Journal of Information Systems and Informatics
Vol. 6, No. 3, September 2024 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v6i3.802 Published By DRPM-UBD

Collaborative Filtering Recommendation System Using A Combination of Clustering and Association Rule Mining

Siti Annisa[1], Dian Palupi Rini[2], Abdiansah[3]

[1,2,3]Department of Computer Science, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia
Email: [1]sitiannisa912@gmail.com, [2]dprini@unsri.ac.id, [3]abdiansah@unsri.ac.id

Abstract

A recommendation system helps collect and analyze user data to generate personalized recommendations for users. A recommendation system for movies has been implemented, considering the vast number of available films and the difficulty users face in finding movies that match their interests. One popular recommendation method is Collaborative Filtering (CF). Although widely applied, CF still has issues. Basic CF uses overlapping user data in evaluating items to calculate user similarity. This study aims to build a collaborative filtering recommendation system using clustering techniques to group users with similar interests into the same clusters. The next step in CF application is to gather recommendation candidate items by finding users with a high level of similarity to the target user. Subsequently, user pattern analysis is carried out by applying association rule mining to predict hidden correlations based on frequently watched items and the ratings given to those movies. This study uses rating data and movie data from the Movielens website. The evaluation of the recommendation results is measured using precision, recall, and f-measure. The evaluation results show that the proposed recommendation system achieves a hit rate of 95.08%, a precision of 81.49%, a recall of 98.06%, and an f-measure of 87.66%.

Keywords: Recommendation System, Clustering, Collaborative Filtering, Association Rule Mining

1. INTRODUCTION

Recommendation systems have been implemented on various websites, such as Amazon, Moviefinder, Netflix, and others. [1]. Netflix is one of the video streaming platforms that adopts data analysis and machine learning technologies to understand user behavior and preferences, enabling it to provide more relevant and personalized movie recommendations. [2].

A recommendation system for movies has been implemented, considering the vast number of available films and the difficulty users face in finding films that match their interests. Different users will have varying preferences for films or actors [3]. One popular recommendation system method is Collaborative

Filtering (CF) [4]. Collaborative Filtering uses user ratings to calculate the similarity between users or items, then makes predictions or recommendations based on the calculated similarity scores. [5]. However, several issues arise in the application of collaborative filtering, such as data sparsity, cold starts, shilling attacks, accuracy, and efficiency. [6]. Problems in collaborative filtering often occur when users rate only a few items, making it difficult to calculate similarities between users. [7]. Collaborative filtering can produce poor recommendations when user ratings for items are sparse compared to the large number of users and items in the user-item matrix. [8].

Krisdhamara et al. conducted research on e-commerce site recommendation systems using the CF method, considering user reviews of items. The research used model-based CF by applying the k-means++ algorithm to produce better quality clusters. This was followed by the application of opinion mining to filter recommendation items, taking into account user reviews of the purchased items. [9].

The research by Obeidat et al. discusses a CF recommendation system that recommends online courses to students based on the similarity of other students' course histories [10]. The data mining technique used involves applying a clustering algorithm. Following that, the association rule method using the apriori algorithm is applied to each cluster. These rules are used to recommend subjects to students based on similarities within the cluster.

This study aims to develop a collaborative filtering recommendation system for movies by combining clustering and association rule mining methods. The k-means algorithm is utilized in the clustering method to reduce the size and dimensionality of the data. [4]. This algorithm helps group users with similar interests into the same cluster and recommends items that users may like [11]. Subsequently, user pattern analysis is conducted by applying association rule mining with the fp-growth algorithm to predict hidden correlations based on frequently watched items.

2. METHODS

The proposed method is carried out in several stages, as shown in Figure 1. Based on Figure 1, the steps involved in implementing the proposed method include data collection, data preparation—from data cleaning to data preprocessing. Once the data is ready for use, the proposed method is implemented.

In this study, the proposed data processing methods are expected to provide a measurement model in the item selection process for the CF recommendation

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

system. The proposed method aims to recommend movies based on user preference similarity. First, the clustering method proposed in this study is used to group user based on genre and rating. Movies with similar genres and ratings are grouped into the same cluster, while different movies are separated into different clusters. The clustering algorithm used in this study is k-means. After the clusters are formed, the CF method is used to calculate the similarity between the target user and other users, resulting in a list of candidate recommendation items. Subsequently, the data mining technique of Association Rule Mining using the FP-Growth algorithm will be implemented to filter the candidate items. The implementation of Association Rule Mining is used to generate recommendations that remain aligned with user preferences. Finally, an evaluation process is conducted to assess the performance of the applied method.

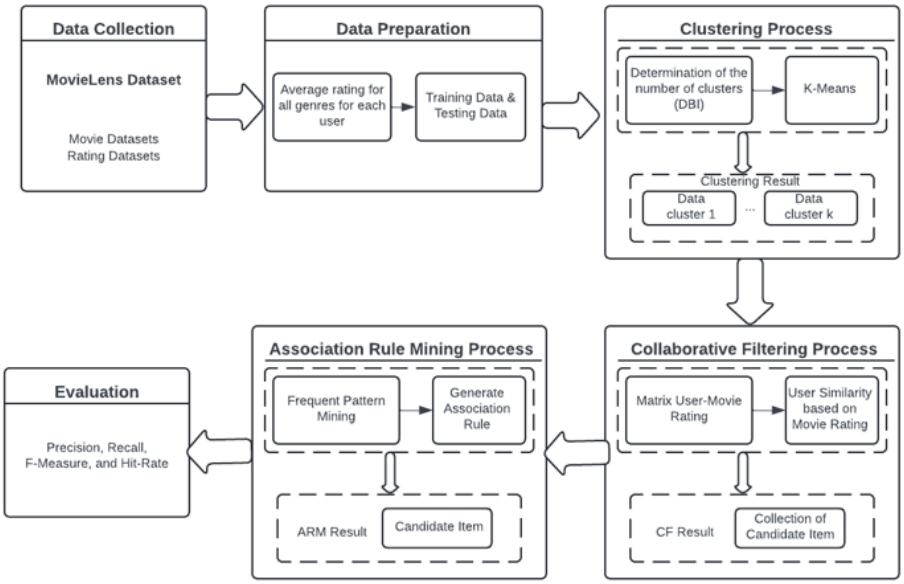

Figure 1. Proposed Recommendation System

2.1. Dataset

Table 1. present a summary of the movie dataset used in this study. This dataset comprises the attributes movieid, title, and genre. Userid refers to a randomly selected MovieLens user included in the dataset, with the userid anonymized. Movieid represents only movies that have at least one rating or tag included in the dataset. Rating refers to the assessment given by a user for a movie they have watched, with ratings provided on a 5-star scale. Timestamps contain the corresponding time codes.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Tabel 1. Summary of Movie Dataset

	Movieid	Title	Genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
…	…	…	…
9739	193585	Flint (2017)	Drama
9740	193587	Bungo Stray Dogs: Dead Apple (2018)	Action|Animation
9741	193609	Andrew Dice Clay: Dice Rules (1991)	Comedy

Table 2 presents a summary of the rating dataset used in this study. This dataset consists of the attributes userid, movieid, and rating.

Table 2. Summary of Rating Dataset

	Userid	Movieid	Rating
0	1	1	4.0
1	1	3	4.0
2	1	6	4.0
…		…	…
9740	610	168252	5.0
9741	610	170875	3.0

2.2. Data Preprocessing

In the preprocessing stage, data without values were cleaned. The cleaning of the rating and movie data involved removing entries that did not have a genre or rating. The next stage involved calculating the average rating for all genres for each user. Information about the average rating per genre for each user can be used to group users into more homogeneous clusters. Users with similar genre preferences can be grouped together, resulting in more accurate recommendations.

2.3. Training Data dan Testing Data

In this study, the test data comprises a set of users who have a history of providing ratings for movies, which will serve as the target for recommendations. The target recommendation is the user who will receive movie recommendations based on their previously watched history. There are 610 users, which will be divided into training and test data. Ten percent of the data is selected as test data, resulting in 61 test data points and 549 training data points. The 61 users in the

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

test data will be the target for recommendations, and the system will provide them with recommended movie items.

2.4. Clustering

Clustering is a machine learning technique that falls under the category of unsupervised learning, and it groups information (observations or datasets) based on similarity measures [9]. In this study, the clustering algorithm used is k-means. Clusters are formed based on user similarity, calculated by considering the similarity between genres and ratings for each film. In a collaborative filtering recommendation system, users can be considered as objects to be clustered. Therefore, when a new user is identified as being similar to a particular cluster (or group of users) through a clustering algorithm, the items favored by that user group are then recommended to the new user [12]. The following are the steps of the clustering method using the K-means algorithm [13] :

1) Determine the value of k, where k represents the number of clusters to be formed.
2) Determine the initial centroid for each cluster, with centroids being chosen randomly.
3) Allocate all data to the nearest cluster by calculating the distance of each data point to each centroid. The distance between all data points and each centroid can be calculated using the Euclidean distance theory with Equation 1:

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \quad (1)$$

Explanation:
$d(x,y)$= the distance of data x to the cluster center u
x_i=data x at the i-th observation
y_i=the center point y of the i-th observation
n=the number of observations

4) Group each data point into the cluster with the shortest distance.
5) Determine the centroid value by calculating the average value of the cluster for each cluster member using Equation 2:

$$centroid=\sum\frac{a_i}{n} \quad (2)$$

a_i=the membership value of each cluster
n=the number of cluster members

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

6) Repeat steps 2-5 until convergence is achieved, i.e., when the members of each cluster no longer change their cluster location.

2.5. Collaborative Filtering

The next stage is to gather candidate recommendation items. The collaborative filtering method begins by identifying users who have a high similarity to the target user. For each user within a cluster, the similarity to the target user is calculated. This process involves searching for transactions that most closely resemble the target transaction. The similarity measure is calculated based on the total number of items that match the target items. User similarity is calculated using cosine similarity between the target user and all other users within the same cluster, as shown in Equation 3.

$$Similarity\ (A,B)=\frac{A.B}{\|A\|\times\|B\|} \tag{3}$$

2.6. Association Rule Mining

The association rule mining algorithm extracts rules that predict the occurrence of an item based on the presence of other items in a transaction [14]. There are 3 (three) main stages in the FP-Growth method:

a) Generation of the conditional pattern base
b) Generation of the conditional FP-Tree
c) Searching for frequent itemset

Basic methodology of association analysis is divided into two stages [15]:

1) Frequent Itemset
This stage searches for item combinations that meet the minimum support threshold in the database. The support of an association rule $X \rightarrow Y$ is defined by Equation 4 and 5.

$$Support\ (X)=\frac{Number\ of\ transactions\ which\ contain\ X}{Number\ of\ all\ transactions} \tag{4}$$

The support value of 2 items is obtained from:

$$Support\ (X\rightarrow Y)=\frac{\Sigma Number\ of\ transactions\ which\ contain\ X\ and\ Y}{\Sigma Number\ of\ all\ transactions\ in\ database} \tag{5}$$

2) Association Rule
After all Frequent Itemsets have been identified, association rules that meet the minimum confidence criterion are then searched by calculating

the confidence of the associative rule $x \rightarrow y$. The confidence of an association rule $X \rightarrow Y$ is defined by Equation 6.

$$Confidence\ (X \rightarrow Y) = \frac{\sum Number\ of\ transactions\ which\ contain\ X\ and\ Y}{\sum Number\ of\ transactions\ which\ contain\ X} \quad (6)$$

Confidence is the strength of the relationship between items in association rules.

2.7. Evaluation Metrics

Table 3 illustrates that the Confusion matrix is used to depict the performance of the classification model. The Confusion matrix below generates True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) values, which will be used to calculate precision and recall scores.

Table 3. Confusion matrix

		Actual	
		Positive	**Negative**
Predicted	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

The matrix used to evaluate the performance of the research method includes precision, recall, and f-measure [4]. F-measure is defined as the average of precision (P) and recall (R). Precision is the number of true positive predictions divided by the total predicted positive results. Recall is the number of true positive predictions divided by all relevant test values. Precision and recall are sufficient to indicate the condition of recommendation results using Equation 7 and 8.

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

F-measure is defined by Equation 7:

$$F\text{-}Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (9)$$

In the context of recommendation systems, Hit Rate is a binary metric, determining whether relevant items are present in the recommendation list for the user. [16]. The higher the hit rate, the better the quality of the generated recommendations. The hit rate value implies that to build a high-quality

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

recommendation system, each recommendation target must receive at least one relevant item that aligns with the target interest [17]. Hit rate is measured using Equation 8.

$$Hit\ Rate = \frac{n\ hits}{n\ users} \tag{10}$$

3. RESULTS AND DISCUSSION

This section presents the results of the research conducted on the collaborative filtering recommendation system using a combination of clustering and association rule mining methods. The evaluation of the research results utilizes a confusion matrix with the parameters hit-rate, precision, recall, and f-measure.

3. 1. Clustering

In the clustering method, the training data is used to train the algorithm. The trained algorithm's results are then used to predict clusters in the test data. To determine the optimal number of clusters from the k-means results, a search for the best cluster variance value is conducted using the Davies-Bouldin Index (DBI) method.

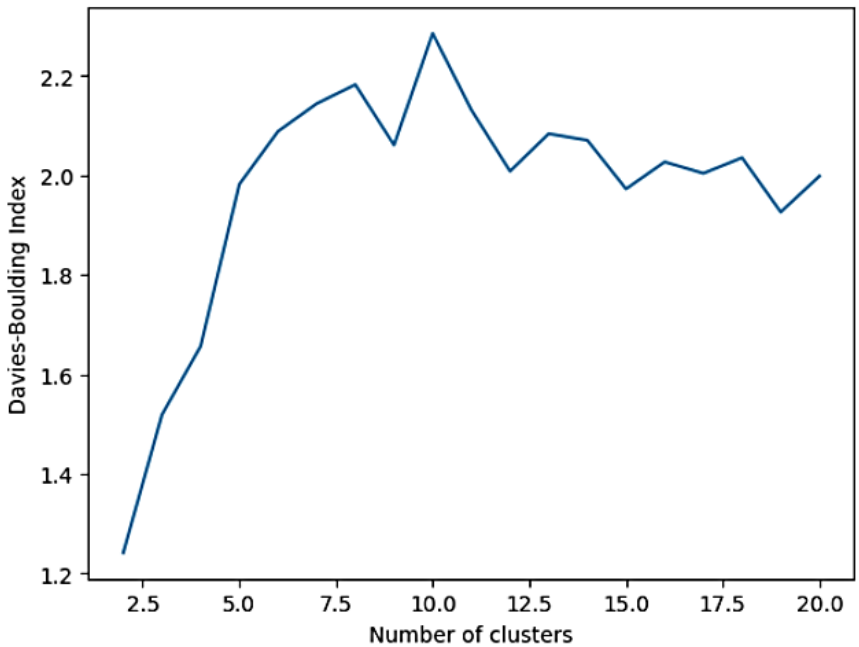

Figure 2. Number of Clusters

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Figure 2 illustrates the variance value changes across different cluster numbers resulting from the applied k-means. Based on the graph, the optimal number of clusters is found to be 2 clusters.

3. 2. Collaborative Filtering

The collaborative filtering approach employed in this study relies heavily on constructing a user-movie rating matrix and calculating user similarity scores, as illustrated in Figures 3 and 4.

```
===== Matrix User Movie Ratings =====
movieId  1       2       3       4       5       6       7       8       \
userId
1          4.0     0.0     4.0     0.0     0.0     4.0     0.0     0.0
4          0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
5          4.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
8          0.0     4.0     0.0     0.0     0.0     0.0     0.0     0.0
11         0.0     0.0     0.0     0.0     0.0     5.0     0.0     0.0

movieId  9       10      ...  193565  193567  193571  193573  193579  193581
userId                   ...
1          0.0     0.0   ...     0.0     0.0     0.0     0.0     0.0     0.0
4          0.0     0.0   ...     0.0     0.0     0.0     0.0     0.0     0.0
5          0.0     0.0   ...     0.0     0.0     0.0     0.0     0.0     0.0
8          0.0     2.0   ...     0.0     0.0     0.0     0.0     0.0     0.0
11         0.0     3.0   ...     0.0     0.0     0.0     0.0     0.0     0.0

movieId  193583  193585  193587  193609
userId
1          0.0     0.0     0.0     0.0
4          0.0     0.0     0.0     0.0
5          0.0     0.0     0.0     0.0
8          0.0     0.0     0.0     0.0
11         0.0     0.0     0.0     0.0
```

Figure 3. User-Movie Rating Matrix

Figure 3 displays the user-movie rating matrix, which serves as the core dataset for the collaborative filtering process. This matrix represents the ratings provided by different users (userId) for various movies (movieId). The entries in the matrix range from numerical values representing the users' ratings to zeroes where no ratings have been given. For example, userId 1 has rated some movies, such as giving a rating of 4.0 to movieId 1 and movieId 4, while other movies remain unrated (0.0). This matrix is fundamental in identifying user preferences, allowing the recommendation system to analyze patterns in how different users rate the same movies.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

```
===== User Similarity Based on Movie Ratings =====
userId       1         4         5         8         11        12        13
userId
1       1.000000  0.194395  0.129080  0.136968  0.132499  0.016458  0.092971
4       0.194395  1.000000  0.128659  0.062969  0.054767  0.049945  0.076949
5       0.129080  0.128659  1.000000  0.429075  0.183805  0.058860  0.017157
8       0.136968  0.062969  0.429075  1.000000  0.235334  0.046195  0.054417
11      0.132499  0.054767  0.183805  0.235334  1.000000  0.031898  0.030579

userId       16        17        18   ...       587       589       592  \
userId                                 ...
1       0.169858  0.264358  0.214868  ...  0.122782  0.118112  0.143934
4       0.164761  0.145058  0.123217  ...  0.138766  0.064485  0.068196
5       0.082171  0.162633  0.121313  ...  0.154315  0.230961  0.359595
8       0.114719  0.199996  0.152172  ...  0.215067  0.392964  0.513471
11      0.053845  0.171089  0.150305  ...  0.054532  0.224502  0.332035

userId       594       595       597       601       606       607       610
userId
1       0.141960  0.110558  0.312843  0.080554  0.164191  0.269389  0.145321
4       0.082614  0.091974  0.275436  0.085938  0.200395  0.131746  0.107683
5       0.137990  0.073238  0.116071  0.068048  0.106435  0.152866  0.060792
8       0.222126  0.051494  0.149305  0.080203  0.099388  0.185142  0.078153
11      0.225668  0.026853  0.213639  0.083663  0.072988  0.269857  0.087531
```

Figure 4. User Similarity Matrix Based on Movie Ratings

Figure 4 presents the user similarity matrix, which is derived from the user-movie rating data shown in Figure 3. This matrix quantifies how similar users are to one another based on their movie ratings. The similarity between two users is measured using techniques such as cosine similarity or Pearson correlation, resulting in a score that indicates the degree of similarity between their movie preferences. For example, the diagonal of the matrix (e.g., "1.000000" for userId 1 with itself) indicates perfect similarity, as a user is always completely similar to themselves. However, the off-diagonal values, such as "0.194395" between userId 1 and userId 4, reflect the similarity scores between different users. Higher similarity scores indicate stronger alignment in movie preferences, while lower scores suggest less similarity.

These similarity scores enable the recommendation system to group users with similar movie tastes. For instance, users with high similarity scores (e.g., userId 5 and userId 8 with a score of 0.429075) are more likely to have overlapping movie preferences, making it easier for the system to recommend movies to one user based on the ratings of the other. Conversely, lower similarity scores (e.g., "0.016458" between userId 1 and userId 12) suggest that these users have different preferences, and therefore, less relevance in terms of cross-recommendations.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3. 3. Association Rule Mining

Figure 5 presents the results of association rule mining, consisting of two main sections: Frequent Itemsets and Rules. These results provide insights into which combinations of items frequently occur together in the dataset and the strength of the associations between these items.

```
===== Frequent Itemsets =====
     support           itemsets
314      0.5  (1200, 1387, 589)
315      0.5  (1200, 1387, 260)
316      0.5   (1387, 260, 589)
317      0.5  (1387, 1291, 589)
318      0.5  (1387, 2028, 589)
319      0.5  (480, 1387, 2028)
320      0.5   (480, 1387, 589)
321      0.5       (1097, 1291)
322      0.5        (480, 1097)
323      0.5        (1097, 589)
============================
===== rules =====
      antecedents  consequents  antecedent support  consequent support  '
990   (480, 1387)       (2028)                0.50                0.80
991   (480, 2028)       (1387)                0.70                0.55
992  (1387, 2028)        (480)                0.50                0.85
993        (1387)  (480, 2028)                0.55                0.70
994   (480, 1387)        (589)                0.50                0.90
995   (1387, 589)        (480)                0.55                0.85
996        (1387)   (480, 589)                0.55                0.75
997        (1097)       (1291)                0.55                0.75
998        (1097)        (480)                0.55                0.85
999        (1097)        (589)                0.55                0.90

     support  confidence      lift  leverage  conviction  zhangs_metric
990      0.5    1.000000  1.250000    0.1000         inf       0.400000
991      0.5    0.714286  1.298701    0.1150       1.575       0.766667
992      0.5    1.000000  1.176471    0.0750         inf       0.300000
993      0.5    0.909091  1.298701    0.1150       3.300       0.511111
994      0.5    1.000000  1.111111    0.0500         inf       0.200000
995      0.5    0.909091  1.069519    0.0325       1.650       0.144444
996      0.5    0.909091  1.212121    0.0875       2.750       0.388889
997      0.5    0.909091  1.212121    0.0875       2.750       0.388889
998      0.5    0.909091  1.069519    0.0325       1.650       0.144444
999      0.5    0.909091  1.010101    0.0050       1.100       0.022222
Banyaknya rekomendasi ARM: 23
[(1200, (4.775997520635277, 14)), (260, (4.688891184025058, 15)), (1198,
```

Figure 5. Generated Rules

The Frequent Itemsets section in Figure 5 lists groups of items that frequently appear together in the dataset, along with their support values. Each itemset represents a combination of items, and its support value indicates the proportion of transactions in which this specific combination appears. For example, the itemset {1200, 1387, 589} has a support value of 0.5, meaning it appears in 50% of the total transactions in the dataset. Several other itemsets also have a support value of 0.5, showing that these combinations are present in half of the recorded transactions. This frequent occurrence suggests that these items are commonly purchased together, which can be valuable information for decision-making, such as product placement, inventory management, and promotional strategies.

The Rules section in Figure 5 shows the associative rules derived from the frequent itemsets, accompanied by various metrics that measure their strength and quality. Each rule consists of an antecedent (the "if" part) and a consequent (the "then" part), indicating a relationship between the itemsets.
For example, the rule {480, 1387} → {2028} has the following metrics:

1) Support: 0.5, meaning this rule applies to 50% of the total transactions, indicating that both the antecedent and the consequent occur together in half of all transactions.
2) Confidence: 1.0, which shows that whenever items {480, 1387} appear in a transaction, item {2028} also always appears. This indicates a perfect association between the antecedent and consequent.
3) Lift: 1.25, indicating that this rule occurs 1.25 times more frequently than it would if {480, 1387} and {2028} were independent of each other. A lift greater than 1 suggests that the presence of the antecedent increases the likelihood of the consequent, making the rule a valuable insight for targeted marketing or cross-selling strategies.

Other rules are also presented with varying support, confidence, lift, leverage, conviction, and Zhang's metric, providing a comprehensive evaluation of the strength and significance of each association. These metrics help determine which rules are most useful for making predictions or decisions, such as suggesting additional items to customers based on their current selections.

3. 4. Recommendation Results

The next step involves separating films that have been watched by the target user from those that have not. If a film has already been watched, it will not be included in the recommended films for the user. The method to filter watched and unwatched films involves displaying the Top-N films already watched by the user and the Top-N recommended results. Table 4 displays the Top-N favorite films rated by userid 183.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Table 4. Top-N Favorite Films of User 183

Movieid	Title	Genres	Rating
110	Braveheart (1995)	Action|Drama|War	5.0
377	Speed (1994)	Action|Romance|Thriller	5.0
457	Fugitive, Th (1993)	Thriller	5.0
589	Terminator 2: Judgment Day (1991)	Action|Sci-Fi	5.0
1200	Aliens (1986)	Action| Adventure| Horror| Sci-Fi	5.0
1287	Ben-hur (1959)	Action| Adventure| Drama	5.0
1370	Die Hard 2 (1990)	Action| Adventure| Thriller	5.0
1387	Jaws (1975)	Action| Horror	5.0

Table 5 shows the Top-N recommended film results for the target user. Table 5 provides an example of items that will be recommended to the user. The recommended items are films that the user has not yet watched but share similar characteristics with the user's previously watched preferences.

Table 5 Top-N Recommended Results for User 183

Movieid	Title	Genres
260	Star Wars: Episode IV – A New Hope (1977)	Action| Adventure|Sci-Fi
1198	Raiders of the Lost Ark (Indiana Jones and the…	Action| Adventure
1197	Princess Bride, The (1987)	Action| Adventure| Comedy| Fantasy| Romance
1097	E.T. the Extra-Terrestrial (1982)	Children| Drama| Sci-Fi
593	Silence of the Lambs, The (1991)	Crime|Horror|Thriller
1291	Indiana Jones and the Last Crusade (1989)	Action| Adventure
2571	Matrix, The (1999)	Action| Sci-Fi| Thriller
1240	Terminator, The (1984)	Action| Sci-Fi| Thriller

3. 5. Evaluation of Recommendation Results

Table 6 presents the performance of the combined recommendation system methods across 3 clusters using the same dataset. Below is a more detailed analysis for each method based on accuracy metrics: Precision, Recall, F-Measure, and Hit-Rate.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Table 6. Accuracy Calculation Results

No	Recommendation System	Precision	Recall	F-Measure	Hit-Rate
1	2 Cluster	0.8149	0.9806	0.8766	0.9508
2	6 Cluster	0.8292	0.9655	0.8816	0.9180
3	16 Cluster	0.8010	0.9954	0.8696	0.9016

Based on Table 6, the combination of clustering, collaborative filtering, and association rule mining methods across 2 clusters resulted in a precision of 81.49% and recall of 98.06%. Although precision is slightly lower than with 6 clusters, recall remains very high. The F-Measure of 0.8766 indicates a balanced relationship between precision and recall. A Hit-Rate of 95.08% indicates that most users receive relevant recommendations.

With 6 clusters, the precision value is relatively high compared to 2 clusters and 16 clusters, indicating that most of the relevant predictions are accurate. However, with 6 clusters, the recall, F-measure, and hit rate values are slightly lower. In contrast, with 16 clusters, the recall value is quite high, meaning that the recommendation system effectively identifies relevant items.

3. 6. Discussion

In this study, the determination of the optimal number of clusters was crucial in enhancing the quality and effectiveness of the recommendation system. The Davies-Bouldin Index (DBI) was employed to identify the optimal number of clusters, directly impacting the grouping of users based on their movie preferences. A sparse dataset with many movies and users but limited rating information posed a challenge to developing a precise recommendation system. Thus, choosing the right number of clusters was essential to ensure meaningful groupings that could enhance recommendation accuracy.

The collaborative filtering approach involved creating a user-movie rating matrix (Figure 3) and a user similarity matrix (Figure 4). The user-movie rating matrix provided the foundational dataset, capturing how different users rated various movies. This matrix was instrumental in identifying user preferences and allowed the recommendation system to analyze patterns in how users rated similar movies. The user similarity matrix, derived from this rating data, quantified the degree of similarity between users using metrics like cosine similarity or Pearson correlation. By identifying users with high similarity scores, the system effectively

grouped users with similar movie tastes, enabling the recommendation of movies based on the preferences of other users with comparable interests.

The integration of clustering techniques using the k-means algorithm further enhanced the recommendation system by grouping users with similar preferences into clusters. For instance, with 2 clusters, the system achieved a balance between precision (81.49%) and recall (98.06%), demonstrating effective grouping for generating relevant recommendations. While the recall remained high across different cluster numbers, precision varied, suggesting that finer-grained clustering (e.g., 6 clusters) could improve the relevance of recommendations by accurately predicting more specific user preferences.

Additionally, association rule mining was applied to uncover frequent itemsets and generate association rules (Figure 5). The frequent itemsets highlighted combinations of items that frequently occurred together in the dataset, such as the itemset {1200, 1387, 589}, which appeared in 50% of transactions. This frequent occurrence suggests that these items are often purchased together, providing valuable insights for targeted marketing and cross-selling strategies. The generated rules, such as {480, 1387} → {2028}, were evaluated using metrics like support, confidence, and lift to determine their strength and usefulness. High confidence and lift values indicated strong associations, suggesting that the presence of certain items could predict the occurrence of others, thus enhancing the relevance of recommendations.

The recommendation results were further refined by filtering out movies that the target user had already watched, as demonstrated in Tables 4 and 5. This filtering step ensured that only new, relevant movies were suggested, aligning with the user's preferences for similar genres or characteristics. For example, User 183 had a preference for action and adventure films, and the top recommended films, such as "Star Wars: Episode IV – A New Hope (1977)" and "Raiders of the Lost Ark," matched these preferences.

The final evaluation (Table 6) showed that the combination of clustering, collaborative filtering, and association rule mining methods was effective across different numbers of clusters. The 2-cluster model provided a balanced relationship between precision and recall, while the 6-cluster model offered a higher precision, indicating that the recommendation system was more accurate in predicting relevant items. The 16-cluster model achieved the highest recall, suggesting its effectiveness in identifying a broader range of relevant items, though at the cost of slightly reduced precision.

This study demonstrates that the combination of clustering, collaborative filtering, and association rule mining can significantly enhance the efficiency and

relevance of a recommendation system. By leveraging user similarity and frequent itemsets, the system provides tailored recommendations that align closely with user preferences, improving both user satisfaction and engagement. The findings suggest that optimizing the number of clusters and integrating different recommendation techniques can yield a highly effective recommendation system capable of adapting to varying user behaviors and data sparsity challenges.

4. CONCLUSION

This study concludes that combining clustering, collaborative filtering, and association rule mining significantly improves the effectiveness and relevance of a movie recommendation system. The use of the Davies-Bouldin Index (DBI) to determine the optimal number of clusters enhanced the system's ability to group users based on similar preferences, with the 2-cluster model providing a balanced performance of high recall (98.06%) and precision (81.49%). Collaborative filtering, utilizing user-movie rating and similarity matrices, effectively identified users with similar tastes for personalized recommendations, while the k-means algorithm refined these recommendations by clustering users with similar preferences. Additionally, association rule mining revealed frequent itemsets and generated strong rules that predicted user behavior and suggested relevant items. By filtering out already watched movies, the system ensured its recommendations remained relevant and aligned with user interests. Overall, this integrated approach addresses data sparsity challenges and provides a robust, adaptive solution for delivering personalized recommendations, enhancing both user satisfaction and engagement.

REFERENCES

[1] 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.

[2] D. A. Mukhsinin, M. Rafliansyah, and S. A. Ibrahim, "Implementation of Decision Tree Algorithm for Movie Recommendation and Rating Classification on the Netflix Platform Implementasi Algoritma Decision Tree untuk Rekomendasi Film dan Klasifikasi Rating pada Platform Netflix," vol. 4, no. April, pp. 570–579, 2024.

[3] S. Halder, M. Samiullah, A. M. J. Sarkar, and Y. K. Lee, "Movie swarm: Information mining technique for movie recommendation system," *2012 7th Int. Conf. Electr. Comput. Eng. ICECE 2012*, no. February, pp. 462–465, 2012, doi: 10.1109/ICECE.2012.6471587.

[4] P. S. Sundari and M. Subaji, "An improved hidden behavioral pattern mining approach to enhance the performance of recommendation system in a big data environment," no. xxxx, 2020.
[5] Y. Lv, Y. Zheng, F. Wei, C. Wang, and C. Wang, "AICF: Attention-based item collaborative filtering," *Adv. Eng. Informatics*, vol. 44, no. February, p. 101090, 2020, doi: 10.1016/j.aei.2020.101090.
[6] K. Patel and H. B. Patel, "A state-of-the-art survey on recommendation system and prospective extensions," *Comput. Electron. Agric.*, vol. 178, no. September, p. 105779, 2020, doi: 10.1016/j.compag.2020.105779.
[7] J. Xu, X. Zheng, and W. Ding, "Personalized recommendation based on reviews and ratings alleviating the sparsity problem of collaborative filtering," *Proc. - 9th IEEE Int. Conf. E-bus. Eng. ICEBE 2012*, pp. 9–16, 2012, doi: 10.1109/ICEBE.2012.12.
[8] M. K. Najafabadi, M. N. ri Mahrin, S. Chuprat, and H. M. Sarkan, "Improving the accuracy of collaborative filtering recommendations using clustering and association rules mining on implicit data," *Comput. Human Behav.*, vol. 67, pp. 113–128, 2017, doi: 10.1016/j.chb.2016.11.010.
[9] C. Zhang, W. Huang, T. Niu, Z. Liu, G. Li, and D. Cao, "Review of Clustering Technology and Its Application in Coordinating Vehicle Subsystems," *Automot. Innov.*, vol. 6, no. 1, pp. 89–115, 2023, doi: 10.1007/s42154-022-00205-0.
[10] R. Obeidat, R. Duwairi, and A. Al-Aiad, "A Collaborative Recommendation System for Online Courses Recommendations," *Proc. - 2019 Int. Conf. Deep Learn. Mach. Learn. Emerg. Appl. Deep. 2019*, pp. 49–54, 2019, doi: 10.1109/Deep-ML.2019.00018.
[11] U. Liji, Y. Chai, and J. Chen, "Improved personalized recommendation based on user attributes clustering and score matrix filling," *Comput. Stand. Interfaces*, vol. 57, no. November 2017, pp. 59–67, 2018, doi: 10.1016/j.csi.2017.11.005.
[12] C. F. Tsai and C. Hung, "Cluster ensembles in collaborative filtering recommendation," *Appl. Soft Comput. J.*, vol. 12, no. 4, pp. 1417–1425, 2012, doi: 10.1016/j.asoc.2011.11.016.
[13] Alith Fajar Muhammad, "Klasterisasi Proses Seleksi Pemain Menggunakan Algoritma K-Means (Study Kasus : Tim Hockey Kabupaten Kendal)," *Jur. Tek. Inform. FIK UDINUS*, vol. 1, no. 1, pp. 1–5, 2015.
[14] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261–273, 2015, doi: 10.1016/j.eij.2015.06.005.
[15] E. T. L. Kusrini, "Algoritma data mining," pp. 63–77, 2009.
[16] H. H. Arfisko, F. Informatika, U. Telkom, A. T. Wibowo, F. Informatika, and U. Telkom, "Sistem Rekomendasi Film Menggunakan Metode Hybrid Collaborative Filtering Dan Content-Based Filtering," vol. 9, no.

3, pp. 2149–2159, 2022.

[17] R. Trihatmaja and Y. D. Wardhana Asnar, "Improving the Performance of Collaborative Filtering Using Outlier Labeling, Clustering, and Association Rule Mining," *Proc. 2018 5th Int. Conf. Data Softw. Eng. ICoDSE 2018*, pp. 1–6, 2018, doi: 10.1109/ICODSE.2018.8705883.