Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

Fashion Recommendation System Using Collaborative Filtering

Muhammad Khiyarus Syiam[1], Agung Toto Wibowo[2], Erwin Budi Setiawan[3]

School of Computing, Informatics, Telkom University, Bandung, Indonesia
Email: [1]rlzmtnck@student.telkomuniversity.ac.id, [2]agungtoto@telkomuniversity.ac.id, [3]erwinbudisetiawan@telkomuniversity.ac.id
Correspondence Author Email: agungtoto@telkomuniversity.ac.id
Submitted: **20/06/2023**; Accepted: **25/09/2023**; Published**: 27/09/2023**

Abstract−Collaborative Filtering is an method used to build a recommendation system with the concept that conclusions from different clients are used to anticipate things that may be of interest to users. This research uses data from Rent the Runway and the method used is Item-based Collaborative filtering, where the system will look for similarities in products that have been purchased by customers and then look for predictive values. Fashion requires recommendations because it plays a crucial role in helping individuals express their identity, personal style, and personality through clothing choices, accessories, and dressing styles.The recommendation system uses the item method based on analyzing the number of purchases or sales and grouping according to each product category so that it can help consumers in choosing fashion products. It was found that the use of Adjusted Cosine Similarity produces better recommendations with an average MAE value of 0.2750, while Cosine Similarity with an average MAE difference of 0.3989. This proves that the use of adjusted cosine similarity can produce better recommendations because the adjustment algorithm not only considers user behavior, but also produces lower performance errors**.**

Keywords: Collaborative Filtering; Cosine Similarity; Recommendation System; Item-based, Fashion;

1. INTRODUCTION

The existence of internet can provide many changes in various fields, one of which is in the world of fashion. Shopping for clothes online will make it easier for consumers to save time and energy. shopping online still requires consideration in the form of suggestions for reviews and ratings from several consumers who have bought the product to determine the quality of the product and its suitability for a person[1], [2]. The wide array of fashion choices often poses a challenge in discovering the perfect ensemble. An ingenious solution lies in recommender systems, which employ Collaborative Filtering to forecast ideal item selections, such as book recommendations. Expanding its application, a collaborative filtering-based recommendation system employing the Item-based approach can also be leveraged to suggest a vibrant community for the youth in Semarang city. Notably, recommendation systems encompass various methodologies, including Content-based Filtering, Hybrid Filtering, and Cooperative Filtering[3], [4].

Based on the problem above, a system is needed that can help in decision making. Collaborative sifting is one of the strategies used to build a recommendation system with the concept that assessments from different clients are used to anticipate things that may be of interest to a user. The CF recommendation system method used by the author in the construction of this system is Item-based CF[5], [6]. This approach relies on the resemblance between items, utilizing user-provided ratings. The quest for item similarity is accomplished through the utilization of adjusted cosine similarity. This technique finds its application in recommendation systems, assessing the likeness between two items based on user preferences. While akin to Cosine Similarity, it incorporates modifications tailored specifically for recommendation systems, enhancing its suitability[7], [8]. The collaborative filtering (CF) approach offers distinct advantages, allowing active users to receive suggestions based on items that were purchased and positively rated by users with similar interests[9], [10]. By leveraging previous ratings from engaged users, a model is constructed to propose a fresh array of comparable products, leveraging transaction history.

Previous research utilizing the method of Item-Based Collaborative Filtering was conducted by Ritdrix in 2018. This research contains the development of a recommendation system at the Library. This research uses the method of collaborative item-based filtering, where the system will look for similarities in the model of borrowing books with others[11]. The system test results reach a value of 95.68% so that it can prove that the system has fulfilled the functionality and in accordance with the needs[12].

Research by Gunawan Using the Thing Based Cooperative Sifting Strategy Based on the Adapted Cosine Method to recommend products in 2022. In this recommendation system made utilizing the method of collaborative filtering items based on the investigation into the number of purchases or the number of sales and grouped according to each product category so that it can help consumers in choosing clothes [10]. This system aims to predict information that is of interest to consumers based on the similarity of items that have been purchased by these consumers [10].

Research conducted by Rizki who recommended Youth Communities in Semarang City utilizing the method of collaboration with the for item-based filtering algorithm for adjusting cosine in 2018. This journal contributes to the creation of a system of recommendations for youth communities in the city of Semarang using the Thing Based Cooperative Separating approach and the Cosine Adjusted method. This method can assist youth in finding communities that match their interests and preferences[15], [16].

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

Based on the research above, in this study, a The collaborative filtering technique is used, especially with item-based on the recommendation system to be developed. This system will display product recommendations, especially fashion.

2. RESEARCH METHODOLOGY

2.1 Research Stages

The system is built on the Google Collaboratory platform utilizing the programming language Python with the following system flow. The first step is to prepare a dataset containing user, item, and rating information. After the dataset is ready, the dataset checking process is carried out to ensure consistent and clean data, the dataset is imported into the platform using Python[17], [18]. After that, the dataset is broken up into subsets of data for testing and training to calculate the average rating for each item in the dataset. Average rating is calculated and then similarity between items and ratings on similar items are calculated. The final recommendation result is given based on the item with the highest predicted rating.

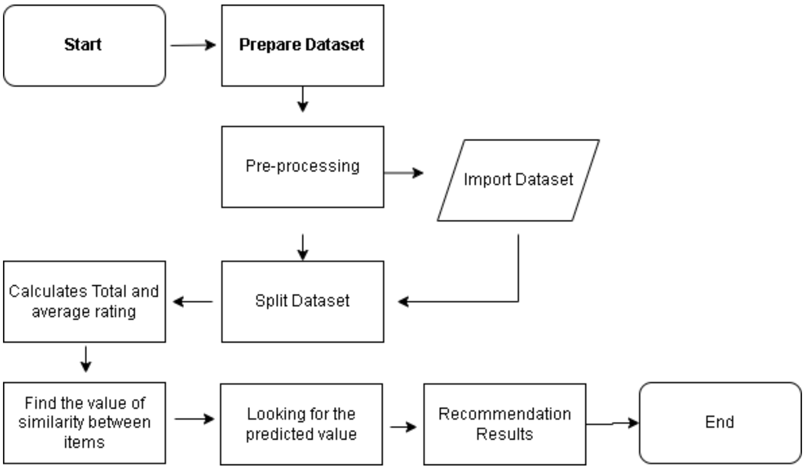

Figure 1. Collaborative Filtering Recommendation System Design

Figure 1 is a technical description of the procedure design. The following is an overview of the System Flow that will be built in plain view. Can be seen in Figure 1 there are several processes. Here are the process steps :

a. Prepare Dataset
The first thing to do is to prepare a dataset that has not been processed. In this study, the dataset was taken from an online shop called Rent the Runway.

b. Pre-processing
Pre-processing entails undertaking data cleansing, transformation, and organization tasks to ensure optimal data quality and readiness for subsequent analysis. In the cleansing phase, any rows containing empty values are removed. Next, in the transformation step, the rating values are adjusted from a 1-10 scale to a 1-5 scale. The original dataset featured ratings ranging from 1 to 10, with an increment of 2. To simplify calculations, these ratings are transformed to a 1-5 range, mirroring the notation of 1, 2, 3, 4, and 5. Lastly, during the organization phase, key metrics such as the number of users, items, and unique ratings in the dataset are counted. Additionally, the Rating, User_id, and Item_id columns are separated, as these columns exclusively contribute to the subsequent processes at hand.

c. Split Dataset
Furthermore, in the data separation process, we separate the test data and training data for the evaluation process

d. Find the value of similarity between items
In this process, we will look for similarity values between items using the Adjusted Cosine Similarity formula. The output of this process produces a pivot table containing the adjusted cosine similarity values

e. Looking for predicted value
In the pursuit of predicting the value for an item where the user's preference remains unknown, the initial stride involves identifying the items that bear the closest resemblance to the aforementioned item, drawing upon the precomputed similarity measures. The prediction of user preference values for these uncharted items can be obtained through the summation of the average user preference alongside the previously computed average weight.

f. Recommendation Result
After the predicted value is obtained, the system can provide item recommendations by selecting the item with the highest predicted value.

2.2 Data Entry Process

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

The input data in the system is data in the form of rating values chosen by customers on the system in the form of values from 1-5 with good to less good information.

Table 1. Example of Rating Data Given by the users to the items

User	A	B	C	D
U1	3	5	3	-
U2	1	-	2	-
U3	4	2	4	2
U4	-	5	4	2

Table 1 is an example of data input on a system where each user gives a rating of the product. Suppose there are 4 products, namely (A, B, C, and D) and 4 users/customers (U1, U2, U3, U4). After purchasing the product, U1 gave a rating to the goods he bought with a product value of A = 3, B = 5 and C = 3, while product D was not given a rating value. U2 gives values to products, namely A = 1 and C = 2, for U3 products A = 4, B = 2, C = 4 and D = 2, for U4 products A = 5, C = 4 and D = 2. Next, the process of calculating the average rating value will be carried out[19], [20].

$$R_k = \frac{a + b + c}{R} \tag{1}$$

Where:
R_k = Rating average value.
a, b, c = Rating products a, b, c.
R = Number of rated products.
After the product value is calculated, the average rating value for K1 is 3.6667. Likewise the calculations for K2, K3 and K4 are also the same as K1. Giving this value is based on satisfaction between consumers, after everything is calculated it will be displayed in a Table 2.

Table 2. User Average Rating

User	A	B	C	D	Average Rating
U1	3	5	3	-	3,6667
U2	1	-	2	-	1,5
U3	4	2	4	2	3
U4	-	5	4	2	3,6667

2.3 Similarity Value

The method used in this study to find the similarity value between items is Adjusted Cosine Similarity [21]. This method is used to find the similarity of users who give different and very striking ratings. For example, if an item is given a rating scale from 1 to 5, some users may give a rating of 5 to some items that they like considering the item is not too bad, while some other users, may give a rating value of 5 only for items that they really like. To overcome this difference in scale problem, the adjusted cosine similarity method is used [22].

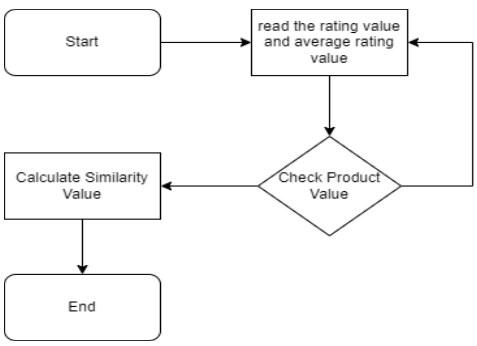

Figure 2. Find Similarity Value Using Adjusted Cosine Similarity

Figure 2 is the method that will be used in this study to find the similarity value between items is Cosine Similarity Adjusted. This method is used to find the similarity of users who give different and very striking ratings.

After getting the average rating, finding the is the next step. similarity value between items. In this case, the algorithm used is Adjusted Cosine Similarity. The following are the steps:

a. Check the rating values owned by the product, for example, select products a and b.
b. Check the first column and row, products a and b if values are found, namely 3 and 5. In the event that one of the products possesses no rating value (blank), then the resemblance cannot be calculated.

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

c. Check the subsequent section and column, until you get a value on products b and c and get a rating value of 4 and 2.
d. Checking columns and rows will continue through the final row, namely row 4.
e. In the subsequent step, the similarity value must be determined based on the rating not entirely set in stone to be suitable between the products.

Table 3 illustrates the rating values for products a and b in the first column and rows a and b have values 3 and 5. The next step is to move to the second row and column, where product a has a value of 1 and b is empty. So it cannot be used as a calculation because it does not meet the calculation requirements. The calculation requirement will be carried out if the two adjacent product values must have a rating value. With the same stage, rating values of 4 and 2 are obtained in the third row. The next example is the calculation of the value of similarity with the equation.

$$\operatorname{sim}(i_p, i_q) = \frac{\sum_{i=1}^{m} \left(R_{k,p} - R_k\right) \cdot \left(R_{k,q} - R_k\right)}{\sqrt{\sum_{i=1}^{m} \left(R_{k,p} - R_k\right)^2} \sqrt{\sum_{i=1}^{m} \left(R_{k,q} - R_k\right)^2}} \quad (2)$$

Where:

$\operatorname{Sim}(i_p, i_q)$ = is the adjusted cosine similarity formula from i_p to i_q
i_p = Active items.
i_q = Items to be compared.
$\sum_{k=1}$ = The arrange of users who provide ratings for items/products p and q.
$R_{k,p}$ = User rating k on item p.
$R_{k,q}$ = User rating k on item q.
R_k = The average value of the rating on user k.

Let's explore the computation of adjusted cosine similarity using Table 3 as a vivid example. The intricate workings of adjusted cosine similarity are vividly demonstrated in Equation (3).

$$S(i_p, i_q) = \frac{(3-3{,}6667)(5-3{,}6667)+(4-3)(2-3)}{\sqrt{(3-3{,}6667)+(4-3)^2}\sqrt{(5-3{,}6667)+(2-3)^2}} \quad (3)$$

Table 3. Comparison of User Ratings

User	$R_{k,p}$	$R_{k,q}$	R_k
U1	3	5	3,667
U3	4	2	2

Table 3 will display a Comparison of User Ratings and After performing calculations using the Adjusted Cosine Similarity equation to determine the values of their similarities products a and b, the similarity that follows are gotten for a value of -0.9430. After doing the calculations in the same way, the consequences of closeness between products are obtained as shown in Table 4.

Table 4. Overall Calculation of Adjusted Cosine Similarity using data from table 1

Products Compared	Products Compared	Adjusted Cosine Similarity Value
A	B	-0,9430
A	C	0,7049
A	D	-1
B	C	-0,5426
D	D	-0,3773
C	D	-0,7592

Process table 4 is a table that reflects the overall results of calculating the value of similarity between products. This table depicts a comparison of the calculated values between each product pair in the dataset. From the comparison of these values it can be obtained a similarity value that can be recommended, namely items A and C of 0.7049. other items are not recommended because the value of adjusted cosine similarity is negative. This similarity value indicates the extent to which the two products have similarities in certain characteristics or properties. The calculation results contained in Process Table 4 have an important role in the next prediction calculation process. The similarity values obtained will be used as an important factor in calculating predictions regarding preferences or user behavior towards products that have not been consumed by them before. By using the similarity values generated from this calculation, we can make predictions about which products users are most likely to like based on

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

their preferences and behavior towards similar products. This helps in presenting recommendations that are more relevant and personalized in the context of the user experience.

2.4 Collaborative Filtering

The recommendation method is called collaborative filtering approach by recommending an item based on what other users have said about it. [25]. This method filter data derived from the similarity of client attributes to provide users with new information because the system gives them information based on a group of users who are almost the same[26]. Differences in interests in some group members make a new source of information. Cooperative sifting gives proposals in view of an assortment of feelings, interests also, interests of a few clients which are usually given in the form of ratings given by users on an item. Generally, collaborative filtering performs two processes[27]:

a. Embark on a quest to uncover users who exhibit a similar ranking pattern to the target user (the user for whom the prediction is to be made).
b. Use the rating values of other users obtained from the above step to calculate predictions for active users.

Collaborative filtering methodologies can be categorized into two distinct classes: client-based collaborative filtering, often referred to as memory-based, and item-based collaborative filtering, also known as model-based. User-based collaborative filtering hinges on the connection between a user and their corresponding group of followers. On the other hand, item-based collaborative filtering is built upon the relationships between items that have been previously accessed by the user, thereby facilitating the recommendation of other items based on past purchasing behavior. Recommendations based on users tend to select items that are like previously selected items. Examples of algorithms that are usually used in item-based CF are cosine-based likeness, pearson relationship comparability and changed cosine similitude [28]. In this research, the recommendations given are obtained in view of the comparability between things S seen from the ratings given by users for these item [29].

2.5 Adjusted Cosine Similarity Algorithm

Adjusted Cosine Similarity is a development of Cosine Similarity where there is consideration the existence empty values that are not owned by the user so that normalizing the data is fundamental to make the empty value data which is assumed to use 0 into a middle value [30]. This method is used to calculate the similarity of users who give different ratings[31]. For example, if an item is given a rating scale from 1 to 5, some users may give a rating of 5 to some items that they like considering the item is not too bad, while some other users, may give a rating value of 5 only for items that they really like. To overcome this difference in scale problem adjusted cosine method is used [32].

$$\operatorname{sim}\left(i_p, i_q\right) = \frac{\sum_{i=1}^{m} \left(R_{k,p} - R_k\right) \cdot \left(R_{k,q} - R_k\right)}{\sqrt{\sum_{i=1}^{m} \left(R_{k,p} - R_k\right)^2} \sqrt{\sum_{i=1}^{m} \left(R_{k,q} - R_k\right)^2}} \tag{4}$$

2.6 Predictions Calculations

After the calculation process to find the similarity value is carried out, then it will be continued with the prediction calculation process. The method that will be used in this study to find predictions for items is a weighted sum. The weighted sum method is a method for calculating item *i* predictions for user *u* by summing the rating given by the client on things that are like thing *I* [3].

$$P_{u,i} = \frac{\sum_{all\ similar\ items,N} \left(S_{i,N} * R_{u,N}\right)}{\sum_{all\ similar\ items,N} \left(\left|S_{i,N}\right|\right)} \tag{5}$$

It can be seen that $P_{u,i}$ is user u the basis for the anticipated rating of thing i, $S_{i,N}$ is the similarity of thing *i* and item *N*, $R_{u,N}$ is the rating that client u gave thing N.

2.7 Performance Evaluation

At this stage, a performance evaluation will be carried out to determine the effectiveness and performance of the recommendations that have been given by the system. To assess the presentation of the recommendation system, a method and, calculation are needed that can calculate the quality level of the predictions produced by a system.

2.8 Mean Absolute Error

The MAE method as widely recognized and employed in statistical analysis as a valuable tool for evaluating the accuracy of prediction systems. Its fundamental purpose is to assess the performance of a system by measuring the disparity between the predicted values and the corresponding actual values. By quantifying the absolute difference between these values, MAE provides a comprehensive understanding of the system's predictive capabilities. In the context of recommendation systems, MAE serves as a crucial metric for assessing the accuracy of predictions made by the system. When applying MAE to recommendation systems, the predicted values represent the system's

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

recommendations, while the original values correspond to the true preferences or ratings provided by users. By comparing the predicted recommendations with the actual user preferences, MAE offers insights into the level of accuracy exhibited by the recommendation system.

A lower MAE value indicates a higher degree of accuracy in predicting items and reflects the system's ability to provide recommendations that closely align with user preferences. It signifies that the system's predictions are consistently close to the actual values, thus enhancing the user experience and increasing the likelihood of user satisfaction. A lower MAE value signifies that the recommendation system has successfully captured the patterns and nuances of user behavior, enabling it to generate recommendations that resonate well with users' preferences and interests. By employing the MAE method, recommendation systems can be fine-tuning to optimize their predictive accuracy. System developers and researchers can analyze the MAE values obtained through testing and validation processes to gain insights into the system's strengths and weaknesses. Lower MAE values not only indicate superior accuracy but also serve as a benchmark for system performance comparison against alternative recommendation algorithms or variations in parameter settings.

In conclusion, the Mean Absolute Error (MAE) method plays a vital role in the evaluation of recommendation systems, providing a quantitative measure of the system's accuracy in predicting item preferences. A lower MAE value signifies higher accuracy, indicating that the system's recommendations closely align with the actual user preferences. By employing MAE as an evaluation metric, developers and researchers can assess and enhance the implementation of recommendations systems, ultimately improving the quality of recommendations and enhancing the overall user experience.

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

2.9 Normalized Discounted Cumulative Gain

The NDCG method as one of methods used to measure the quality of ratings in the list of recommendations provided to users. This method combines the concepts of item relevance and item ranking in the recommendation list, item relevance is usually given by the user or based on existing data and can be expressed on a numerical scale such as [0.5]. When a value is higher, it means that it is more important. The value of the NDCG itself ranges from [0.1], where a higher worth demonstrates a better rating and more relevant recommendations.

$$NDCG = \left. \sum_{i=1}^{n} \frac{relevance_i}{\log_2(i+1)} \middle/ \sum_{i=1}^{n} \frac{ideal_i}{\log_2(i+1)} \right. \tag{7}$$

3. RESULT AND DISCUSSION

In this study, the Run The Runway dataset was used from Amazon totaling 192,544 rows of data with a total of 15 feature columns. Based on the available column features, in this study the feature columns that will be used are only the User_ID, Item_ID, as well as a Rating with an original value of 1-10 and then, the rating value is converted to 1-5.

3.1 Data Exploration

Data exploration is performed in more depth to find out some errors such as missing values, outliers, duplication, encodings, noisy, and other errors in the data. The following is the initial description of the dataset obtained [33].

Table 5. Total Unique Value Dataset

Feature	Total
Unique Users	105508
Unique Items	5850
Unique Rating	[1 , 2 , 3 , 4 , 5]

Table 5 is the total dataset of unique values of users, items and ratings. Descriptive description is done to be able to see the distribution of data to reduce noise and data bias in the dataset, the results are obtained as in table 6.

Table 6. Agregat Data Item

Item_id	Average Rating	Total Rating
item_7	4,480036	2229
item_100	4,506961	1724
item_1	4,768827	1713
…	…	…
Item_5018	4,375000	32

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

item_1654	4,562500	32
item_1089	4,656250	32

Table 6 is an average rating, and the results of these items are quite significantly far from the average rating, in the section item_5018, item_1654, and item_1089 have the same total rating.

Table 7. Descriptive Statistics of Item Aggregate Data

	Average Rating	Total Rating
count	1455,000000	1455,000000
mean	4,563601	98,180756
std	0,191198	151,811228
min	3,488889	33,000000
25%	4,450432	41,500000
50%	4,588235	57,000000
75%	4,703704	91,500000
max	4,957746	2229,000000

Table 7 Is a descriptive statistic of aggregate data items that are calculated equally starting from the number, average, min, 25-75%, and Max. it will have different result values and the data is quite accurate.

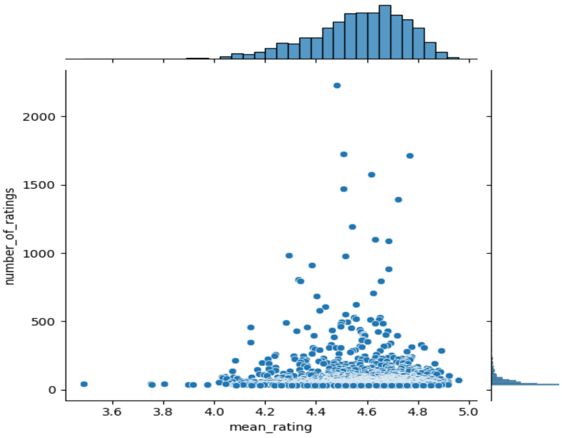

Figure 3. Visual Distribution of Item Data

Figure 3. Is the distribution of item data calculated using average data. And the value obtained is quite significant, namely at 4.0 to almost 5.0.

Table 8. Agregat Data User

User_id	Average Rating	Total Rating
user_72726	4,472727	165
user_3492	5,000000	91
user_87893	4,583333	72
…	…	…
user_32363	4,000000	10
user_10235	4,700000	10
user_51590	4,700000	10

Table 8 is the average rating and results from those users whose scores are quite significantly far from the average rating, but in the section user_32363, user_10235, and user_51590 have the same total rating.

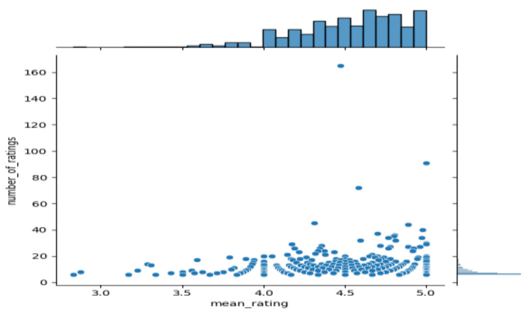

Figure 4. Visual Distribution of User Data

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

Figure 4 After thorough exploration, the Dataset is then converted into a Pivot-Table that has into a Pivot-Table that has User, Item, and Rating. This is done to make it easier for researchers to analyze the dataset used more efficiently and to be ready to compute the Likeness esteem in Collaborative. more efficiently and to be ready to compute the Likeness esteem in Collaborative Filtering. Filtering requires a dataset with a Pivot-Table representation. The outcomes of the PivotTable as shown in Table 9.

Table 9. Pivot Table

Item ID	User_974	User_1810	…	User_103895	User_104887
item_0	4,0	0,0	…	0,0	0,0
item_1	0,0	3,0	…	0,0	0,0
…	…	…	…	…	…
item_5826	0,0	0,0	…	0,0	5,0
item_5827	0,0	0,0	…	5,0	0,0

Table 9 Similarity calculation on Adjusted Cosine Similarity method, Pivot-Table that has been owned needs to be converted into the form of Mean-Adjusted Rating Pivot-Table by reducing each rating to the average User Rating, this is done to eliminate individual user bias against existing ratings. eliminate individual user bias against existing ratings.

Table 10. Mean-Adjusted Pivot Table

Item ID	User_974	User_1810	…	User_103895	User_104887
item_0	-0,222	0,0	…	0,0	0,0
item_1	0,0	-1,167	…	0,0	0,0
…	…	…	…	…	…
item_5826	0,0	0,0	…	0,0	0,2
item_5827	0,0	0,0	…	0,273	0,0

Table 10 is a Pivot-Table that has been owned needs to be converted into a Mean-Adjusted Rating Pivot-Table form by reducing each Rating to the average User Rating.

3.2 Collaborative Filtering

Adjusted Cosine Similarity calculation to see Item-Based similarity, Adjusted Cosine Similarity has a range of results of -1 to 1 where when the worth is near 1 then the thing can be said to be related while when the worth is near - 1 then the thing can be said to be unrelated [34].

Table 11. Adjusted Cosine Similarity

Item ID	Item_0	Item_1	Item_2	…	Item_5826	Item_5827
item_0	1,000	0,0	-0,018258	…	0,0	0,0
item_1	0,0	1,000	0,034696	…	0,008143	0,0
…	…	…	…	…	…	…
item_5826	0,0	0,021734	…	…	1,000	0,0
item_5827	0,0	0,0	…	…	0,0	1,000

Table 11 Item-Based similarity, Adjusted Cosine Similarity has a range of results of -1 to 1 where when items can be said to be related when the value is close to 1, not related when the value is close to 0, and conflicting when the value is close to -1.

3.2 Evaluation Result

The data is evaluated to see the MAE value by applying several test scenarios in the form of different numbers of missing values, different numbers of Top-n Neighborhood Selection and researchers will compare the test results that are cosine-similar to see the performance of Adjusted Cosine Relativity. The following are the test results obtained:

Table 12. Comparison of the results of cosine similarity and adjusted cosine similarity calculated using MAE and NDCG

K-Fold	Top-n	MAE		NDCG	
		Cosine Similarity	Adjusted Cosine Similarity	Cosine Similarity	Adjusted Cosine Similarity
4	5	0,4774	0,4461	0,9905	0,9989
5	5	0,4774	0,4461	0,9900	0,9988
10	5	0,4774	0,4468	0,9887	0,9887
4	10	0,4815	0,4424	0,9921	0,9989
5	10	0,4815	0,4424	0,9919	0,9989

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

10	10	0,4815	0,4424	0,9908	0,9988

Based on the extensive analysis and evaluation of the results presented in Table 12, it becomes evident that Adjusted Cosine Similarity, specifically when the value of Top-n is set to 10, consistently outperforms the Cosine Similarity method across various evaluation metrics. The MAE (Mean Absolute Error) value obtained using Adjusted Cosine Similarity is recorded at an impressively low value of 0.4424, indicating the model's ability to accurately predict user preferences for unrated items. Additionally, the NDCG (Normalized Discounted Cumulative Gain) value achieved is an exceptional 0.9988, further solidifying the advantage of the Cosine Similarity Adjusted approach.These outstanding results provide substantial evidence to support the claim that the utilization of Adjusted Cosine Similarity yields superior recommendations compared to the traditional Cosine Similarity method. By incorporating adjustments that account for the variations in user behavior and item similarities, Adjusted Cosine Similarity can catch the intricacies can catch the more effectively. This enhanced understanding of user behavior, coupled with the comprehensive consideration of item similarities, results in more accurate and relevant recommendations being generated.

The superior performance of Adjusted Cosine Similarity can be attributed to its ability to mitigate the limitations associated with the Similarity of Cosine method. Cosine Relativity solely relies on the angle's cosine. between the vectors representing user-item interactions, disregarding other crucial factors such as the overall rating patterns, biases, and deviations within the dataset. On the other hand, Adjusted Cosine Similarity successfully addresses these limitations by normalizing the ratings based on user and item averages, ensuring a fair and unbiased comparison. This normalization process allows the model to provide more precise recommendations, enhancing the overall user experience and satisfaction.

In conclusion, the extensive evaluation of the results in Table 12 clearly demonstrates the superior performance of Cosine Similarity Adjusted over the Cosine Likeness strategy. The exceptional MAE value of 0.4424 and the impressive NDCG of 0.9988 achieved using the Adjusted Cosine Similarity approach further emphasize its effectiveness in generating accurate and relevant recommendations. By incorporating adjustments that consider user behavior and item similarities, Adjusted Cosine Similarity overcomes the limitations of the traditional Cosine Similarity method, resulting in improved recommendation quality and user satisfaction.

4. CONCLUSION

Based from the outcomes of experimentation conducted on Collaborative Filtering Item-Based utilizing both Cosine Similarity and Adjusted Cosine Similarity. Adjusted Cosine Similairy results with Top-n Neighbors = 10 produce the best performance in each scenario with an average MAE value of 0.44242 at 4-Folds, 0.44246 at 5-Folds, and 0.44245 at 10-Folds. This proves that Adjusted Cosine Similarity can outperform Cosine Similarity by higher MAE difference of 0.03493 at 10-Folds, 0.03492 at 5-Folds, and 0.03495 at 4-Folds. In Table 10, Adjusted Cosine Similarity produces more consistent item recommendations on Top-n = 5 and Top-n = 10, where in 3 trials each user gets a recommendation for items that are more consistent with Top-n = 5 and Top-n = 10. 10, where in 3 trials each user gets the same item recommendation. While the recommendation results from Cosine Similarity experience item changes. Based on the explanation above, it can be concluded that the use of Adjusted Cosine Similarity can produce better recommendations because it not only considers user behavior, but also produces lower error performance. because it not only considers the behavior of the user, but also produces lower error performance.

REFERENCES

[1] Y. Li, S. Wang, Q. Pan, H. Peng, T. Yang, and E. Cambria, “Learning binary codes with neural collaborative filtering for efficient recommendation systems,” *Knowledge-Based Syst.*, vol. 172, pp. 64–75, 2019, doi: 10.1016/j.knosys.2019.02.012.

[2] Arvin Jay C Nadal, Angelo S Reyes, Mark Christoper R Blanco, Antolin J Alipio, and Dan Michael A Cortez, “Enhancement of collaborative filtering using myers-briggs type indicator (mbti) applied in recommendation system,” *South Asian J. Eng. Technol.*, vol. 12, no. 1, pp. 104–110, 2022, doi: 10.26524/sajet.2022.12.016.

[3] L. Chen, L. Wu, R. Hong, K. Zhang, and M. Wang, “Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach,” *AAAI 2020 - 34th AAAI Conf. Artif. Intell.*, pp. 27–34, 2020, doi: 10.1609/aaai.v34i01.5330.

[4] C. Chen, M. Zhang, Y. Zhang, W. Ma, Y. Liu, and S. Ma, “Efficient Heterogeneous Collaborative Filtering without negative sampling for recommendation,” *AAAI 2020 - 34th AAAI Conf. Artif. Intell.*, pp. 19–26, 2020, doi: 10.1609/aaai.v34i01.5329.

[5] X. Wang, R. Wang, C. Shi, G. Song, and Q. Li, “Multi-component graph convolutional collaborative filtering,” *AAAI 2020 - 34th AAAI Conf. Artif. Intell.*, pp. 6267–6274, 2020, doi: 10.1609/aaai.v34i04.6094.

[6] S. Benkessirat, N. Boustia, and R. Nachida, “A New Collaborative Filtering Approach Based on Game Theory for Recommendation Systems,” *J. Web Eng.*, vol. 20, no. 2, pp. 303–326, 2021, doi: 10.13052/jwe1540-9589.2024.

[7] M. Ammad-ud-din *et al.*, “Federated Collaborative Filtering for Privacy-Preserving Personalized Recommendation System,” 2019, [Online]. Available: http://arxiv.org/abs/1901.09888.

[8] C. Shi *et al.*, “Deep Collaborative Filtering with Multi-Aspect Information in Heterogeneous Networks,” *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1413–1425, 2021, doi: 10.1109/TKDE.2019.2941938.

[9] L. Xia, C. Huang, Y. Xu, J. Zhao, D. Yin, and J. Huang, “Hypergraph Contrastive Collaborative Filtering,” *SIGIR 2022 -*

Building of Informatics, Technology and Science (BITS)
Volume 5, No 2, September 2023 Page: 376−385
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i2.3690

Proc. 45th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr., pp. 70–79, 2022, doi: 10.1145/3477495.3532058.

[10] W. Sun, P. Shafto, and W. Sun, "ThinkIR : The University of Louisville ' s Institutional Repository Debiasing the Human-Recommender System Feedback Loop in Collaborative Filtering Debiasing the Human-Recommender System Feedback Loop in Collaborative Filtering," 2019.

[11] S. Rendle, W. Krichene, L. Zhang, and J. Anderson, "Neural Collaborative Filtering vs. Matrix Factorization Revisited," *RecSys 2020 - 14th ACM Conf. Recomm. Syst.*, pp. 240–248, 2020, doi: 10.1145/3383313.3412488.

[12] D. Ferreira, S. Silva, A. Abelha, and J. Machado, "Recommendation system using autoencoders," *Appl. Sci.*, vol. 10, no. 16, pp. 1–17, 2020, doi: 10.3390/app10165510.

[13] F. Xue, X. He, X. Wang, J. Xu, K. Liu, and R. Hong, "Deep item-based collaborative filtering for top-N recommendation," *ACM Trans. Inf. Syst.*, vol. 37, no. 3, 2019, doi: 10.1145/3314578.

[14] M. Singh Kumar and R. Prakas, "EVENT DRIVEN RECOMMENDATION SYSTEM FOR E-COMMERCE USING KNOWLEDGE BASED COLLABORATIVE FILTERING TECHNIQUE," vol. 301, no. 3, pp. 2014–2016, 2020, doi: 10.12694:/scpe.v21i3.1709.

[15] P. Valdiviezo-Diaz, F. Ortega, E. Cobos, and R. Lara-Cabrera, "A Collaborative Filtering Approach Based on Naïve Bayes Classifier," *IEEE Access*, vol. 7, pp. 108581–108592, 2019, doi: 10.1109/ACCESS.2019.2933048.

[16] H. Tang, G. Zhao, X. Bu, and X. Qian, "Dynamic evolution of multi-graph based collaborative filtering for recommendation systems," *Knowledge-Based Syst.*, vol. 228, p. 107251, 2021, doi: 10.1016/j.knosys.2021.107251.

[17] J. Bobadilla, S. Alonso, and A. Hernando, "Deep learning architecture for collaborative filtering recommender systems," *Appl. Sci.*, vol. 10, no. 7, 2020, doi: 10.3390/app10072441.

[18] C. Feng, J. Liang, P. Song, and Z. Wang, "A fusion collaborative filtering method for sparse data in recommender systems," *Inf. Sci. (Ny).*, vol. 521, pp. 365–379, 2020, doi: 10.1016/j.ins.2020.02.052.

[19] L. Wu, X. He, X. Wang, K. Zhang, and M. Wang, "A Survey on Accuracy-Oriented Neural Recommendation: From Collaborative Filtering to Information-Rich Recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4425–4445, 2023, doi: 10.1109/TKDE.2022.3145690.

[20] W. Chen, F. Cai, H. Chen, and M. D. E. Rijke, "Joint neural collaborative filtering for recommender systems," *ACM Trans. Inf. Syst.*, vol. 37, no. 4, pp. 1–30, 2019, doi: 10.1145/3343117.

[21] K. R. Dwi, "Sistem Rekomendasi Komunitas Pemuda Di Kota Semarang Berbasis Item Based Collaborative Filtering Dengan Metode Adjusted Cosine," *Tek. Inform.*, vol. 3, p. 8, 2018.

[22] A. GUNAWAN, "Implementasi Sistem Rekomendasi Produk Dengan Metode Item-Based Collaborative Filtering Berbasis Algoritma Adjusted Cosine …," 2022.

[23] M. F. Aljunid and M. Dh, "An Efficient Deep Learning Approach for Collaborative Filtering Recommender System," *Procedia Comput. Sci.*, vol. 171, no. 2019, pp. 829–836, 2020, doi: 10.1016/j.procs.2020.04.090.

[24] M. Srifi, A. Oussous, A. A. Lahcen, and S. Mouline, "Recommender systems based on collaborative filtering using review texts-A survey," *Inf.*, vol. 11, no. 6, pp. 1–21, 2020, doi: 10.3390/INFO11060317.

[25] X. Wang, H. Jin, A. Zhang, X. He, T. Xu, and T. S. Chua, "Disentangled Graph Collaborative Filtering," *SIGIR 2020 - Proc. 43rd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 1001–1010, 2020, doi: 10.1145/3397271.3401137.

[26] I. W. Jepriana and S. Hanief, "Collaborative Filtering Untuk Sistem Rekomendasi Konsentrasi Di Stmik Stikom Bali," vol. 9, pp. 171–180, 2020.

[27] S. Natarajan, S. Vairavasundaram, S. Natarajan, and A. H. Gandomi, "Resolving data sparsity and cold start problem in collaborative filtering recommender system using Linked Open Data," *Expert Syst. Appl.*, vol. 149, 2020, doi: 10.1016/j.eswa.2020.113248.

[28] X. Wang, X. He, M. Wang, F. Feng, and T. S. Chua, "Neural graph collaborative filtering," *SIGIR 2019 - Proc. 42nd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 165–174, 2019, doi: 10.1145/3331184.3331267.

[29] Anderias Eko Wijaya and Deni Alfian, "Sistem Rekomendasi Laptop Menggunakan Collaborative Filtering Dan Content-Based Filtering," *J. Comput. Bisnis*, vol. 12, no. 1, pp. 11–27, 2018.

[30] S. Saha *et al.*, "Feature selection for facial emotion recognition using cosine similarity-based harmony search algorithm," *Appl. Sci.*, vol. 10, no. 8, pp. 1–22, 2020, doi: 10.3390/APP10082816.

[31] S. C. Mana and T. Sasipraba, "Research on cosine similarity and pearson correlation based recommendation models," *J. Phys. Conf. Ser.*, vol. 1770, no. 1, 2021, doi: 10.1088/1742-6596/1770/1/012014.

[32] J. Yang, Y. Li, W. Cheng, Y. Liu, and C. Liu, "EKF-GPR-Based fingerprint renovation for subset-based indoor localization with adjusted cosine similarity," *Sensors (Switzerland)*, vol. 18, no. 1, 2018, doi: 10.3390/s18010318.

[33] Z. Wang *et al.*, "NeuralCubes: Deep Representations for Visual Data Exploration," *Proc. - 2021 IEEE Int. Conf. Big Data, Big Data 2021*, pp. 550–561, 2021, doi: 10.1109/BigData52589.2021.9671390.

[34] R. Langner, T. Horak, and R. Dachselt, "Demonstrating vistiles: Visual data exploration using mobile devices," *Proc. Work. Adv. Vis. Interfaces AVI*, pp. 1–3, 2018, doi: 10.1145/3206505.3206583.