IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-9, Sept 2019 E-ISSN: 2347-2693

A Reliable Solution for Sparsity Problem in Collaborative Filtering Using Demographic Approach

Kaira Nithin Goud

Department of Computer Science and Engineering, Gitam University, Visakhapatnam, Andhra Pradesh, India

**Corresponding Author: kairanithin6921@gmail.com*

DOI: https://doi.org/10.26438/ijcse/v7i9.5459 | Available online at: www.ijcseonline.org

Accepted: 12/Sept/2019, Published: 30/Sept/2019

***Abstract*—** Now a day's online resources are increasing very rapidly like amazon and flipchart, eBay etc. The main role of recommendation systems is to provide recommendations based upon the ratings given by the users.it suffers from the sparsity to reduce that we are going to introduce a reliable solution that motives to perform better results using a demographic approach. Each prediction consorts with a reliability measure. Reliability is a measure of how liable a prediction is. So each recommendation for a user is associated with a pair of values those are Prediction and reliability. Quality of reliability is also discussed. Experimental results show that our proposed reliable solution using demographic approach has increased the overall recommendation and reduced the sparsity.

***Keywords*—**Recommender systems, Collaborative filtering, prediction, reliability, location.

I. INTRODUCTION

Collaborative filtering approaches overcome some of the limitations of content-based one. Recommender systems aim Providing suggestions to the user from the huge amount of data. They aid users in simplifying their task [3]. They may use two techniques called Collaborative filtering and content-based filtering. Content-based filtering makes use of attributes of the items and similar products will be recommended.

For example, when a user buys a Canon Camera the system starts recommending lenses, other similar model cameras. Item-Item collaborative filtering approach is based on the neighbourhood of similar items using some similarity metrics [4].All these techniques make use of ratings given by the user but there arise some problems while using these techniques. They are:

new user problem, however, new Item problem, Data Sparsity. The new user problem or the new item problem will usually arise when a new user is added. However, Recommendation systems cannot decide on the type of the items to be suggested to the user. Data Sparsity exists when there is not enough information available to make accurate predictions. The long tail problem relates to suggesting the new or unpopular items and not purely suggesting the popular items. Each technique in the recommender systems makes use of the ratings. But again there is a problem with the ratings given by the user i.e. reliability.

II. RELIABILITY MEASURES FOR RECOMMENDATIONS

Recommender systems generate suggestions based on the available information. The ratings of the users depend on the user's interest, mood and his biasedness. So, the suggestions may not always be appropriate.

According to Antonio Hernando [1], the problem of reliability of ratings can be reduced by incorporating some reliability measures and provide suggestions based on these values.

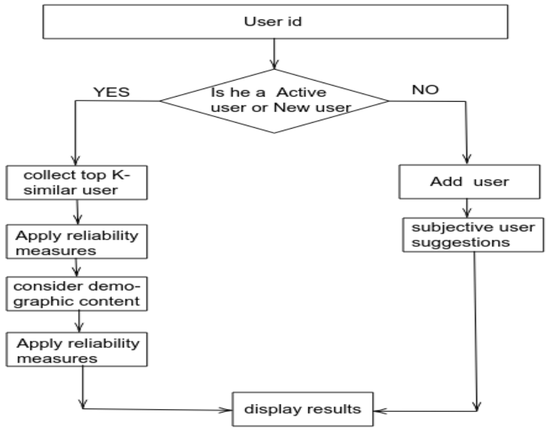
For Example, the music albums Planet Pit and Immortal were rated 4.4 and 4.1 on a scale of 5, respectively. The album Immortal was rated by 10 users whereas the other one was rated by 108 users. Here we cannot suggest only based on similar user ratings. We need some reliable ways to generate suggestions.

If the reliable measures are i) number of neighbours who have rated a particular item ii) the disagreement between the users, the reliability values for the albums Planet Pit and Immortal were 0.5, 0.8 respectively. Then by considering the item rating and reliability measure for that item as a pair recommendation will be based on reliability measures. So, the album Immortal is suggesting as it is more reliable.
In this paper, we have used demographic content in assisting the user.

Algorithm:

```
INPUT:
$I$:Set of Items
$U$:Set of Users
$U_E$:Expert User
$U_T$:Target User
Set of Unrated Items for User $U_T$ : $S_{UR}$
OUTPUT:
Top N appropriate Recommendations
Method:
Take the user ID $U_{id}$of the user and check whether he/she is existing user or not.
If$U_{id}$ is a new user:
Call “new user method”
Else:
Call “old user method”
End if
Call “Location based recommendations method”
End
Method new user():
        Get the Expert ratings dataset
        Sort the ratings of the movies and provide the suggestions.

Method old user():
    Calculate the similarity between the users.
        Calculate the predictions for the unrated items of the user
        Calculate Mean Absolute Error (MAE)
        Find the reliability measure for the predictions
        Suggest the Items to the user based on Reliability measure.
        Return;
Method LocationBasedRecommendations():
        Consider the location of the user
        Find the nearest user based on the location
    Evaluate the similarity between the nearest users.
        Calculate the predictions for the unrated user
        Calculate Mean Absolute Error (MAE)
        Find the reliability measure for the predictions
        Return;
```


User id
Is he a Active user or New user
YES
NO
collect top K-similar user
Apply reliability measures
consider demo-graphic content
Apply reliability measures
Add user
subjective user suggestions
display results

III. PROPOSED ARCHITECTURE

The above flowchart of the proposed framework takes user-id u_{id} as input and checks whether u_{id} is an existing user or not. If it is an active user, collects the top k similar users data. Apply the reliability measures on similar user data then here we consider demographic content as a key and apply the reliability measure on the result of demographic. If the $u_{id}$is not an existing user then we will add the user and give subjective suggestions. We will explain each step in detail as follows

A. Collect Top K-Similar users:

Based on the ***cosine similarity*** we will get the top k-similar users

B. Apply Reliability measure:

For the top k-similar user results from the above module, we will get the predicted values. We apply reliability measure on predicted values that specifies how much the ratings are liable.

C. Demographic content:

We will consider the locations for the existing users.

D. Apply Reliability measure:

For the demographic content results from the above module, we will get the predicted values for a specified location. We apply reliability measure on predicted values that specifies how much the ratings are liable in that location.

E. Add User:

We will add the user-related information into the dataset. We suggest subjective user information to the new user.

IV. RELATED WORK

Demographic content contains distinguishable attributes of a set of users. For example, the demographic content may include the location, age, gender, date, timestamp etc.

In this paper, we considered the location of the particular user by obtaining the PIN code of the user. Each user will be accessing the site from a particular location and that will decide the PIN code of the user. Generally, the PIN code will be assigned sequentially for every location. So the more difference between the PIN codes, the more is the distance between the locations. The reason for selecting the PIN code as an attribute, with an assumption that users of the similar location may share similar tastes and preferences [1].

A.*Equations*

Formula to calculate nearest neighbors,

$$\mathbf{D_{U1,U2}=|L_{U1}-L_{U2}|}$$

U1: User1

U2: User2
$D_{U1,U2}$:represents the distance between the users U1 and U2
L_{Ux}: represents PIN code of the user U1 and U2 respectively

Here if the modulus is not considered in the equation then there arises a problem, the nearest user will become farthest and the farthest user will become nearest because of the '-' sign.

Nearest neighbor algorithm adopted to find the nearest users. If we use reverse knn computation reduces that will help to improve the performance of the recommendation system.

Table 1. Showing users and their corresponding PIN code

User	L_{Ux}
U1	53004
U2	53002
U3	53007
U4	52415
U5	47168
U6	54258
U7	52786
U8	51236
U9	51483

If U1 is taken as the target user then the nearest users can be obtained by using the above formula and then sorting it out. The result obtained is shown in the table2.

Table 2. Showing distances between target user U1 with other users.

$D_{U1,U2}$	U1
U1	0
U2	2
U3	3
U4	589
U5	5836
U6	1254
U7`	218
U8	1768
U9	1521

V. APPLYING RELIABILITY MEASURES

Reliability measure calculated as follows [1]

$$\mathrm{R}_{u,i} = \sqrt{f_K\left(\left|K_{u,i}\right|\ \right) f_V\left(V_{u,i} \right)} = \frac{\left|K_{u,i}\right|}{3\sqrt{1+V_{u,i}}}$$

Where $R_{u,I}$ represent the reliability factor for the item i.
$|K_{u,i}|$ represents the users who have given at least some value to an item i. This is the modulus value which increases the reliability factor as its value increases.
$V_{u,I}$ represent the degree of disagreement value between the users and rating that item[1].

$$\mathrm{V}_{u,i} = \frac{\sum_{v \epsilon K_{u,i}} sim(u,v).(R_{v,i} - \bar{R_v} - P_{u,i} + \bar{R_u})}{\sum_{v \epsilon K_{u,i}} sim(u,v)}$$

Input Data: The following table shows the ratings of users for items I1 through I10. A rating to an item of -1 represents that the user has not rated that item. Here considered user-item rating matrix initially user has provided the ratings to the items based upon the ratings the respective operations performed user-user similarity and item-item similarities are performed. In this paper, the user-user similarity is calculated to recommend the items to the particular users thereafter reliability measures are discussed.

Table 3: Showing example matrix of users and rated items

	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10
U1	1	2	-1	4	2	-1	3	4	-1	4
U2	1	-1	4	5	1	5	3	4	1	5
U3	1	2	5	2	-1	1	-1	3	4	5
U4	2	1	4	4	1	-1	3	5	5	4
U5	2	2	4	-1	1	-1	3	-1	4	-1
U6	1	-1	5	2	1	-1	2	4	-1	4
U7	2	-1	-1	4	2	-1	2	5	-1	5
U8	2	2	4	4	1	-1	2	5	-1	5
U9	5	1	-1	1	5	2	5	5	-1	4

Similarity computation:
The popular similarity is used to compute the similarity between the users ie Pearson correlation similarity

$$\mathrm{PC}(u,v) = \frac{\sum\limits_{i \in \mathcal{I}_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum\limits_{i \in \mathcal{I}_{uv}} (r_{ui} - \bar{r}_u)^2 \sum\limits_{i \in \mathcal{I}_{uv}} (r_{vi} - \bar{r}_v)^2}}.$$

Example: simil(u_1,u_2)=(1−2.75)*(1−3)+(2−2.75)*(4−3)+⋯..+(1−2.75)*(1−3)/√ ((1−2.75)^2+(2-2.75)^2+…..+(1-2.75)^2)*((1-3)^2+(4-3)^2+……+(1-3)^2)
simil(u_1,u_2)= 0.9159631900820036

Table 4. Showing similarity values between users

	U1	U2	U3	U4	U5	U6	U7	U8	U9
U1	0	0.92	-0.15	0.84	-0.05	0.13	-0.49	0.91	0.09
U2	0.92	0	-0.10	0.85	0.05	0.20	0.43	0.91	0.07
U3	-0.15	-0.10	0	0.04	0.94	0.84	0.25	0.1	0.04
U4	0.84	0.85	0.04	0	0.18	0.20	-0.37	0.88	0.05
U5	-0.05	0.05	0.94	0.18	0	0.80	0.71	-0.01	-0.35
U6	0.13	0.20	0.84	0.20	0.80	0	0.16	0.34	-0.25
U7	-0.49	0.43	0.25	-0.37	0.71	0.16	0	-0.68	-0.50

U8	0.91	0.91	0.1	0.88	-0.01	0.34	-0.68	0	0.15
U9	0.09	0.07	0.04	0.05	-0.35	-0.25	-0.50	0.15	0

VI. EVALUATION METRICS

This section deals with the evaluation metrics, which focuses on how to evaluate the performance of the recommendation system in collaborative filtering. MAE and RMSE
MAE is defined as the sum of the differences between the actual value and predicted values.

$$MAE = \frac{\sum_{(u,i)\in J} |r_{u,i} - p_{u,i}|}{|J|}$$

It perform differences between the actual rating and predicted rating. The representation of the terms as follows

$r_{u,i}$= actual rating provided by the users.

$P_{u,i}$.= predicted rating

where |j| is the set of predictions

$P_{u,i}$ that the recommender system can make and such that $r_{u,i} \neq 1$.

While recommending the items to the users sometimes some considerations are adapted. However, these considerations are useful to Recommending the items to the users. Therefore,
TopN Recommendations
Top 10 Recommendations
Actual rating > Threshold then called as relevant item
Predicted rating >Threshold called as Recommended item

Threshold value let us consider 3.5 if the rating is between 1 to 5

Example: MAE=|2-3.75|+|3-5|+|2-4.125|/|15|
MAE=0.34615386
Precision=(Relevant*Recommended)/Recommended

VII. RESULTS AND DISCUSSION

MAE calculation for each user is represented as in the following table

Table 5. showing MAE values for different users Reliability factor:

Users	MAE
U1	0.307692
U2	0.265756
U3	0.213333
U4	0.123505
U5	0.446213
U6	0.323052
U7	0.558293
U8	0.201909
U9	0.192746

Fig.6, Table showing reliability values using Location attribute

S.NO	User	Items	Reliability factor based on Location
1	U_1	3 6 9	0.28937683 0.1604215 0.18790528
2	U_2	2	0.07740858
3	U_3	5 7 12	0.17226474 0.27057502 0.16130581
4	U_4	6	0.06902004
5	U_5	4 6 8 10	0.2371072 0.13033281 0.30239445 0.31395003
6	U_6	2 6 9	0.14438017 0.1390989 0.14781429
7	U_7	2 3 6 9	0.14090325 0.21913633 0.13589607 0.14429416
8	U_8	6 9 12	0.06719369 0.079162344 0.08712783
9	U_9	3 9	0.13435885 0.08318374

Fig.7, Table showing reliability values without using Location attribute

S.NO	User	Items	Reliability factor
1	U_1	3 6 9	0.08932004 0.05358633 0.066894956
2	U_2	2	0.050392892
3	U_3	5 7 12	0.06816235 0.09979845 0.066235915
4	U_4	6	0.057067823
5	U_5	4 6 8 10	0.05180035 0.028638411 0.04835426 0.0489838
6	U_6	2 6 9	0.091014735 0.08551569 0.11647485
7	U_7	2 3 6 9	0.08727095 0.15470108 0.092668556 0.134687

8	U_8	6 9 12	0.108101144 0.11826689 0.12627119
9	U_9	3 9	0.15331857 0.11627653

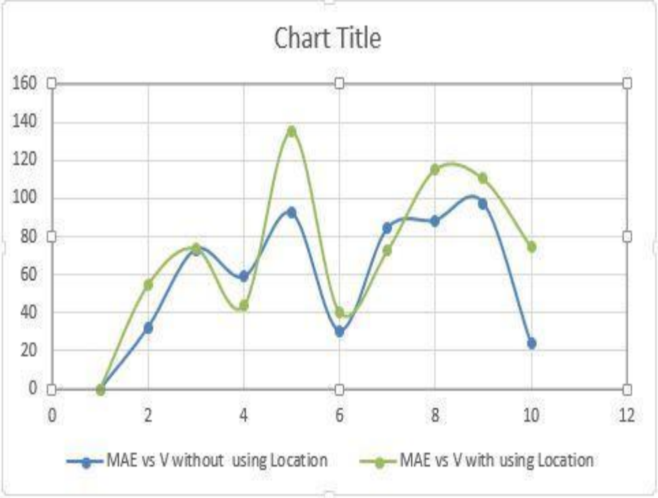


Figure 1. Graph showing MAE vs with Location and without location

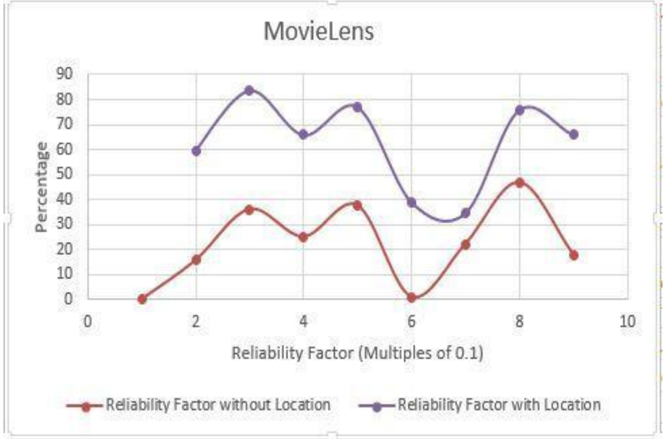


Figure.2 Graph showing Reliability factor with Location and without location.

Fig.8, Table showing Quality measure of Reliability as shown in the following table

Reliability value	Prediction error	Result of reliability	Quality of reliability
High	High	Small mistake	Small penalty
Low	High	Hit	Reward
High	Low	Small Hit	Small reward
Low	Low	Mistake	penalty

VIII. CONCLUSION AND FUTURE WORK

In this paper, we have used a demographic approach to improve the performance of the recommendation system and also mentioned an evaluation measure of reliability along with the predictions made by a recommender system. However, Demographic content is used to refine the users to display the results of a specific location. Recommending the item to the users based upon the rating which is provided by the user initially.: a prediction of how much he will rate this item; and the reliability measure of this prediction. Using these two values, users can make decisions which predictions are more reliable by considering all the pair of recommendations. We have calculated predictions based on the similarity of the user.

The measure of reliability may be adapted to a specific application by including certain additional information. Further, these recommended systems can be extended to calculate new general recommendations based on particular attributes such as timestamp, network etc.

As part of future work, we intend to explore the efficiency and performance of the algorithm on various datasets in group recommendation and also add social information, however, behavioural information it will be helpful to improve the quality of recommendation.

REFERENCES

[1] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE.

[2] S. Alonso, E. Herrera-Viedma, F. Chiclana, F. Herrera, A web based consensus support system for group decision making problems and incomplete preferences, Information Sciences 180 (23) (2010) 4477–4495.

[3] N. Antonopoulus, J. Salter, Cinema screen recommender agent: combining collaborative and content-based filtering, IEEE Intelligent Systems (2006) 35–41.

[4] A. Barragáns-Martínez, E. Costa-Montenegro, J. Burguillo, M. Rey-López, F. Mikic-Fonte, A. Peleteiro, A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition, Inform. Sci. 180 (22) (2010) 4290–4311.

[5]S. Zhang, W. Wang, J. Ford, F. Makedon, Learning from incomplete ratings using non-negative matrix factorization, in: SDM, SIAM, 2006, pp.549–553.

[6] G. Beliakov, G. Li, Improving the speed and stability of the k-nearest neighbors method, Pattern Recognition Letters 33 (2012) 1296–1301.

[7] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowledge-Based Systems 46 (2013) 109–132.

[8] P. Resnick, H.R. Varian, Recommender systems, Communications of the ACM 40 (1997) 56–58.

[9]S.AmerYahia,S.B.Roy,A.Chawlat,G.Das,C.Yu,Group recommendation : semantics and efficiency ,in: Proceedings of the 35th International Conference on Very Large DataBases,2009,pp.754–765.

[10]B.Sarwar,G.Karypis,J.Konstan,J.Riedl,Item-based collaborative filtering recommendation algorithms, in: Proceedings of the 10th International Conference on World Wide Web, WWW'01, ACM, NewYork, NY, USA, 2001, pp. 285–295.

[11] K. Ali and W. van Stam, "TiVo: Making show recommendations using a distributed collaborative filtering architecture," in *ACM KDD '04*, pp. 394–401, ACM, 2004.

[12] X. Amatriain, J. Pujol, and N. Oliver, "I like it. . . I like it not: Evaluating user ratings noise in recommender systems," in *UMAP 2009*, vol. 5535 of *LectureNotes in Computer Science*, pp. 247–258, Springer, 2009.

[13] X. Amatriain, J. M. Pujol, N. Tintarev, and N. Oliver, "Rate it again:Increasing recommendation accuracy by user re-rating," in *ACM RecSys '09*,pp. 173–180, ACM, 2009.

[14] C. De Rosa, J. Cantrell, A. Havens, J. Hawk, L. Jenkins, B. Gauder, R. Limes, D. Cellentani, and OCLC,Sharing, privacy and trust in our networked world: A report to the OCLC membership. OCLC, 2007.

[15] X. Zhou, Y. Xu, Y. Li, A. Josang, and C. Cox, "The state-of-the-art in personalized recommender systems.

Authors Profile

Kaira Nithin Goud is currently pursuing Bachelor of Technology from Gitam University, Andhra Pradesh, India. He has 6 months of industrial and 3 months of Research Experience and his main research work focuses on recommendation systems using deep learning and reinforcement learning.