JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 37, 281-293 (2021)
DOI: 10.6688/JISE.202103_37(2).0001

Building an Adaptive Recommendation Model Based on Fuzzy MP Neuron and Weighted Similarity Indicator

BAM BAHADUR SINHA AND DR. R. DHANALAKSHMI
Department of Computer Science and Engineering
National Institute of Technology Nagaland
Dimpaur, Nagaland, 797103 India
E-mail: {bahadurbam43; dhanalakshmiparthiban}@gmail.com

Recommender system is one of the most common data filtering techniques used. It helps to discover hidden patterns of information from a wide range of omnipresent products and services. When dataset drifts from scarcity to abundance, the most common methods such as collaborative filtering suffer from information sparsity complication, over-specification, and elevated computational complexity. We have created a hybrid model in this respect that considers between precision and computation time to produce the most appropriate products for customers in real time. We made use of imputation technique, fuzzy logic using novel similarity technique and McCulloch-Pitts (MP) Neuron to cope up with aforementioned complications. The experimental evaluation on MovieLens dataset and comparison with numerous state of art personalization models shows that the proposed model yields high efficiency and effectiveness. We tested the resultant classification accuracy of our proposed model using precision and recall.

Keywords: recommendation, fuzzy *c*-means, MP neuron, similarity, precision, recall

1. INTRODUCTION

Recommender program extracts knowledge to identify the most relevant products from a large pool of products. It is widely used in the area of e-commerce, without any thrilling or specific queries [1]. Collaborative filtering is one of the main recommending mechanisms that operates with the idea that users who have similar interests in the past should taste similar in the future [2]. The rapid growth of items on the market has created a huge problem of efficient item management and the advice given to the right customer on the basis of his/her interest at the right moment. Many methods for recommending items to consumers with high levels of accuracy have been suggested. Nevertheless, the over-specification problem affects most versions of existing recommendation models. The problem with over-specification is that the recommendation model recommends redundant products, which means that it cannot provide any users in the system with relevant and consistent product or details [3]. Hence the existing methodologies must be revised to deal with sparse data, irrespective of all changes, and the final recommendation list must be divided into interesting and uninteresting category of products [4].

Received August 10, 2019; revised January 13 & February 23, 2020; accepted March 30, 2020.
Communicated by Jimson Mathew.

The proposed work in this paper endorses following contributions: (i) Complete the missing data set values using the most common user-specified rating and most common item rating. This technique of imputation helps to tackle sparsity problem; (ii) Calculation of similarity between users using a new technique for similarity. The user identity is measured by the review of the commonly rated products. The optimal values of parameters for the designed similarity metric is decided using Genetic Algorithm; (iii) Placement of all users in the system based on their ranking actions in specific clusters. An application of the elbow method and the approach to the dendrogram on the available utility (user-item rate) matrix provides the optimum number of cluster decision; (iv) Fuzzy c-means implementation to decide which user will be located in which cluster on the basis of Fuzzy's membership value; (v) Use of the MP Neuron model to identify the final list of recommendations as important and uninteresting. By analyzing precision and recall, the accuracy of the model has been analyzed. After successful model implementation, it can be extended to real-life applications such as web-service guidelines with much less user information available.

2. METHODOLOGIES

In the area of personalization, a lot of work has been done. But in the case of sparse data, most models struggle and suffer from the issue of strong computational sophistication. We used the following approaches to develop a new hybrid suggestion model in order to resolve all the above problems. The efficiency of our model is ensured with the precision and recall analysis of the classification.

2.1 Frequent Rating Imputation

In order to decide the ratings for unrated items, we used the most common rating method of matching imputation. This process focuses on the user rating most frequently given and the rating received the most frequently by the item for which the rating is calculated. The two values are assigned an average to the unrated item. Therefore, the most popular matching imputation contributes to a decrease in sparsity through Eq. (1).

$$r(I_{u,i}) = \frac{r_f(I_u) + r_f(I_i)}{2} \tag{1}$$

where $r(I_{(u,i)})$ represents the rating received by item 'i' given by user 'u'. $r_f(I_u)$ reflects the frequent rating given by user 'u' to the set of items present in the system and $r_f(I_i)$ denotes the rating received by item 'i' by the set of users present in the system.

2.2 Indicator of User-Similarity

The new similarity function was considered, which calculates the similarity between users based on the commonly rated items and the vector component 'w' which represents the weighing of frequently rated item "F" [5]. The value of weight is in the rage of [–1, 1] and weight vector has following five components associated with it: $w \in (w_0, w_1, w_2, w_3, w_{(X-x)})$, where '$X$' and '$x$' represents the maximum and minimum possible rating present in the system. The value of weight factor 'w' is classified as:

w_0 = [0.6, 1]: It considers the high positive values considered for similarity computation. For instance, any user 'u_1' and 'u_2' giving same rating to any item 'i', the rating difference between both users turns out to be 0 and thus gives a conclusion that they possess high similarity in rating behavior.

w_1 = [0.2, 0.6]: It reflects the intermediate behavior of the user in a positive direction. It considers the rating difference of value '1'.

w_2 = [–0.2, 0.2]: In order to ignore the neutral items, we took zero consideration of items that possess rating difference as 2.

w_3 = [–0.6, –0.2]: The rating difference of 3 reflects the intermediate behavior of users in negative direction.

w_4 = [–0.6, –1]: This vector component of similarity indicator helps in discarding the high negativity possessed by the user. In a system where rating scale ranges from +1 to +5, where +1 and +5 denotes minimum and maximum rating in the system respectively. The maximum negativity in terms of rating difference in such scenario is '4'.

We consider only the items that has been rated by both users for which similarity computation is to be done. Suppose I_U and I_V represents two set of items such that I_U includes rating values (4, 3, 2, 2, 1, *, 5) and I_V = (2, *, 5, 4, 2, *, 5) rated by two users U and V respectively. $F^{(U,V)} = (F_0^{(U,V)}, F_1^{(U,V)}, F_2^{(U,V)}, F_3^{(U,V)}, F_4^{(U,V)})$, the components of '$F$' vector comprise of the possible ratings that any user can give to any item present in the system. In the proposed similarity indicator, we consider only the ratings which has been rated by both the users among which we are computing similarity. In the above scenario, '$*$' represents the unrated item. $F_a^{(U,V)} = \frac{A}{B}$, where '$A$' denotes the absolute difference 'a' in terms of rating given by user U and user V. 'B' represents the total items commonly rated by both users U and V. $F_a^{(U,V)} = (\frac{1}{5}, \frac{2}{5}, \frac{1}{5}, \frac{1}{5}, \frac{0}{5})$. Resultant similarity indicator formed after combining the aforementioned components is represented mathematically via Eq. (2).

$$Sim_w^{(U,V)} = \frac{1}{X - x + 1} \sum_{a=0}^{X-x} w_a F_a^{(U,V)} \tag{2}$$

Where U and V are two users, X and x represents the maximum and minimum possible rating value in the system. 'w_a' denotes the weight vector associated with the user importance. The optimal values of weight vector are decided using genetic algorithm [6]. F_a analyzes the commonly rated items.

2.3 Optimal Cluster Selection

We performed thorough analysis of the elbow method and the dendrogram in order to determine the optimum number of clusters on the available utility matrix reflecting user ration behavior. This is done to divide the complete set of users in the system into different clusters.

2.3.1 Elbow method

It is a cluster analysis method which helps to detect the optimal number of clusters that can be established by using Fuzzy c-means in given utility matrix [7]. The mathematical representation for elbow detection in cluster analysis is given by Eq. (3).

$$W_z = \sum_{k=1}^{z}[\frac{\sum_{x=1}^{z_k-1}\sum_{y=1}^{z_k}||P_x - P_y||_2}{z_k}] \tag{3}$$

where 'z' denotes the total number of clusters to be formed, z_x represents the number of points that resides in any specific cluster 'z', P_k and P_l are points that lie in cluster 'k' and 'l' respectively. $|(|P_k - P_l|)|_2$ gives a mathematical representation for calculation of L_2 norm between the two points. This L_2 norm helps in deciding the similarity between the point 'k' and 'l'. The primary goal of elbow method is help in deciding the optimal cluster value such that C_z results in giving a minimum value.

2.3.2 Dendrogram

The structure of the tree is similar to that of the elbow test. We used this methodology to help our optimum judgment on the cluster with the elbow method. Eq. (4) shows the dendrogram's mathematics [8].

$$\frac{1}{|x_1|.|x_2|}\sum_{x_1\in X_1}\sum_{x_2\in X_2} d(x_1,x_2) \tag{4}$$

where X_1 and X_2 are any two clusters and $d(x_1,x_2)$ represents the distance between the two points x_1 and x_2 lying in cluster X_1 and X_2 respectively.

2.4 Fuzzy *c*-means

This allows to measure the affiliation weights for the consumers of different clusters. The versatility to encourage the same individual to lie in more than one cluster is one of the main advantages of Fuzzy C-means. In the event of a simultaneous datasets the output of Fuzzy c-means is greater than that of a K-means clustering [9]. This has guided us, instead of the most common K-means algorithm, to use Fuzzy c-means. The logical representation of Fuzzy c-means in Eq. (5) suggests the goal function to be decreased during the Fuzzy c-means implementation [10].

$$g(r) = \sum_{k=1}^{n}\sum_{l=1}^{K}[\frac{1}{\sum_{x=1}^{K}(\frac{||X_k-K_l||}{||X_k-K_x||})^{\frac{2}{n-1}}}]||X_k - \frac{\sum_{k=1}^{n} C_{kl}^{r} X_k}{\sum_{k=1}^{n} C_{kl}^{r}}||^2 \tag{5}$$

where C_{kl} denotes the membership value obtained by fuzzy logic computation for which point X_k lie in cluster 'l'. The value of variable 'r' is always kept as greater than 1.

2.5 McCulloch Pitts Neuron

It is a very simplistic computational neural model [11]. It is divided into two sections, using a two-function rule, *i.e.* 'g' and 'f'. The 'g' function takes a binarily assembled input. The 'f' function takes a decision according to the value received from the 'g'

function. Below, MP Neuron will be addressed with the mathematical representation through Eq. (6).

$$
\begin{aligned}
F(x_1, x_2, x_3 \ldots, x_n) = F(x) = \sum_{j=1}^{n} z_j \\
Y = G(F(x)) = 1, F(x) \geq \phi \\
Y = G(F(x)) = 0, F(x) < \phi
\end{aligned} \tag{6}
$$

x_i represents the interface 'u' binary ranking values for any object 'x'. The forecast outcome is '1' if the value is greater than that estimated ϕ, otherwise the expected value will be 0. The optimal value of ϕ is decided by minimizing the loss obtained by the loss function in Eq. (7).

$$
Loss = \sum_{j=1}^{n} (P_i - A_i)^2 \tag{7}
$$

The actual and predicted value is denoted by A_i and P_i respectively. The ϕ value is decided by analyzing the loss function. The point at which the loss function yields minimum loss, we use that ϕ value in our computation.

	Classified: 1	**Classified: 0**
Predicted: 1	*True Positive (TP)*	*False Positive (FP)*
Predicted: 0	*False Negative (FN)*	*True Negative (TN)*

2.6 Metrics for Performance Evaluation

Precision and Recall and performance metrics are the main criteria for evaluating our classification accuracy. The classification efficiency of our model is expressed by the confusion matrix [12].

Precision represents the correctly expected (good) findings of overall valid observations [13]. The low rate of false positive is a high precision value. The mathematical representation for precision calculation is given by Eq. (8).

$$
Precision(P) = \frac{TP(True\ Positive)}{TP(True\ Positive) + FP(False\ Positive)} \tag{8}
$$

Recall is also known as sensitivity. It measures the ratio of correctly forecast positive observations to the actual class total observations [13]. The value above 0.5 recall is usually called good. The mathematical representation for computation of recall is given by Eq. (9).

$$
Recall(R) = \frac{TP(True\ Positive)}{TP(True\ Positive) + FN(False\ Negative)} \tag{9}
$$

3. PROPOSED FLOW OF RECOMMENDATION MODEL

The proposed model starts by converting the data into a structured utility matrix and then using imputation techniques in order to reduce dataset sparsity. The process concludes with the grouping of fascinating and uninteresting element classes of the corresponding suggestion chart. Precision and recall values were used to test the performance of the model. The process flow model proposed is communicated through Fig. 1. The R and Python languages have been used for all five phases of the layout. As we have cut the utility matrix based on the member values obtained by Fuzzy c-means, the computation time of the model is less than other models. This protected the model from overly determined random users that were less close. In the first step, the MovieLens 100K dataset is

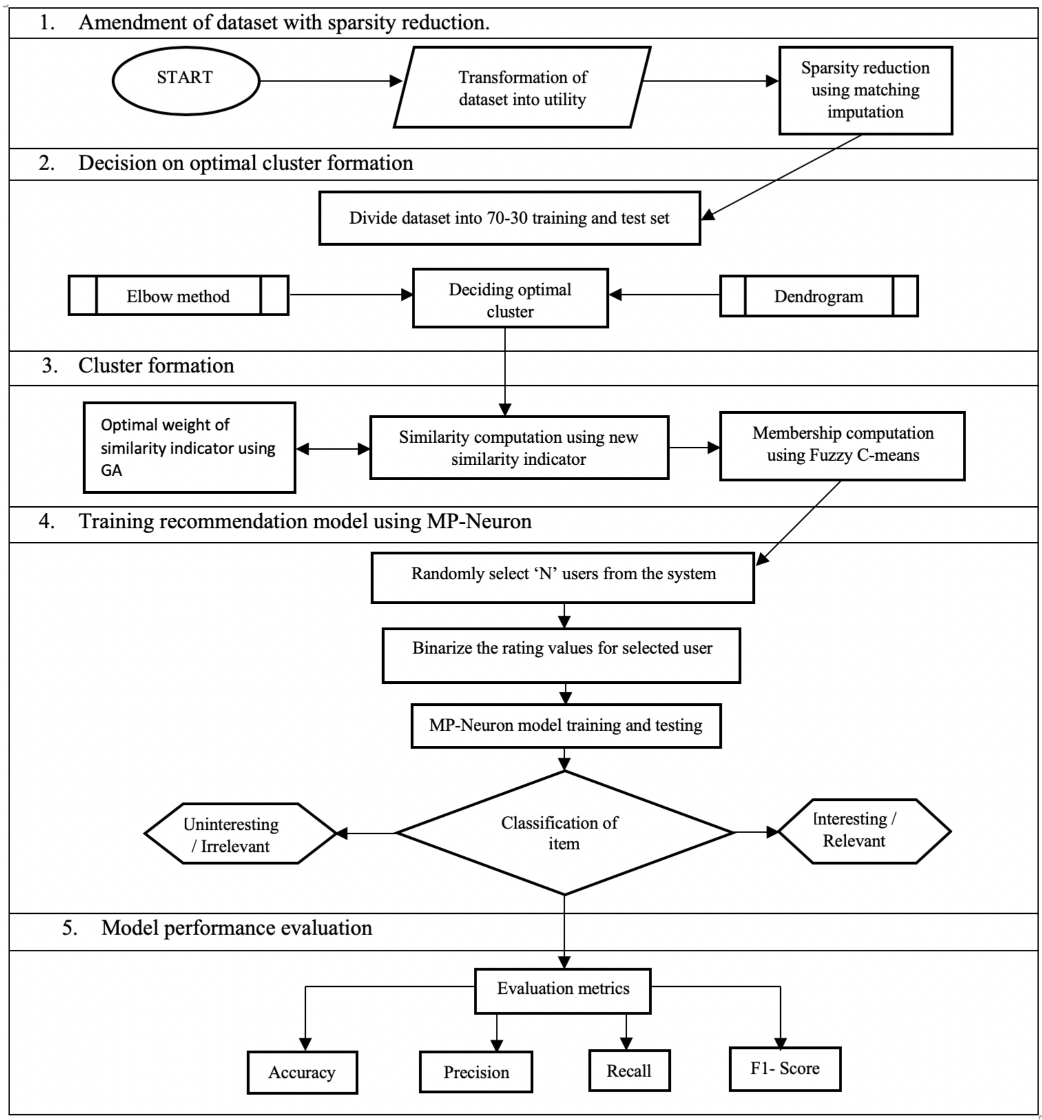


Fig. 1. Overall process flow model.

translated into a utility matrix (user rating), so that the column is the people, and the rows are the objects. The unrated items have been substituted by the most frequent user rating and the most commonly rating received by unrated item. Further in second step, by using the elbow method on the transformed utility matrix with imputed values, we decide the optimum number of clusters. Another technique named Dendrogram checks the optimal cluster analyzed by the elbow method. In the third step of proposed model, the similarity between the users are calculated using a new technique of similarity and clusters are formed using Fuzzy c-means. The optimal weights of parameters available in new similarity indicator is decided using genetic algorithm. The description of the range of weigh parameters has been discussed in Section 2.2. The Fuzzy c-means methodology brings us the participation values in any specific cluster of each individual. We then use only that cluster where we want to make recommendations to any particular user. Therefore, the efficiency matrix is trimmed based on membership values and the code sophistication of the decision process is reduced in turn. In order to test the proposed model, we have chosen 10 random users in fourth step of proposed model. We picked the category they belong to and only searched for the users in that cluster. The functionality matrix is supplied in a binarized form to the MP-Neuron model. Binarization is achieved with rating $>$ 3 as important and interesting components otherwise, irrelevant. The MP Neuron model sums up the binary data and measures the threshold value. By measuring the loss function value at the expected and real level, the threshold value is determined. The obtained resultant set of items is classified broadly into two categories namely: Interesting/Relevant and Uninteresting/Irrelevant. Fig. 2. depicts the functioning of MP-Neuron model. Precision and Recall are used to evaluate model performance. High precision and recall values represent the superiority of our proposed model.

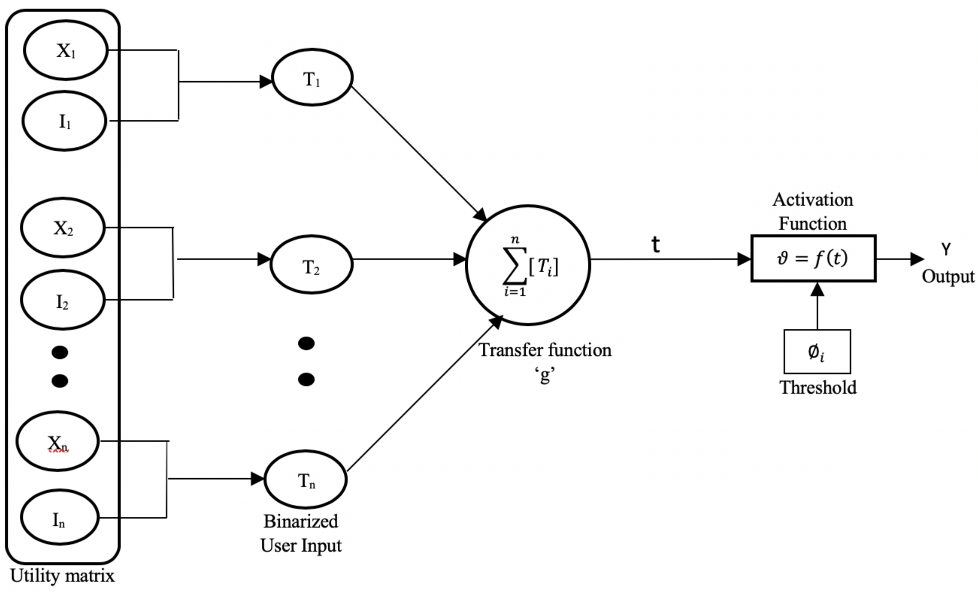


Fig. 2. MP neuron model to classify interesting and uninteresting items.

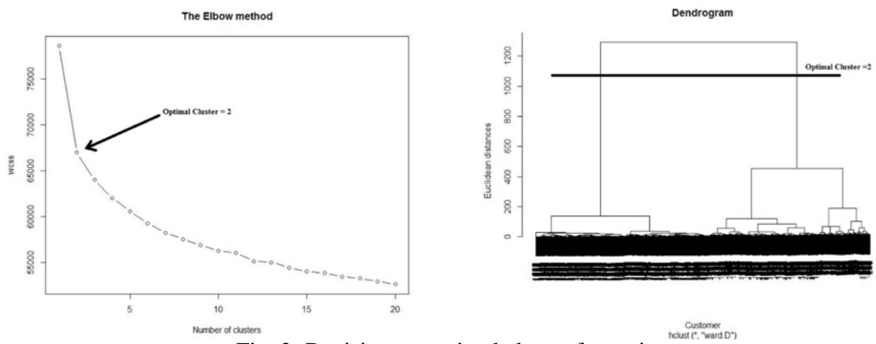


Fig. 3. Decision on optimal cluster formation.

4. DEPLOYMENT AND TESTING

In this portion, we addressed the specifics of our concept implementation. We divided our data set into 70% training set and a 30% test set after we had amended the data and imputed the missing values. The definition of the training dataset and sparsity cuts are conveyed by Table 1.

Table 1. Sparsity cutback in training set.

Total possible ratings in training data	147615
Available rating	10809
Percentage of missing values (Sparsity)	92.67%
Ratings generated using matching imputation	136806

By evaluating the visual representation of Elbow method and Dendrogram approach as shown in Fig. 3, we evaluate the optimal number of clusters to be formed. When gazing at the sharp curve produced using the elbow method, we can see distinctly that the magnitude of the WCSS (Within Cluster sum of squares) increases with expanded attention of a cluster. Likewise, we can validate the optimum cluster formation number as 2 with the analysis of the maximum distance threshold value obtained from dendrograms. Euclidean distance is the measurement used by the dendrogram approach.

After we have determined the optimal number of clusters as 2, the similarities between users are calculated through a new similarity technique, as discussed in Section 2.2. The high value of similitude is reflected in the high interest similarity in common articles. Then we pass the similitude levels of the Fuzzy c mean with the number of cluster formation equal to 2 using the similarity indices obtained after the similarity calculation phase. The Fuzzy c-mean implies that each individual has affiliation values for their participation in a common cluster. For example:

User Id	Membership value for Cluster 1	Membership value for Cluster 2
265	0.718792	0.281208

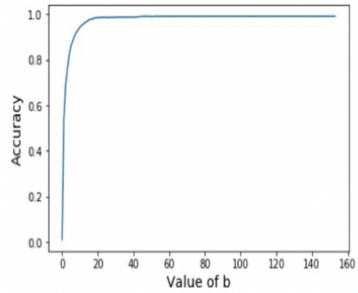
The higher participation value means that the individual is primarily in that class. The Fuzzy c-means logic tends to trim the utility matrix by allowing the model to choose the optimal cluster category and thus prevents the model from heavily computing meaning-

Table 2. MP neuron model learning on train data.

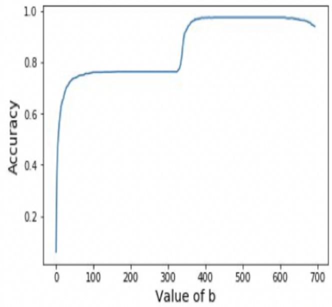
Learning of MP Neuron model for:
U_ID – 512
U_ID – 273
U_ID – 403
U_ID – 721
U_ID – 675

Set of Users (TEST)

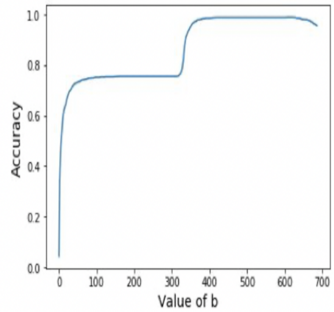
X-axis: 'b' value. Here, 'b' denotes ∅ of our learning algorithm
Y-axis: It represents the model accuracy which is calculated via examining Loss function.



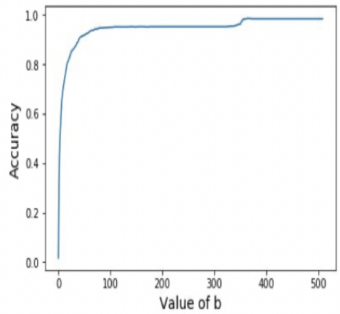
U_ID - 512



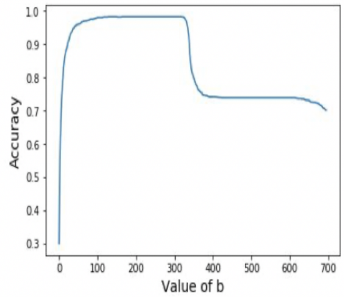
U_ID - 273



U_ID - 403



U_ID - 721



U_ID - 675

less user rate results. We have chosen the cluster for the selected user to be tested and predict test items using MP Neuron model, as outlined in the next paragraph. We binarize the ranking values in the utility matrix for the chosen consumer and move it on to 'g' as described in section 2.5. By learning the algorithm as shown in the Table 2, we have chosen the optimal value of ϕ. The value of ϕ is decided after analyzing the point at which the model received maximum accuracy for our 'f' function. Upon selecting the best value of ϕ we are using the same value to check our model on a test dataset. After testing the model performance using accuracy, precision and recall, we conclude that the model yields better performance.

5. RESULTS AND DISCUSSION

The model performance has been analyzed using several evaluation metrics namely: Precision, and Recall. The high value for precise measurement reflects the success of our model with a precise value selection of ϕ. As we have reduced large number of irrelevant users from the system, the computational time required for model training and prediction has been reduced. Table 3 depicts the performance evaluation of our proposed model.

Table 3. Evaluation result on test users.

Evaluation results for:

U_ID – 512
U_ID – 273
U_ID – 403
U_ID – 721
U_ID – 675

Set of Users (TEST)

Evaluation Metrics:
Precision, Recall, F1-score and Accuracy.

User: 512

```
Confusion Matrix :
[[497   0]
 [  7   0]]
Accuracy Score : 0.9861111111111112
Report :
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       497
           1       0.00      0.00      0.00         7

    accuracy                           0.99       504
   macro avg       0.49      0.50      0.50       504
weighted avg       0.97      0.99      0.98       504
```

User: 273

```
Confusion Matrix :
[[494   0]
 [ 10   0]]
Accuracy Score : 0.9801587301587301
Report :
              precision    recall  f1-score   support

           0       0.98      1.00      0.99       494
           1       0.00      0.00      0.00        10

    accuracy                           0.98       504
   macro avg       0.49      0.50      0.49       504
weighted avg       0.96      0.98      0.97       504
```

User: 403

```
Confusion Matrix :
[[497   0]
 [  7   0]]
Accuracy Score : 0.9861111111111112
Report :
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       497
           1       0.00      0.00      0.00         7

    accuracy                           0.99       504
   macro avg       0.49      0.50      0.50       504
weighted avg       0.97      0.99      0.98       504
```

User: 721

```
Confusion Matrix :
[[497   0]
 [  7   0]]
Accuracy Score : 0.9861111111111112
Report :
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       497
           1       0.00      0.00      0.00         7

    accuracy                           0.99       504
   macro avg       0.49      0.50      0.50       504
weighted avg       0.97      0.99      0.98       504
```

User: 675

```
Confusion Matrix :
[[470   0]
 [ 11  23]]
Accuracy Score : 0.9781746031746031
Report :
              precision    recall  f1-score   support

           0       0.98      1.00      0.99       470
           1       1.00      0.68      0.81        34

    accuracy                           0.98       504
   macro avg       0.99      0.84      0.90       504
weighted avg       0.98      0.98      0.98       504
```

Table 4. Parameters of MP Neuron model for testing users.

U_ID	'ϕ'	Train	Accurate	Accuracy	Test	Accurate	Accuracy	cluster	#users (cluster)
512	116	1178	1158	98.98%	504	497	98.61%	2	154
273	550	1178	1165	98.36%	504	494	98.01%	1	694
403	607	1178	1147	97.89%	504	497	98.61%	1	687
721	364	1178	1166	98.47%	504	497	98.61%	2	509
675	319	1178	1160	98.30%	504	497	98.81%	1	696

Table 5. Performance comparison with few existing models.

Methodology	Recall	Precision
MovieMagician Hybrid [14]	56%	73%
OurSystem Genres-based [15]	62%	61%
OurSystem Stars-based [15]	62.5%	59.3%
OurSystem Synopsis-based [15]	65.6%	59.4%
MovieMagician Clique-based [14]	73%	74%
MovieLens system Collaborative System	74%	66%
MovieMagician Feature-based [14]	75%	61%
OurSystem Collaboration [15]	78.5%	72%
Proposed model	**81%**	**77%**

We will verify the confusion matrix and their precision and recall values for each checked user. The model parameters for which the results shown in Table 3 were obtained has been discussed in Table 4. Table 3 depicts the top 5 best results obtained from the model. Average precision and recall value obtained from the proposed model is 77% and 81% respectively. Table 4 shows the number of users in a specified user group. It shows that while we perform predictions, we have reduced the list of users to minimize the complexity of the model. Since the final result set classifies the items as interested (user-relevant) and uninteresting (non-user-relevant), the formation of redundant items is ultimately reversed and this resolves the issue of over-specifications. We compared our obtained results with numerous state-of-art comparative models as discussed in Table 5. Our proposed model gave promising results as compared to existing models. High recall value achieved by the model reflects model efficiency in terms of finding all relevant items from the dataset. High precision of proposed model expresses the proportion of items our recommender model predicted as relevant were actually relevant.

6. CONCLUSION AND FUTURE DIRECTION

In this paper, a fuzzy MP Neuron model to enhance prediction precision with reduced measurement sophistication is proposed. Even with large datasets like 1M, 10M and 20M, the computation period appears to be feasible. The viability of the proposed model is demonstrated by several interventions that contributed to improved performance of the experiment. The proposed model introduced a weight variable that represents the importance of the element when measuring similarities among any number of users. Contrary to K-means, consumers are not required to lie in a group and thus we are encouraged to use Fuzzy c-means against conventional K-means. After testing the model performance using Precision and Recall, it is observed that the proposed model outperforms several state-of-art comparative models.

REFERENCES

1. M. Jugovac, D. Jannach, and L. Lerche, "Efficient optimization of multiple recommendation quality factors according to individual user tendencies," *Expert Systems with Applications*, Vol. 81, 2017, pp. 321-331.
2. M. Nilashi, "An overview of data mining techniques in recommender systems," *Journal of Soft Computing and Decision Support Systems*, Vol. 3, 2016, pp. 16-44.
3. N. T. Toan, P. T. Cong, N. T. Tam, N. Q. V. Hung, and B. Stantic, "Diversifying group recommendation," *IEEE Access*, Vol. 6, 2018, pp. 17776-17786.
4. A. K. Sahu and P. Dwivedi, "User profile as a bridge in cross-domain recommender systems for sparsity reduction," *Applied Intelligence*, Vol. 49, 2019, pp. 2461-2481.
5. J. Bobadilla, F. Ortega, A. Hernando, and J. Alcalá, "Improving collaborative filtering recommender system results and performance using genetic algorithms," *Knowledge-Based Systems*, Vol. 24, 2011, pp. 1310-1316.
6. C. R. Houck, J. Joines, and M. G. Kay, "A genetic algorithm for function optimization: a matlab implementation," *Ncsu-ie tr*, Vol. 95, 1995, pp. 1-10.
7. M. Syakur, B. Khotimah, E. Rochman, and B. Satoto, "Integration k-means clustering method and elbow method for identification of the best customer profile cluster," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, Vol. 336, 2018, p. 012017.
8. W. Huang and A. Ribeiro, "Hierarchical clustering given confidence intervals of metric distances," *IEEE Transactions on Signal Processing*, Vol. 66, 2018, pp. 2600-2615.
9. M. K. Najafabadi, M. N. Mahrin, S. Chuprat, and H. M. Sarkan, "Improving the accuracy of collaborative filtering recommendations using clustering and association rules mining on implicit data," *Computers in Human Behavior*, Vol. 67, 2017, pp. 113-128.
10. T. Lei, X. Jia, Y. Zhang, L. He, H. Meng, and A. K. Nandi, "Significantly fast and robust fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering," *IEEE Transactions on Fuzzy Systems*, Vol. 26, 2018, pp. 3027-3041.
11. S. Chakraverty, D. M. Sahoo, and N. R. Mahato, *Concepts of Soft Computing: Fuzzy and ANN with Programming*, Springer, Singapore, 2019.
12. C.-F. Tsai and C. Hung, "Cluster ensembles in collaborative filtering recommendation," *Applied Soft Computing*, Vol. 12, 2012, pp. 1417-1425.
13. M. S. Sajjadi, O. Bachem, M. Lucic, O. Bousquet, and S. Gelly, "Assessing generative models via precision and recall," in *Advances in Neural Information Processing Systems*, 2018, pp. 5228-5237.

14. S. Grant and G. I. McCalla, "A hybrid approach to making recommendations and its application to the movie domain," in *Proceedings of Conference of the Canadian Society for Computational Studies of Intelligence*, 2001, pp. 257-266.
15. C. Christakou, S. Vrettos, and A. Stafylopatis, "A hybrid movie recommender system based on neural networks," *International Journal on Artificial Intelligence Tools*, Vol. 16, 2007, pp. 771-792.

Bam Bahadur Sinha received the M.Tech degree in Computer Science and Engineering from National Institute of Technology Nagaland, India. He is currently pursuing the Ph.D. degree in the Department of Computer Science and Engineering. His research interests include data analytics, artificial intelligence and optimization techniques.

Dr. R. Dhanalakshmi is an Associate Professor at Indian Institute of Information Technology Tiruchirappalli in CSE department. She received her first Ph.D. degree in Computer Science from Mother Teresa Women's University, India in 2011 and second Ph.D. degree in Computer Science and Engineering from JNT University Hyderabad, India in 2016. Her research interests include computer networks, artificial intelligence and optimization techniques.