recommendation calculation are high rating items, that is the items most liked by users. Therefore, the system only processes items with high relevance to the user.

Then testing the diversity metrics of the recommender system is carried out. Testing is carried out to measure PD using Equation 8 and measure AD using Equation 9. The test was carried out by looking at the effect of the number of k on the diversity metric. Suppose the k value has a value range of 6, 8, 10, 12, and 14, and the threshold rating value is 5. Table 7 show the test results.

Table 7.System diversity by k

k (recommendation list size)	Metrics	
	PD	AD
6	0.0508	0.1251
8	0.0489	0.0959
10	0.0489	0.0776
12	0.0481	0.0661
14	0.0482	0.0570

Test result in Table 7 explains that the number of items influences PD and AD differently. Based on these results, it can be seen that the PD value tends to stable when the number of items increases. This stable PD value indicates there is no much change in the dissimilarity between items in the recommendation list. Meanwhile, the AD value tends to decrease when the number of k increases. This shows that the number of unique items for each user decreases when the number of recommendation items increases.

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

We compare the output of the recommender system before and after diversity balancing to see how diversity balancing affects the results of recommendations given to a user. For example, for an active user, the number of items on the recommendation list (k) are 10, and the number of items to be replaced is 3, then the recommendation results are shown in Table 8. Table 8 show that the system balances diversity by replacing the 3 items with the highest predicted rating with other items outside the recommendation list. Replacement items are selected based on the dissimilarity value between the recommendation item and the replacement item which is measured using Equation 5. The top 3 items on the recommendation list are the items that are least similar to the previous item, as well as items that are least relevant to active users.

Table 8.Recommendation result before and after diversity balancing

Collaborative Filtering		Collaborative Filtering with Diversity Balancing	
ISBN	Predicted-Rating	ISBN	Predicted-Rating
0312924585	9.35	0399501487	3.92
0142000205	8.56	0061009059	6.50
006101351X	8.56	0345361792	6.32
0425180638	8.35	0425180638	8.35
0385508042	8.16	0385508042	8.16
0440211263	8.16	0440211263	8.16
0345313860	8.15	0345313860	8.15
0345337662	7.84	0345337662	7.84
0345351525	7.84	0345351525	7.84
0060930535	7.54	0060930535	7.54

For comparison, we compare the performance of the proposed system with a recommender system with diversity balancing using Hierarchical Clustering (HC). The benchmark system (HC) is tested for performance with the same test data and test methods. Figure 6 shows the performance comparison between the proposed system and the HC system.

According to Figure 6 it can be explained that in general the proposed system has better precision and recall compared to the HC system. Although there are notable differences in the recall of the two systems, the recall trends for both systems are similar, increasing up to k=16 and then tending to stabilize. Precision results also reveal a similar trend which decreases when the thresholds are 1,3, and 5.

This occurs because the low threshold rating value used in diversity balancing causes many items with medium to high ratings to be replaced. Meanwhile, at threshold 7, the precision of both systems tends to be stable. In terms of diversity, the two systems do not have significant differences. When considering both accuracy and diversity of the system, the optimal condition is observed at threshold 7 and k = 8. In this parameter, the system achieves the highest PD and AD values, which are 0.048 and 0.095, respectively. Additionally, the recall is 0.301, and precision is 0.282.

Based on the results of this research, it can be concluded that Two-stage Collaborative Filtering can be applied to book recommendation systems. However, the small personal diversity and aggregate diversity scores indicate that the diversity balancing in the proposed system has not been effective. This differs from other studies that have reported higher diversity balancing scores, indicating more diverse recommendation results.

Jurnal Teknik Informatika Vol. 16 No.2, October 2023 (194-203)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v16i2.36580

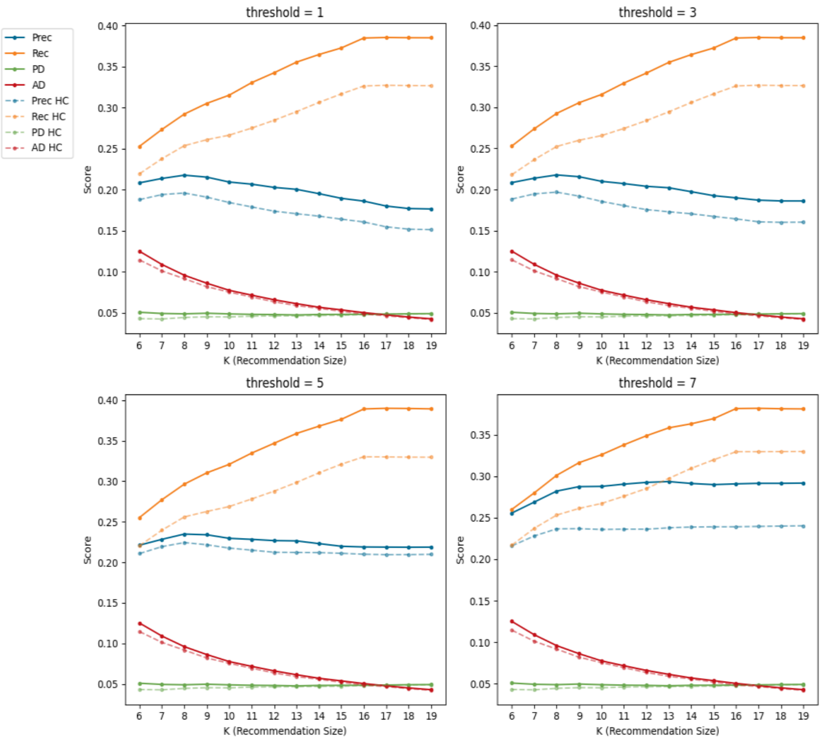

Figure 6.Comparison of the performance of the proposed system with HC systems

CONCLUSION

In this research, we have implemented a recommendation system based on Two-stage Collaborative Filtering using the Diversity Balancing method in the book domain. The accuracy of the proposed system is measured using precision and recall. Meanwhile, the diversity of the system is measured using Personal Diversity and Aggregate Diversity. Based on test results, our system has not been able to balance the diversity of recommendation items well. Test results show that system accuracy increases when more items are recommended. As a result, the diversity of recommended items continues to decline. In optimal conditions with threshold rating 7 and the optimal number of recommendation items to users is 8 items our system achieves a recall score of 0.301, a precision score of 0.282, a PD score of 0.048, and an AD score of 0.095.

Therefore, this research can be developed better in the future by overcoming the limitations experienced by researchers and implemented in new domains. Apart from that, the development of Diversity Balancing in Two-stage Collaborative Filtering can be done with other algorithms, such as Hierarchical Clustering and DBSCAN.

REFERENCES

[1] Himani Jain, B. B. (2023). BOOK RECOMMENDATION USING MACHINE LEARNING. International Research Journal of Modernization in En-gineering Technology and Science.

[2] Khalid Anwar, J. S. (2020). Machine learning-based book recommender system: a survey and new perspective. Int. J. Intelligent Information and Database Systems.

[3] Ramakrishnan, G., Saicharan, V., Chandrasekaran, K., Rathnamma, M.V., Ramana, V.V. (2020). Collaborative Filtering for Book Recommenda-tion System. In: Das, K., Bansal, J., Deep, K., Nagar, A., Pathipooranam, P., Naidu, R. (eds) Soft Computing for Problem Solving. Advances in Intel-ligent Systems and Computing, vol 1057. Springer, Singapore. https://doi.org/10.1007/978-981-15-0184-5_29

[4] E. Uko Okon, B. O. (2018). An Improved Online Book Recommender System using Collaborative Filtering Algorithm. International Journal of Computer Applications.

[5] Sebastia, L. et al. (2009) ‘E-Tourism: A tourist recommendation and planning

application', International Journal of Artificial Intelligence Tools, 18, pp. 717–738.

[6] Asanov, D., (2011). Algorithms and methods in recommender systems. Berlin Institute of Technology, Berlin, Germany

[7] Zhang, Liang et al. (2020) 'Diversity Balancing for Two-stage Collaborative Filtering in Recommender Systems', Applied Sciences, 10(4).

[8] Rohit, S. S. (2017). Proposed Approach for Book Recommendation Based on User k-NN. Advances in Computer and Computational Sciences.

[9] Andra, D., & baizal, A. B. (2022). E-commerce Recommender System Using PCA and K-Means Clustering. Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi), 6(1), 57 - 63. https://doi.org/10.29207/resti.v6i1.3782

[10] Hamedani, E. M. and Kaedi, M. (2019) 'Recommending the long tail items through personalized diversification', Knowledge-Based Systems, 164, pp. 348–357.

[11] Park, yoon-joo (2013) 'The Adaptive Clustering Method for the Long Tail Problem of Recommender Systems', Knowledge and Data Engineering, IEEE Transactions on, 25, pp. 1904–1915. doi: 10.1109/TKDE.2012.119.

[12] F. W. Ardiyanto et al., "Diversity Balancing pada Two-stage Collaborative Filtering untuk Mengatur Less Popular Item dalam Domain Film," in eProceedings of Engineering, 2023, vol. 10, no. 3, pp. 3455–3463.

[13] Gogna, A. and Majumdar, A. (2017) 'Balancing Accuracy and Diversity in Recommendations Using Matrix Completion Framework', Know.-Based Syst., 125(C), pp. 83–95

[14] Adomavicius, G. and Tuzhilin, A. (2005) 'Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions', IEEE Transactions on Knowledge and Data Engineering, 17(6), pp. 734–749

.