Recommendation System using Modified UPCSim Algorithm

Akshara P S (106118005), Painthamizh Paavai A S (106118069),
Smrithi Prakash (106118089)

1 ABSTRACT

Collaborative filtering is a widely used recommendation system approach. It overcomes the limitations of content-based filtering by using similarities between users and items simultaneously. The main focus of research involving collaborative filtering techniques is modifying a similarity algorithm to increase the accuracy of the recommendation system. One such similarity algorithm, UPCSim, was proposed, which combines user rating and user behavior values. Here, the user behavior value is calculated using user score probability in assessing genre data. Capturing user behavior using genre data alone is inefficient.

Our study proposes a new similarity algorithm - User Profile Correlation-based Similarity (UPCSim) using the Jaccard Similarity. The final similarity matrix is obtained by combining the similarity based on behavior and rating data. This user profile data is used to calculate the weights of similarities of user rating and user behavior values.

2 INTRODUCTION

A recommender system, a recommendation system, or a recommendation engine is a subclass of information filtering systems that seeks to predict the "preference" or "rating" a user would give to an item. Recommendation systems are widely used these days and play a significant role in advertising. It helps users of multiple online platforms to look for the right products based on their needs, interests, and likings. Recommendation systems have played a vital role in helping build large online shopping websites with a broader range of products than a traditional store.

Collaborative filtering is a technique used by recommendation engines. There are two approaches to collaborative filtering - user-user based and item-item based. In user-user-based collaborative filtering, the ratings are given based on the ratings assigned by similar users. Here, the similarity is calculated between the users. In item-item-based collaborative filtering, the rating is given based on the ratings given to similar products. Here, the similarity is calculated between different products.

Our research work proposes an algorithm that incorporates all user data - rating, behavior, and profile. Our paper focuses on predicting the rating given by a user for a movie based on the similarity obtained from user behavior values, user rating values, and user profile values. We try multiple ways of calculating similarity, including cosine similarity, Pearson correlation coefficient, and Jaccard's similarity, and find Jaccard similarity to give better results.

3 LITERATURE SURVEY

Memory based collaborative filtering (MBCF) is a prevalent collaborative filtering method with several commercial applications. This approach utilizes a similarity algorithm like Cosine, Pearson or Jaccard to take the weighted average of ratings. The advantage of MBCF is that it is easy for the creation and explainability of results. MBCF approaches can be divided into two main sections : item- item filtering and user-item filtering.

Wu et al. [2] uses a combination of two similarities in his study. But has several limitations. The calculation of similarity based on only the user behavior value considers the item's genre data without considering the user profile. Based on this issue, some researchers used user profile data to indicate similar users' preferences in recommender systems.

Al-Shamri [11] had used three attributes of the user profile data - age, gender, and occupation excluding the genre values to calculate the similarity between users in the demographic recommendation system.This experiment was conducted using MovieLens100K dataset. Similarity and prediction computation involves each and every attribute in the dataset. The final prediction is an aggregate of a simple average of all the predictions based on each attribute. MAE and RMSE values of 0.91 and 1.22 were obtained from this research.

Yassine et al. [12] had used two user profile data attributes: gender and age. The dataset used for the experiment was MovieLens100K. Collaborative filtering was combined with K-means clustering on each user profile attribute. This combined approach is used to clused movies and the user profile data was used for finding user segmentation. Model based collaborative filtering with SVD was used for making recommendations in the resulting segmented users. The result of the experiment showed that the F-measure of the k-means collaborative filtering on gender attribute and the k-means collaborative filtering on age attribute are 2.23 and 1.04, respectively.

4 CONTRIBUTION

We experimented with multiple similarity metrics of calculating the similarity between users based on ratings and behavior. The UPCSim algorithm proposed by Widiyaningtyas, T., Hidayah, I. & Adji, T.B.[1] uses Cosine similarity for calculating the similarity between users based on ratings and Pearson's correlation coefficient to calculate the similarity between users based on behavior. We experimented with both Cosine Similarity and Pearson's Correlation Coefficient and with Jaccard similarity metrics to calculate the similarity between users based on both ratings and behavior. We found that the results obtained using Jaccard similarity were better than those obtained using Cosine similarity and Pearson's correlation coefficient.

5 PROPOSED ALGORITHM

OVERVIEW

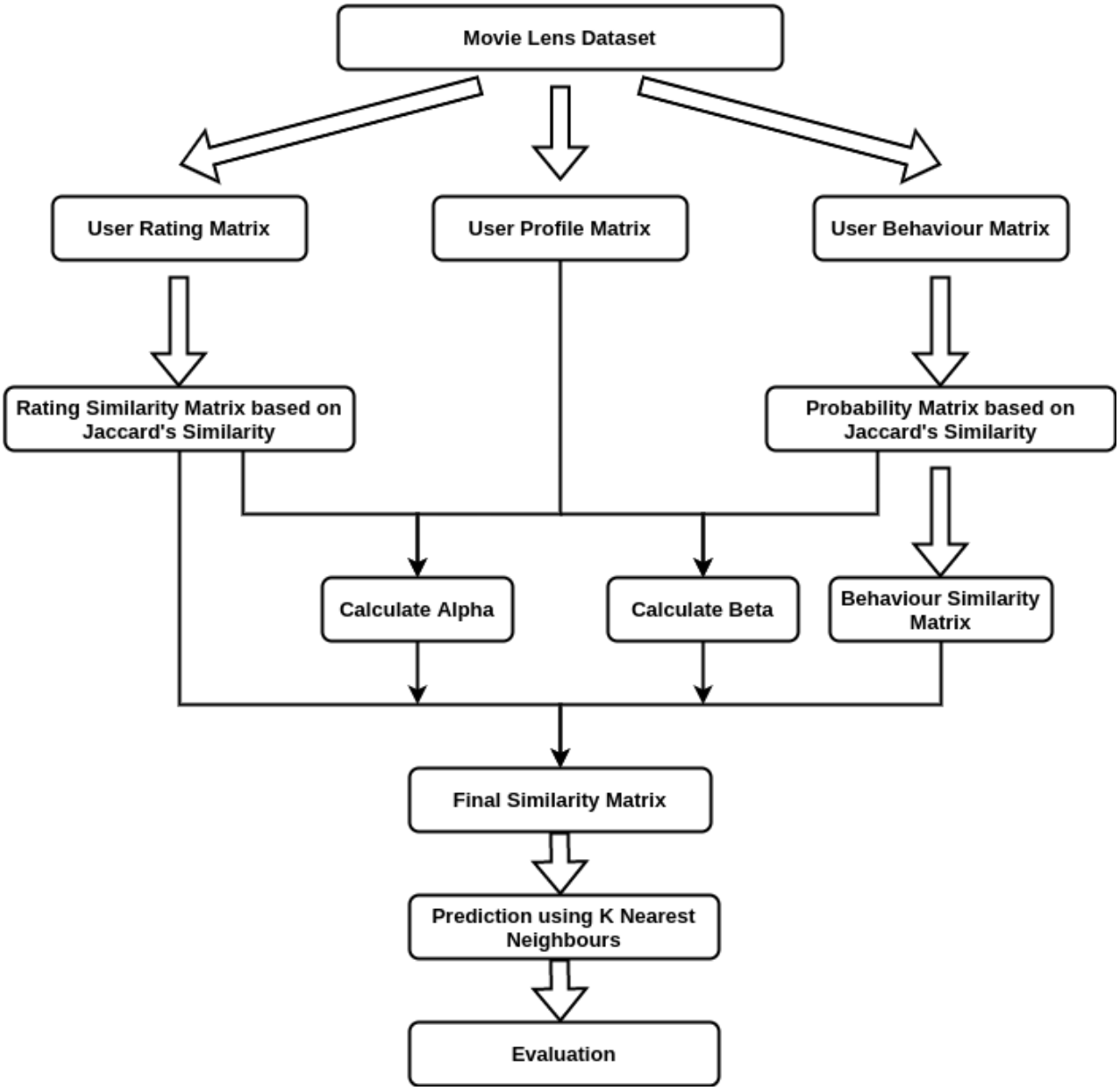
A modified version of the UPCSim algorithm is proposed in this study. In this algorithm, a similarity matrix is used for predicting the rating given by a user for a particular movie using the k nearest neighbor algorithm. The similarity matrix is obtained by a weighted sum of two similarity matrices - the user rating similarity matrix and the user behavior similarity matrix. Correlation coefficients between user profile values and user behavior values and between user profile values and user behavior values are used to compute the similarity weighting. The rating prediction is made using the k nearest neighbor algorithm based on the final similarity matrix calculated.

DATASET

The dataset used in our study is the Movie Lens dataset collected by the "GroupLens Study Group of the University of Minnesota". It consists of several versions including ml-100K, ml-1M, ml-10M, ml-20M, etc. We chose the ml-100K version for our experiment. MovieLens 100K dataset is a stable benchmark dataset and contains 100,000 movie ratings given by 943 users for 1682 movies across 19 genres. The 19 genres include action, adventure, animation, children, comedy, crime, documentary, drama, fantasy, film-noir, horror, musical, mystery, romance, sci-fi, thriller, war, western and unknown. The dataset also includes the profile information of all 943 users. The profile information includes age, gender, occupation, and zip code.

ALGORITHM

The diagram below gives an overview of the various steps involved in the algorithm.



I. CALCULATING SIMILARITY

To formulate the similarities, it can be assumed that the user and item sets are defined as $U=\{u_1,u_2,\ldots,u_m\}$ and $I=\{i_1,i_2,\ldots,i_n\}$. $R=[r_{ui}]_{mxn}$ denotes the user rating value matrix. Here m and n are the number of users and the number of items, respectively, and r_{ui} is the rating given by user u on item i.

A. COSINE SIMILARITY

Cosine similarity is an approach to measure the similarity between two vectors of an inner product space. The cosine of the angle between two vectors is calculated and it determines whether two vectors are pointing in roughly the same direction.

$$Sim(u_1, u_2)^{COS} = \frac{\vec{r}_{u_1} . \vec{r}_{u_2}}{\|\vec{r}_{u_1}\| . \|\vec{r}_{u_2}\|} = \frac{\sum_{i \in I_{u_1} \cap I_{u_2}} r_{u_1 i} . r_{u_2 i}}{\sqrt{\sum_{i \in I_{u_1} \cap I_{u_2}} r_{u_1 i}^2} . \sqrt{\sum_{i \in I_{u_1} \cap I_{u_2}} r_{u_2 i}^2}}$$

B. PEARSON'S CORRELATION COEFFICIENT

Pearson's correlation coefficient gives a measure of the statistical relationship, or association, between two continuous variables. Information about correlation, or the magnitude of the association, can be obtained using Pearson's correlation coefficient.

$$Sim(u_1, u_2)^{PCC} = \frac{\sum_{i \in I_{u_1} \cap I_{u_2}} (r_{u_1 i} - \bar{r}_{u_1}) . (r_{u_2 i} - \bar{r}_{u_2})}{\sqrt{\sum_{i \in I_{u_1} \cap I_{u_2}} (r_{u_1 i} - \bar{r}_{u_1})^2} . \sqrt{\sum_{i \in I_{u_1} \cap I_{u_2}} (r_{u_2 i} - \bar{r}_{u_2})^2}}$$

C. JACCARD SIMILARITY

The Jaccard similarity coefficient or the Jaccard index is a statistic used for gauging the similarity and diversity of sample sets. It can be used for finding similarities between vectors.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

II. CONSTRUCTING SIMILARITY MATRIX

A. RATING SIMILARITY MATRIX (S_R)

Here, the similarity calculation is done based on the user rating value. The user rating matrix is constructed. The user rating matrix is a 943 x 1682 matrix. Here, 943 represents the number of users, and 1682 represents the number of movies in the dataset. Each row corresponds to the ratings given by a user to the 1682 movies.

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} & R_{14} & \cdots & R_{1\_1682} \\ R_{21} & R_{22} & R_{23} & R_{24} & \cdots & R_{2\_1682} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ R_{943\_1} & R_{943\_2} & R_{943\_3} & R_{943\_4} & \cdots & R_{943\_1682} \end{bmatrix}$$

R_{943_1682} is the user rating value given by the 943rd user for the 1682nd item. The values of R_{11} to R_{943_1682} range from 0 to 5. Here a value of 0 shows that the user has not given a rating for the movie. After forming the user rating value matrix, the next step is to calculate the S_r similarity matrix using Jaccard's similarity formula. The final result of the S_r similarity calculation forms the S_r similarity matrix of order 943 × 943, as shown below.

$$S_r = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \cdots & S_{1\_943} \\ S_{21} & S_{22} & S_{23} & \cdots & S_{2\_943} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{943\_1} & S_{943\_2} & S_{943\_3} & \cdots & S_{943\_943} \end{bmatrix}$$

S_{1_943} is the similarity value between the 1st user and the 943rd user based on the movie ratings given by the two users.

B. BEHAVIOUR SIMILARITY MATRIX (S_B)

The S_b similarity matrix indicates the similarity based on the user behavior values. Every movie in the MovieLens 100K dataset can be represented as a vector of size 19 based on the genre information. Each row in the user behavior matrix is obtained by taking the aggregate of the movie vectors for which the user gave a rating.

The user behavior matrix of order 943 × 19 is as shown below. 943 indicates the number of users, and the number 19 represents the number of genres.

$$B = \begin{bmatrix} B_{11} & B_{12} & B_{13} & \cdots & B_{1\_19} \\ B_{21} & B_{22} & B_{23} & \cdots & B_{2\_19} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ B_{943\_1} & B_{943\_2} & B_{943\_3} & \cdots & B_{943\_19} \end{bmatrix}$$

B_{943_19} is the 943rd user behavior value for the 19th genre, representing the total number of movies of the 19th genre watched by the 943rd user. Once the user behavior value matrix is formed, the next step is to compute the probability of genre occurrence from the user behavior value matrix to create a probability matrix of user behavior value using the following formula.

$$P = B(g) / N$$

B(g) represents the user behavior value for the target genre g, and N represents the number of users who had given a rating to the target genre g. An illustration of the probability matrix of user behavior value is shown below.

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} & \cdots & P_{1\_19} \\ P_{21} & P_{22} & P_{23} & \cdots & P_{2\_19} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ P_{943\_1} & P_{943\_2} & P_{943\_3} & \cdots & P_{943\_19} \end{bmatrix}$$

Here P_{943_19} is the probability value of the 943rd user behavior for the 19th genre. The probability matrix of user behavior value is utilized for computing the S_b similarity. The results of the S_b similarity calculation form a matrix of order 943 × 943, shown as follows.

$$S_b = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \cdots & S_{1\_943} \\ S_{21} & S_{22} & S_{23} & \cdots & S_{2\_943} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{943\_1} & S_{943\_2} & S_{943\_3} & \cdots & S_{943\_943} \end{bmatrix}$$

C. UPCSIM ALGORITHM

The UPCSim algorithm is a component of the similarity calculation which is done using the UPCSim algorithm, which calculates the weights of the similarities S_r and S_b, based on the user profile attributes such as gender, age, location, and occupation as given in the MovieLens 100K dataset. The weights of these two similarities are computed based on the correlation coefficient (R) using multiple linear regression.

$$Y = a + b_1X_1 + b_2X_2 + b_3X_3 + \cdots + b_nX_n$$

$$R = \sqrt{\frac{b_1 \sum X_1Y + b_2 \sum X_2Y + b_3 \sum X_3Y + \cdots + b_n \sum X_nY}{\sum Y^2}}$$

Here X represents the independent variables and Y represents the dependent variable. The variable b is the regression coefficient for each independent variable and a is a constant. In our study the independent variable X denotes the user profile data with four independent variables, namely age, gender, occupation and location. The dependent variable Y represents user rating value or user behaviour value.

The value of the correlation coefficient between user profile and user rating values is used to calculate the weight of S_r similarity. This is denoted by α.

The value of the correlation coefficient between user profile and user behaviour value is used to calculate the weight of S_b similarity. This is denoted by β.
After calculating weights, the next step is the calculation of the final similarity matrix by combining the weighted S_r and S_b similarities. This final similarity matrix is of the order 943 x 943, defined as follows.

$$S(u,v) = \alpha S_r(u,v) + \beta S_b(u,v)$$

i. S(u,v) represents the final similarity between user u and user v.
ii. S_r(u,v) represents the similarity based on user rating value between user u and user v.
iii. S_b(u,v) represents the similarity based on user behaviour value between user u and user v.
iv. α represents the weight of S_r
v. β represents the weight of S_b

III. PREDICTION

A. K NEAREST NEIGHBOURS

The K Nearest Neighbours algorithm is used to predict the rating given by a user for a particular movie. An important part of this algorithm is finding the best value of k for predicting the most accurate results. In this study, we experimented with values 20, 40, 60, 80 and 100 to make predictions.

When predicting the rating value given by a user u for a movie m, we consider the k most similar users to user u based on the values in the similarity matrix and use the below formula to predict the rating.

$$p_{ui} = \bar{r}_u + \frac{\sum_{v \in NNu} S(u,v).(r_{vi} - \bar{r}_v)}{\sum_{v \in NNu} |S(u,v)|}, v \neq u$$

While considering k most similar users, we make sure that all the k users have given a rating to the movie m.

6 PERFORMANCE EVALUATION

For performance evaluation the dataset was split into two parts as training data and testing data. The k-fold cross-validation method was applied with k=5 separating 80% of the dataset for training and the remaining 20% for testing. The results for each round of training and testing were recorded.

For evaluating the performance of the recommendation system, hence developed, two metrics were used : Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The formulas for MAE and RMSE are as given below.

$$MAE = \frac{1}{TN} \sum_{u \in U, i \in I} |p_{ui} - r_{ui}|$$

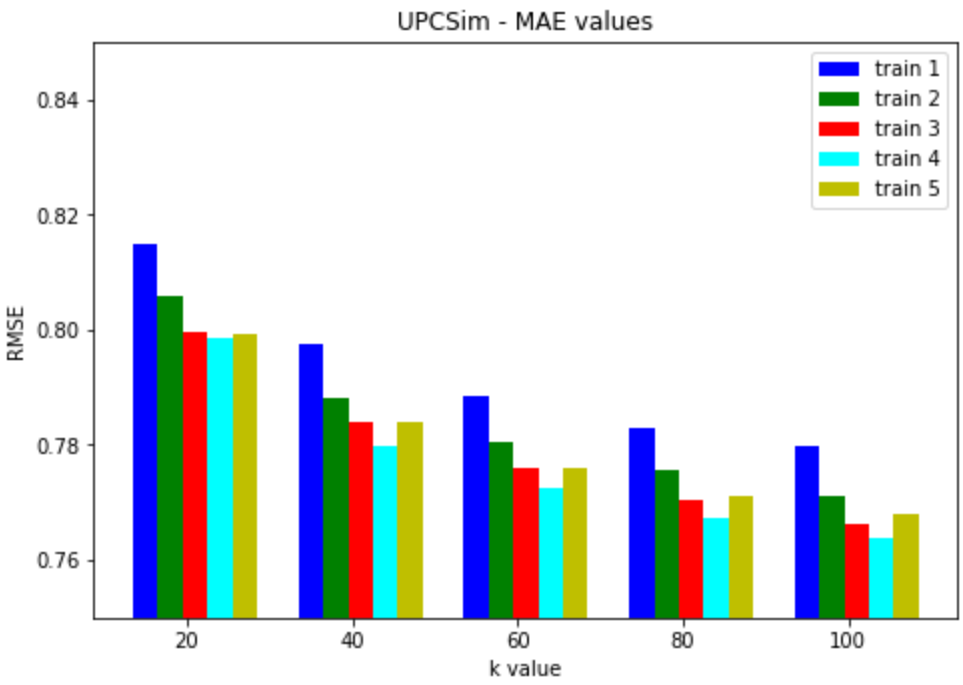
$$RMSE = \sqrt{\frac{1}{TN} \sum_{u \in U, i \in I} (p_{ui} - r_{ui})^2}$$

The algorithm developed was first experimented using the Cosine similarity and Pearson's Correlation Coefficient as per the original UPCSim model followed by Jaccard's Similarity.
The results obtained are as follows.

COSINE-PEARSON RESULTS

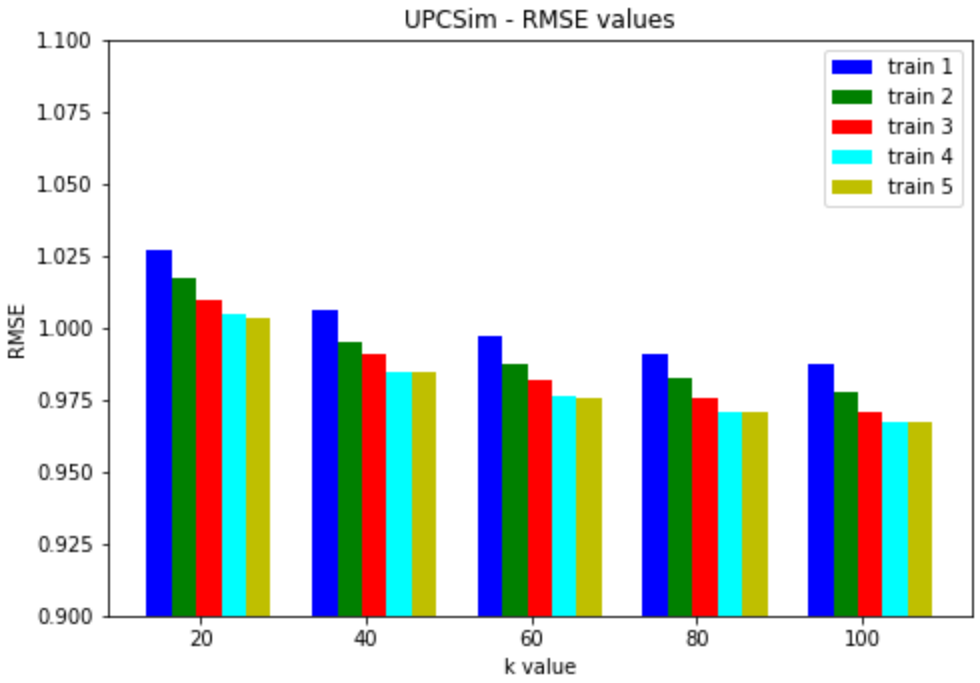
MAE Values

	k=20	**k=40**	**k=60**	**k=80**	**k=100**
train 1	0.8150	0.7975	0.7886	0.7829	0.7797
train 2	0.8059	0.7882	0.7806	0.7755	0.7712
train 3	0.7996	0.7839	0.7759	0.7703	0.7661
train 4	0.7987	0.7798	0.7724	0.7674	0.7639
train 5	0.7991	0.7838	0.7758	0.7712	0.7680
average	0.8037	0.7866	0.7787	0.7735	0.7698



RMSE Values

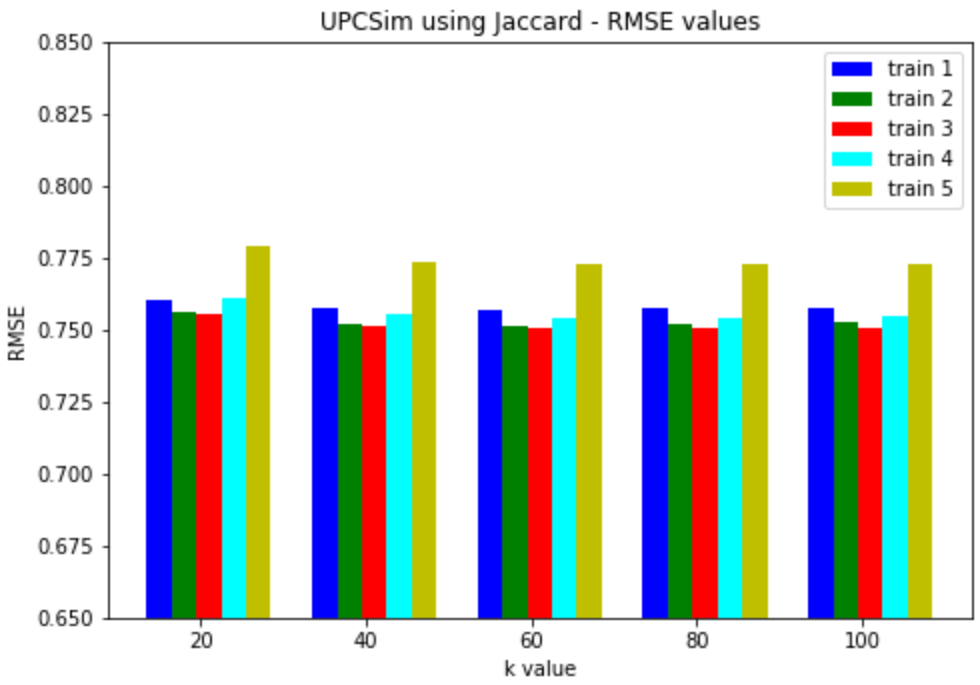
	k=20	**k=40**	**k=60**	**k=80**	**k=100**
train 1	1.0269	1.0063	0.9968	0.9907	0.9872
train 2	1.0172	0.9953	0.9877	0.9822	0.9779
train 3	1.0098	0.9911	0.9818	0.9756	0.9710
train 4	1.0049	0.9849	0.9765	0.9708	0.9672
train 5	1.0033	0.9847	0.9758	0.9707	0.9675
average	1.0124	0.9925	0.9837	0.9780	0.9742



JACCARD SIMILARITY RESULTS

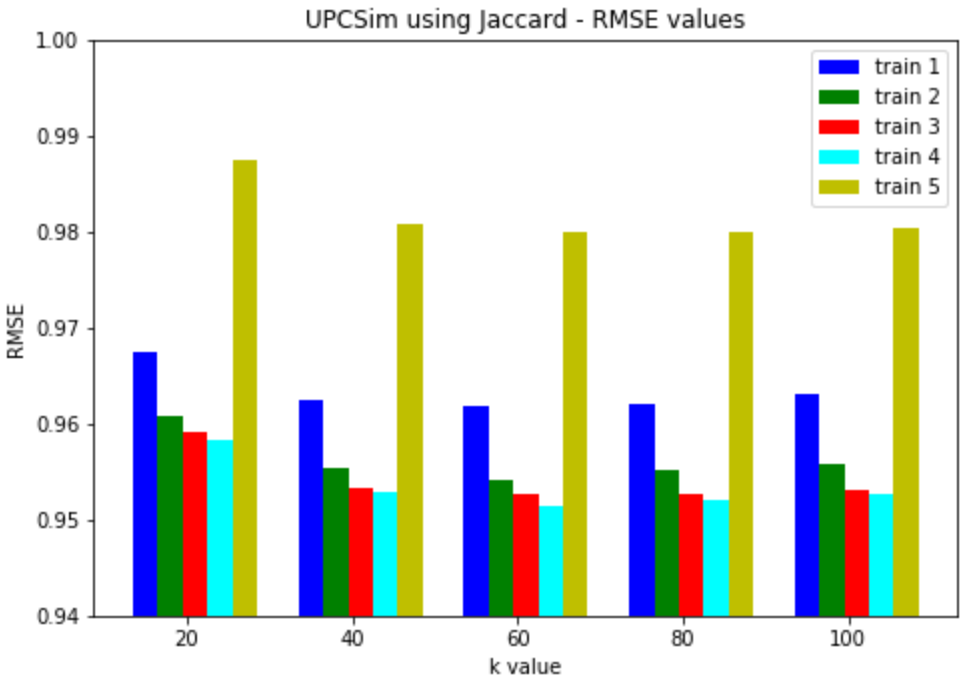
MAE VALUES

	k=20	**k=40**	**k=60**	**k=80**	**k=100**
train 1	0.7606	0.7578	0.7571	0.7573	0.7578
train 2	0.7563	0.7519	0.7514	0.7517	0.7523
train 3	0.7557	0.7515	0.7509	0.7507	0.7509
train 4	0.7608	0.7556	0.7542	0.7543	0.7547
train 5	0.7790	0.7738	0.7728	0.7729	0.7731
average	0.7625	0.7581	0.7573	0.7574	0.7578



RMSE VALUES

	k=20	**k=40**	**k=60**	**k=80**	**k=100**
train 1	0.9675	0.9624	0.9618	0.9621	0.9630
train 2	0.9607	0.9554	0.9542	0.9551	0.9558
train 3	0.9592	0.9533	0.9527	0.9526	0.9531
train 4	0.9583	0.9529	0.9515	0.9520	0.9526
train 5	0.9875	0.9808	0.9799	0.9799	0.9804
average	0.9666	0.9610	0.9600	0.9603	0.9610



COMPARISON WITH EXISTING SOLUTION

For the UPCSim algorithm based on the Cosine similarity and Pearson's Correlation Coefficient, we find that the best average MAE value and RMSE value is obtained for k = 100. The MAE value obtained is 0.7698 and the RMSE value obtained is 0.9742. For the UPCSim algorithm based on Jaccard similarity, the best average MAE value and RMSE value is obtained for k = 60. The MAE value is 0.7573 and the RMSE value is 0.9600. We find that by using Jaccard similarity, the error in prediction reduces and the value of k also becomes lower reducing the computation cost.

COMPARISON TABLE

Comparison table for the average MAE and RMSE values of UPCSim and Modified UPCSim algorithm.

	UPCSim MAE	**UPCSim RMSE**	**Modified UPCSim MAE**	**Modified UPCSim RMSE**
Train 1	0.8037	1.0124	0.7625	0.9666
Train 2	0.7866	0.9925	0.7581	0.9610
Train 3	0.7787	0.9837	0.7573	0.9600
Train 4	0.7735	0.9780	0.7574	0.9603
Train 5	0.7698	0.9742	0.7578	0.9610

CONCLUSION

The main objective of this paper was to study the UPCSim algorithm and experiment with multiple ways of calculating similarities and observing the results obtained. We experimented with Cosine similarity and Pearson Correlation Coefficient and with Jaccard's similarity and found Jaccard's similarity to perform better.

REFERENCES

[1] Widiyaningtyas, T., Hidayah, I. & Adji, T.B. User profile correlation-based similarity (UPCSim) algorithm in movie recommendation system. *J Big Data* 8, 52 (2021). https://doi.org/10.1186/s40537-021-00425-x

[2] Xu G, Tang Z, Ma C, Liu Y, Daneshmand M. A collaborative filtering recommendation algorithm based on user confidence and time context. J Electr Comput Eng. 2019;2019:1–12. https://doi.org/10.1155/2019/7070487.

[3] Feng J, Fengs X, Zhang N, Peng J. An improved collaborative filtering method based on similarity. PLoS ONE. 2018;13(9):1–18. https://doi.org/10.1371/journal.pone.0204003.

[4] Liu H, Hu Z, Mian A, Tian H, Zhu X. A new user similarity model to improve the accuracy of collaborative filtering. Knowl Based Syst. 2014;56:156–66. https://doi.org/10.1016/j.knosys.2013.11.006.

[5] Wu, Y., ZHao, Y. & Wei, S. Collaborative filtering recommendation algorithm based on interval-valued fuzzy numbers. Appl Intell 50, 2663–2675 (2020). https://doi.org/10.1007/s10489-020-01661-z

[6] Wu Y, Zhang X, Yu H, Wei S, Guo W (2017) Collaborative filtering recommendation algorithm based on user fuzzy similarity. Intell Data Anal 2:311–327

[7] C. Martinez-Cruz, C. Porcel, J. Bernabé-Moreno, E. Herrera-Viedma,
A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling, https://doi.org/10.1016/j.ins.2015.03.013.

[8] Keunho Choi, Yongmoo Suh,
A new similarity function for selecting neighbors for each target item in collaborative filtering, https://doi.org/10.1016/j.knosys.2012.07.019.

[9] P. Jaccard, “Nouvelles recherches sur la distribution florale,”Bull. Soc. Vandoise des Sci. Nat.,44, 223–270 (1908).

[10] Adomavicius, G., Tuzhilin, A.: 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), 734–749 (2005)

[11] Al-Shamri MYH. User profiling approaches for demographic recommender systems. Knowl Based Syst. 2016;100:175–87. https://doi.org/10.1145/2827872.

[12]Yassine A, Mohamed L, Al Achhab M. Intelligent recommender system based on unsupervised machine learning and demographic attributes. Simul Model Pract Theory. 2020;107:1–9. https://doi.org/10.1016/j.simpat.2020.102198.