International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-9 Issue-1, May 2020

Enhancing Item-Based Collaborative Filtering with Item Correlations for Music Recommendation System

M. Sunitha, T. Adilakshmi, Mir Zahed Ali

Abstract*: *Music recommendation systems are playing a vital role in suggesting music to the users from huge volumes of digital libraries available. Collaborative filtering (CF) is a one of the well known method used in recommendation systems. CF is either user centric or item centric. The former is known as user-based CF and later is known as item-based CF. This paper proposes an enhancement to item-based collaborative filtering method by considering correlation among items. Lift and Pearson Correlation coefficient are used to find the correlation among items. Song correlation matrix is constructed by using correlation measures. Proposed method is evaluated on the benchmark dataset and results obtained are compared with basic item-based CF.

Keywords: Music recommendation system, Collaborative filtering, User-based CF, item-based CF, Lift, Pearson Correlation coefficient, Song correlation matrix

I. INTRODUCTION

Because of music streaming services such as Last.fm, Wink, Gaana, Spotify, Pandora, Apple Music etc, users interested in music are able to access millions of songs. This growth leads to the problem of information overloading [1] i.e. difficult to find interesting music. Music recommendation systems (MRS) are playing vital role and helping users to deal with huge search space. Research in MRS has gained lot of interest from academia and industry. MRS handle the huge volume of data by using filters and provide suggestions which fit the interest of users. Most of the MRS currently available is able to filter the songs and give limited choices to the users. But as the musical taste of a user is quite complicated and is very difficult to produce satisfactory suggestions always. Research in MRS is divided into Collaborative Filtering, Content-based Filtering and Hybrid approach. The core concept in Collaborative filtering is the interaction between user and item whereas Content-based filtering is dependent on the features of the item. Hybrid approach combines Collaborative and content based methods.

Collaborative filtering faces with the challenges such as Cold-start, Popularity bias, Sparsity and Long-tail problem[1][3]. This paper addresses the Popularity bias by considering item correlations.

Revised Manuscript Received on May 15, 2020.

***** Correspondence Author

***M. Sunitha**, Assistant Professor, Department of CSE, Hyderabad, (Telangana), India, E- mail id: sashu2006@gmail.com no 9849212814

Dr. T. Adilakshmi, Head & Professor, Department of CSE, Hyderabad, (Telangana), India, E-mail: t_adilakshmi@rediffmaill.com, no 9908911700

Mir Zahed Ali, M.Tech Student, CSE, Hyderabad, (Telangana), India. E-mail: mirzahdali@gmail.com, no 6304140472

The rest of the paper is organized as follows. Related work is described in Section 2. Proposed algorithm is explained in Section 3. Section 4 showcases the results obtained for the item based clusters combined with correlations and Section 5 describes conclusion and future directions of research.

II. RELATED WORK

Major components of a recommendation system are Users, Items and Recommendation engine. Let U={U1,U2,..........,Un} be the n number of users and I={I1,I2,.............Im} be the m number of items, the ratings of each user is recorded for each item in a data structure known as User-Item matrix as shown in the fig.1.

Fig.1 User-item matrix

User/Item	I_1	I_2	I_3	I_4	I_5	....	…	I_m
U_1	12	0	0	0	3			2
U_2	0	23	4	0	0			0
U_3	1	8	0	0	0			4
…								
….								
U_n	0	7	0	0	4			5

Each entry Rij is the number of times Ui listened item Ij. As is it clear from the user-item matrix, the rating matrix is very sparse. User-item matrix is the basic data available for recommendation system. Objective of a music recommendation system is to suggest music interesting to the users and also help users in discovering new artists, songs based on their interest. Many music service providers such as Allmusic, Pandora, Audiobaba6 , Mog7 , Musicovery 8 , Spotify9 , Apple "Genius" aggregated millions of users and suggest music based on their interest. Music is different from other items such as books, movies, restaurants etc. User ratings are obtained from implicit feedback i.e. if a user listen to a particular song x number of times, this frequency is considered as the implicit rating of that user. Most of the research in this field focused at suggesting a list of artists and a sequence of songs (playlist addressing personal interest of user). Research done in music recommendation system can be categorized as Demographic based model, Collaborative filtering model and Content based model[4][5]. Context based recommendation systems are also proposed in recent times.

A. Demographic-based model: It is the easiest model in music recommendation system and based on the metadata such as the title of the song, artist name, and lyrics to find the target songs. Even though it is easy and fast, this model requires user to know about the metadata of the songs and difficult to maintain metadata. The important limitation of this method is users will never get a chance to explore new and novel songs [3].

B. Collaborative Filtering model: This is the most fundamental and most popularly used model for music recommendation system. The underlying principle of CF based method is if two users have similar taste in the past i.e. they likes few songs in the past, then they may also like similar kind of songs in the suture . In CF most commonly used approach is to find nearest neighbors for any given user to provide recommendations. Collaborative filtering is further classified into three subcategories given as memory-based, model-based, and hybrid collaborative filtering. Memory-based Collaborative Filtering is to predict the item based on the entire collections of previous ratings. Model-based Collaborative Filtering In contrast to memory-based CF, model-based CF uses machine learning and data mining algorithms which allow the system to train and model the users' preferences. Hybrid Collaborative filtering model is to make prediction by combining different CF models.

C. Content based model: Content based model takes musical content such as rithym, tempo etc. into consideration for recommendation. Some of the content based models also take lyrics of the music for recommendation. The major drawback of the content based models is finding meaningful recommendations based on the content. Another issue with content-based model is the complexity of the algorithm as the size of the data increases.

III. PROPOSED SYSTEM

Proposed method is implemented by using the item correlations. This section explains about the dataset, steps used in the proposed approach. Lift and Pearson correlation coefficients are used to find the song correlation matrix.

A.Obtaining user-item matrix

The dataset considered in this research work is obtained from Last.fm. Statistics about the dataset are given below

Total Lines:	19,150,868
Unique Users:	992
Artists with MBID:	107,528
Artists without MBDID:	69,420

Each user's activity is recorded consists of the fields such as UserID, Timestamp, AlbumID, AlbumName, TackID, TrackName. One such sample is shown in the fig.3.1.1.

Data pre-processing is performed to handle noisy data and missing values.. User-item rating matrix obtained after pre-processing is a matrix of dimensions 200 X14458. Sample user-item matrix is shown in fig.3.1.2.

B.Forming item clusters

Item-based collaborative filtering is based on the principle that "similar songs of the songs already listened by a target user is used to provide recommendations". Most of the research is carried out in this area by using nearest neighbor songs of the target users preferred song. This research work is focused on finding item clusters which in turn are used in recommendations. Similar to user-based CF, item based CF also considers user-item matrix for training set as the basis to form item clusters. Every song is considered as a vector in terms of users as given in table 3.2.1.

Table 1 Song Vector

U1	U2	U3	U4	....	....	....	U198	U199	U200
12	0	0	14				0	6	0

K-Means clustering as given in fig 3. is used to find item clusters. Optimal value of K is calculated by using Elbow method. K value obtained in the research work presented in this thesis is 160.

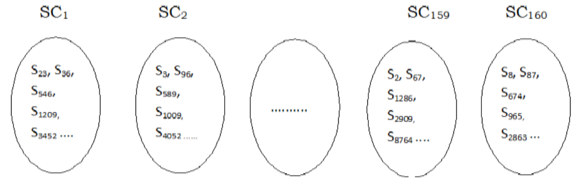

Fig 2. Sample song clusters

Here SC_1,SC_2,...Represent Song Clusters

Algorithm item_based_K-means()

Input: Item-user matrix

Output: K item clusters

Method:

Begin

1. Find_K_initialclusters();
2. For i=1 to n
3. Compute distance from i^{th} item to each initial cluster centre
4. Assign item to the closest centre
5. Update the centers based on the assignment in step 4
6. Repeat step 4 & 5 until the centers remain same

End

Fig.3. Item_based_K-means algorithm

Algorithm Find_initial_Kclusters()

Input: user-item rating matrix

Output: K- Number of clusters

Method:

Begin

1. Compute proximity matrix using any of the similarity or dissimilarity measures
2. Consider threshold value α
3. //For each item find the number of neighbors within the threshold distance α
   For i in 1 to n
   For j in 1to n
   Count (Distance(Si,Sj) < α)
   Compute K value from step 3

End

Retrieval Number: A2016059120/2020©BEIESP
DOI:10.35940/ijrte.A2016.059120

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

C. Mapping test users to the nearest cluster

Test dataset is considered to validate the song clusters formed. Each test user is the songs listened by the test user is considered to find the closest song cluster. For example let us consider a test user shown in the table 3.3.1. if S1 , S3, S4, are the first few songs in the listening history of a test user. Song vectors of these songs are used to find mean song vector to represent the test user vector. This user vector is of the same dimension as the mean vector of each song cluster. Similarly mean song vector is computed for each test user using Compute_mean_test item vector(). During the recommendation phase each test user is mapped to the closest song cluster. Songs are recommended to each test user by using the recommendation algorithm given in fig.

Table 2. Sample User Vector

S1	S2	S3	S4	....	....	. . . .	S14881	S14457	S14458
12	0	5	14				9	6	0

Algorithm Compute_mean_test item vectors()
Input: Test users user_item matrix of size 60X 14456
Ouput: mean user vectors
Method:

Begin

1. For each test user
2. For all the items listened by the user
3. Get the item vector from the original user-item matrix
4. Find the mean item_vector for the test user

End

D.Finding song correlation matrix

In the process of enhancing item-based collaborative filtering method, item correlations are computed using Lift and Pearson Correlation coefficient.

Lift: Lift is an objective measure of interestingness used in various fields such as data mining, statistics etc. It is one of the measures used to find interesting association rules. Lift is a numerical measure used to determine how items are correlated. Support and Confidence are the most commonly used measures but they fail to find association between negatively related items. Consider the example in table given below. Here we are trying to find relation between Orange buyers and apple buyers.

Support (orange, apple)=2

Confidence({orange}->{apple})=Support(orange, apple) = (2/20) = 28%
 Support(orange) (7/20)

But support of apple buyers is 40%, which is contrasting the statement that 28% of orange buyers also buys apple. Here orange buyers and apple buyers are negatively related. In such cases support and confidence framework does not give useful information. Lift is an alternate measure that can be used. Contingency table to show the relation between Apple and Orange buyers

	Buy apple	Not buy Apple	Row total
Buy Orange	2	5	7
Not Buy Orange	6	7	13
Column total	8	12	20

Lift between two items is defined by using the equation (1). Lift measure value ranges from -1 to 1. Lift value decides the relation between items as shown in below

$Lift(U_i,U_j)$ = <1 Ui, Uj are negatively related
>1 Ui, Uj are positively related
=1 Ui, Uj are not related i.e. independent of each other

Lift(orange, apple)= (2/20)/ (7/20)* (8/20)
= (2 *20)/(56)=40/56=0.714 <1.

So orange buyers and apple buyers are negatively related.

$$Lift(Ii,Ij) = \frac{s(Ii \cup Ij)}{s(Ii)s(Ij)} \text{ ---------(1)}$$

Fig. 4. Sample user-item matrix

User/Item	Item 1	Item 2	Item 3	Item 4	Item 5
User1	0	4	0	6	0
User2	2	3	0	1	0
User3	1	4	0	1	1
User4	6	0	0	4	5
User5	0	5	3	0	0

Pearson Correlation coefficient

Pearson Correlation Coefficient (PCC) is one of the most popular similarity measures for Collaborative filtering recommender system, to evaluate how much two items are correlated. PCC is defined as

$$PCC(Ii,Ij) = \frac{\sum_k (R_{ki} - R_i^{-})(R_{kj} - R_j^{-})}{\sqrt{\sum_k (R_{ki} - R_i^{-})^2 ((R_{kj} - R_j^{-})^2}}$$

R_{ki} is the number of times k^{th} user listened to S_i song, R^{-}_i is the average frequency of song S_i. Similarly R_{kj} is the number of times k^{th} user listened to S_j song, R_j^{-} is the average frequency of song S_j.

PCC values ranges from 0 to 1.

If PCC(Ii , Ij) >0 S_i,S_j are positively related
<0 S_i,S_j are negatively related
=0 S_i,S_j are independent

User-item matrix given in the fig. 4. .is considered and sample correlation matrix is obtained as given in the table 3 and 4 .

E. Combining item clusters with item correlations

Each test user is mapped to the nearest song cluster.

Let the list of songs in the mapped cluster be $\{S_1,S_2,\ldots\ldots S_k\}$. Consider the 10% of the songs listened by each test user to find the correlation with the songs from the mapped cluster. Consider only positively correlated songs in the recommendation process. The procedure to combine item clusters with correlation is shown in the fig.3.4.3.1.

Algorithm Item_correlation_Lift()
Input: nearest song cluster, test user-item matrix
Output: recommendation vectors for test users
Method:
Begin

1. For each test user in $\{U_{141},U_{142}\ldots U_{200}\}$
2. Find the correlation with each song in the mapped cluster
3. List the songs positively correlated to test user
4. Recommend the songs listed in step 3

End

Table 3. Song Correlation Matrix With Lift

Item/Item	$Item_1$	$Item_2$	$Item_3$	$Item_4$	$Item_5$
$Item_1$	1	1.25	0	1.25	1.67
$Item_2$	1.25	1	1.25	0.98	0.62
$Item_3$	0	1.25	1	0	0
$Item_4$	1.25	0.98	0	1	1.25
$Item_5$	1.67	0.62	0	1.25	1

Table. 4 Song Correlation Matrix With Pcc

Item/Item	$Item_1$	$Item_2$	$Item_3$	$Item_4$	$Item_5$
$Item_1$	1	0.82	-0.22	0.48	0.93
$Item_2$	0.82	1	0.33	0.24	-0.04
$Item_3$	-0.22	0.33	1	-0.06	-0.02
$Item_4$	0.48	0.24	-0.06	1	0.12
$Item_5$	0.93	-0.04	-0.02	0.12	1

IV. PERFORMANCE EVALUATION MEASURES

Recommendation systems are the software tools useful to solve information overloading problem. RS are popular in both academics and industry. They help service providers in retaining users and users in reducing searching cost and decision making. Vast amount of research has been done in recommendation systems and many algorithms have came in to existence. It is very important to evaluate and compare the performance of RS. The research work done in this thesis also proposed and implemented new approach for music recommendation system. Evaluation measures are used to compare the newly proposed algorithm with already existing algorithms. Music recommendation systems suggest top-N songs to test users. The performance of a recommendation system depends on the number of songs liked by the user from the recommended list. So to achieve this we have used evaluation measures from information retrieval such as Precision, Recall and F-Measure. These measures are obtained by using confusion matrix as given in fig.5.

Fig..5. Confusion matrix

Actual/Recommended	Recommended to test user	Not recommended to test user
Listened by test user	True positives(TP)	False negatives (FN)
Not listened by test user	False positives (FP)	True negatives (TN)

True positives (TP) indicate the songs actually listened by a test user and also recommended by the recommendation system.
True negatives (TN) indicate the songs actually not listened by a test user and also not recommended by the recommendation system.
False positives (FP) indicate the songs actually not listened by a test user but recommended by the recommendation system.
False negatives (FN) indicate the songs actually listened by a test user but not recommended by the recommendation system.
Precision is used to indicate how many songs recommended by the system actually liked by a test user as given in equation shown below

$$Precision\,(Ui) = \frac{Number\ of\ songs\ actually\ listened\ by\ a\ test\ user}{Total\ number\ of\ songs\ recommended} = \frac{TP}{TP+FP}$$

For each test user precision is calculated using the equation mentioned above. Average precision for all test users are calculated as given by

$$Average\ Precision\,(AP) = \sum_{i=1}^{m} \frac{Precion\,(U_i)}{m}$$

mAP of the recommendation system is calculated as the mean of average precisions over k cross validations.

$$mAP = \sum_{i=1}^{k} \frac{AP_i}{k}$$

Recall indicates fraction of songs recommended out of total songs listened by a test user as given below

$$Recall\,(U_i) = \frac{TP}{TP+FN}$$

Average recall for all test users is found by using the equation given below

$$Average\ Recall\,(AR) = \sum_{i=1}^{m} \frac{Recall\,(U_i)}{m}$$

$$mAR = \sum_{i=1}^{k} \frac{AR_i}{k}$$

mAR is calculated for each composition of training and test data.
F-measure is a harmonic mean of precision and recall. It is calculated for each test user by using the equation given below

$$F-measure\,(U_i) = \frac{Precision(U_i) + Recall\,(U_i)}{2 * Precision(U_i) * Recall\,(U_i)}$$

V. RESULTS AND DISCUSSION

As discussed in section II, item based clusters algorithm is implemented on the user-item matrix obtained from Last.fm. Each users listening history is available for three years from 2006 to 2009. After performing data pre-processing, item-based clustering algorithm is applied on the resulting user-item matrix. K value to form item clusters are obtained by using Elbow method. K obtained for 200 sample users is 65 in this research work. Clusters are obtained by applying different proximity measures such as Euclidean distance, Manhattan distance, and Supremum diatance Precision, Recall, F-measure and Accuracy are calculated with all the proximity measures. Obtained results are compared by combining item-clusters with correlation. Comparison of this proposed method with item clusters are shown in the figures shown below.Table 6. shows precision for item-based clusters is obtained with various proximity measures. Manhattan distance is giving good precision compared to other proximity measures. Other evaluation measures are also mentioned along with precision

Table 5. Comparison of Precision, Recall, F-Measure And Accuracy For Different Proximity Measures

		Euclidean	Supremum	Manhattan
Precision		0.13	0.14	0.21
Recall		0.27	0.22	0.11
F-measure		0.13	0.12	0.09
Accuracy	All Songs	0.66	0.71	0.85
Precision		0.11	0.11	0.12
Recall		0.01	0.02	0.05
F-measure		0.02	0.03	0.06
Accuracy	Top-50 Songs	0.89	0.88	0.88
Precision		0.11	0.11	0.11
Recall		0.01	0.02	0.02
F-measure		0.02	0.03	0.03
Accuracy	Top-60 Songs	0.88	0.88	0.88

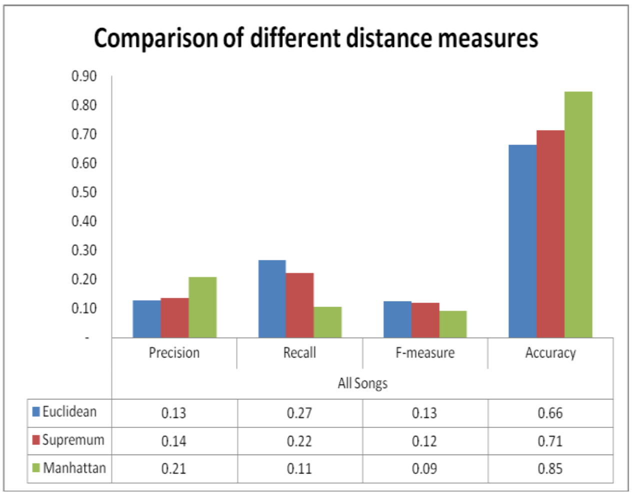

Fig. 6. Comparison of different distance measures for All songs recommendation

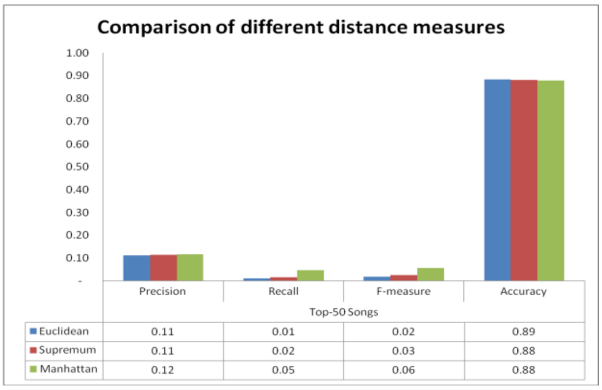

Fig. 7. Comparison of different distance measures for Top-50 songs recommendations

Table. 6. Comparison of Precision, Recall F-Measure And Accuracy For Different Type Of User-Item Matrix

		Binary U-I Matrix	Original U-I Matrix	Normalized U-I Matrix
Precision		0.34	0.13	0.15
Recall		0.08	0.27	0.27
F-measure		0.10	0.13	0.14
Accuracy	All Songs	0.89	0.66	0.68
Precision		0.10	0.11	0.08
Recall		0.08	0.01	0.05
F-measure		0.07	0.02	0.05
Accuracy	Top-50 Songs	0.84	0.89	0.89

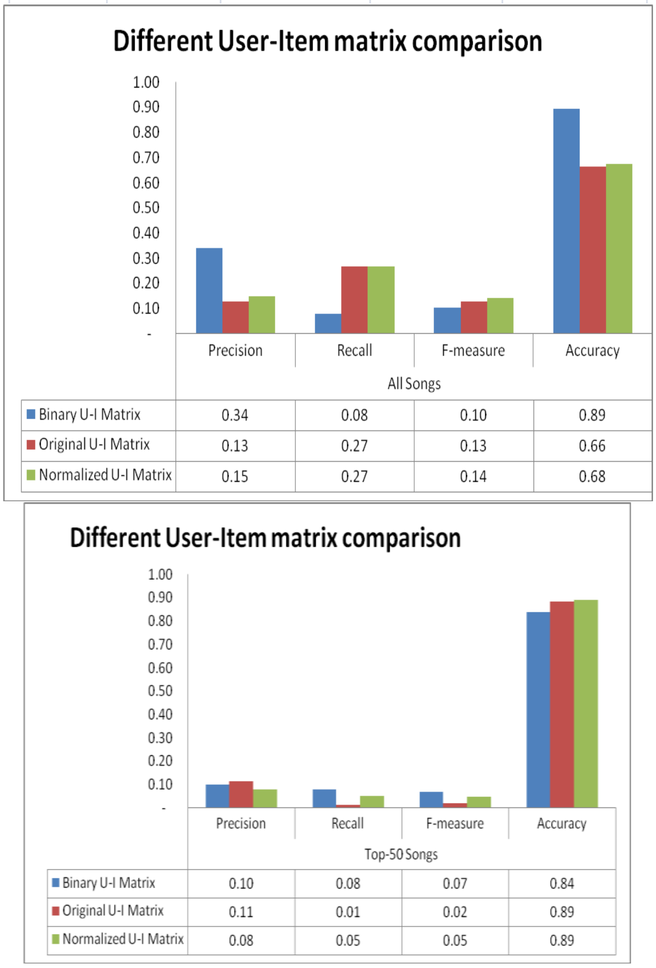

Fig. 8. Comparison of different distance measures with item-based clusters

Table. 7. Comparison of Precision, Recall F-Measure and Accuracy for Item-Based Clusters With Correlation Combined With Item Clusters

Measure	No.of Songs Recommended	Item Correlations combined with item clusters	Binary U-I Matrix	Original U-I Matrix	Normalized U I Matrix
Precision	All Songs	0.21	0.34	0.13	0.15
Recall		0.98	0.08	0.27	0.27
F-measure		0.30	0.10	0.13	0.14
Accuracy		0.88	0.89	0.66	0.68

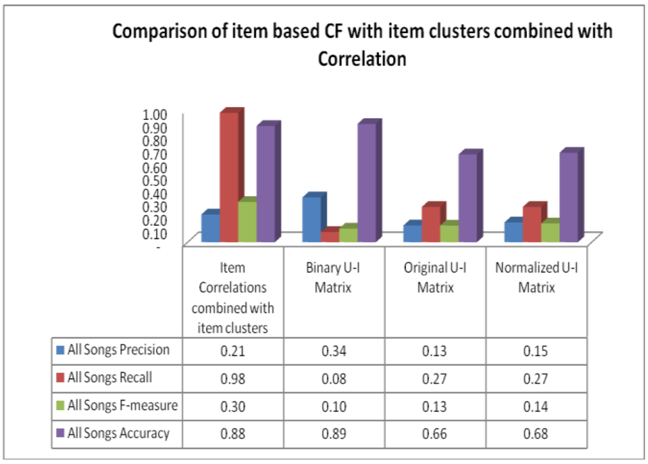

Fig.9. Comparison of item-based CF with item correlations

VI. CONCLUSION

In this research work, comparison of item-based clusters is performed with combining item correlations. Initially data-preprocessing is completed by handling missing values, noise and random errors. User-item matrix is constructed from cleaned data. Elbow method is used to find the optimal value for K, to be used in cluster formation. With the obtained K value, item clusters are formed. Various proximity measures are used to compare the precision of recommendation system. These results are compared with item correlations combined with item-based clusters. The results obtained are better compared to the item-based clusters.As music taste of a user depends on multiple parameters, the proposed recommendation approach can be improved by including context of item also into consideration.

REFERENCES

1 Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl, Item-Based Collaborative Filtering Recommendation Algorithms, May 1-5, 2001, Hong Kong. ACM 1-58113-348-0/01/000

2 Xiaoyuan Su , Taghi M. Khoshgoftaar,A Survey of Collaborative Filtering Techniques, Advances in Artificial Intelligence Volume 2009, Article ID 421425, doi:10.1155/2009/421425

3 Anand Neil Arnold, Vairamuthu S., Music Recommendation using Collaborative Filtering and Deep Learning, International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-7, May, 2019

4 Nakamura, K., Fujisawa, T., & Kyoudou, T. (2017, October). Music recommendation system using lyric network. In Consumer Electronics (GCCE), 2017 IEEE 6th Global Conference on (pp. 1-2)

5 Chang, S. H., Abdul, A., Chen, J., & Liao, H. Y. (2018, April). A personalized music recommendation system using convolutional neural networks approach. In 2018 IEEE International Conference on Applied System Invention (ICASI)(pp. 47-49). IEEE

AUTHORS PROFILE

M.Sunitha She completed M.Tech from JNTU, Hyderabad in 2010. Currently pursuing Ph.D from JNTU, Hyderabad. She got more than 15 yrs experience and currently working with Vasavi college of Engineering, Hyderabad. She got more than 10 publications to her Credit.

Dr. T. Adilakshmi She completed Ph.D from HCU in 2006. Two students awarded Ph.D under her guidance. Currently guiding more than 20 scholars from JNTU and OU. She got more than 30 yrs of experience and more than 50 publications are there to her credit. Currently she is working with Vasavi college of Engineering, Hyderabad.

Mir Zahed Alir is currently pursing M.Tech from Vasavi college of Engineering