ADV MATH SCI JOURNAL
Advances in Mathematics: Scientific Journal **9** (2020), no.8, 5731–5744
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.9.8.41

ENHANCED RECOMMENDER SYSTEM FOR MANAGING SPARSE DATA IN SECURED CLOUD FOR E-BUSINESS MANAGEMENT

K. INDIRA[1] AND M.K. KAVITHA DEVI

ABSTRACT. Nowadays, Cloud Computing is a compelling paradigm for all enterprises where different services such as the server, storage, and application are delivered through the internet to the organization's computer and devices. It serves computing needs for both enterprise and end-user, so it became on-demand services as all enterprises emerging to cloud computing technology. So the cloud platform is incorporated in recommender a system that is mainly used for sorting a massive amount of data to identify user's interest by rating or feedback and makes item search easier for the end-user. Therefore, the Cloud-based Recommender system applied in a variety of applications to increase sales and user satisfaction in the market. Despite advances in recommender, the system still issues like sparsity, capability, and accuracy are needed to be addressed. In this paper, the cloud-based approach addresses the serious sparse data and capability issue by constructing a trustworthy recommender system. The proposed technique increases the density of the similarity matrix and coverage for accurate prediction. At the simulation result, the accuracy of recommendation is analyzed using evaluation metrics shows that the framework is useful and better prediction achievement.

1. INTRODUCTION

Cloud computing is a big scale distributed computing with the advancement of the internet; it becomes emerging technology. Here Cloud computing called

[1]*corresponding author*

2010 *Mathematics Subject Classification.* 90B50.

Key words and phrases. Cloud Computing, Recommender system, Sparsity, User preference, Trust, Rating, Hadoop.

"cloud" since a cloud server has several configurations and can access anywhere in the world. It provides multiple services to the end-user, allowing them to use resources, accumulate and manipulate the model. Thus, it aims to provide reliable, secure, scalable services and share large datasets through the internet. Commonly a user outsources their data and process to cloud providers by paying for used services.

The characteristic of cloud computing includes:

- Resources in cloud i.e., storage, network, and process, are virtualized and achieved at a different level, including Virtual Machine(VM) and platform level. In the virtual machine level, the different web application is executed within the container on the same physical machine. Platform level enables the faultless mapping of a web application to resources provided by various cloud platforms providers.
- Resources in cloud i.e., storage, network, and process required for applications, are dynamically provisioned and varied depending upon user requirements.
- Cloud is deployed using the Service Oriented Architecture model, where all the resources are accessible over the network. The resources like software and platform is offered as a service

So the cloud is utilized for the Recommender system, which uses information and statistical methods for recommending different kinds of items to users. It is popular and used in different web applications. Recommender System is a tool that is used to offer suggestions according to the user's requirement. Recent E-Commerce systems have a million products for sale and have to predict preferences of the user and suggest items optimize sales. Recommendation systems collect preferences of user items and recommend the new item to the user, predicting by the preferences of that user for those particular products. The technique of recommender the system plays a vital role in social media and another online service.

The major contribution of this paper objective can be summarized as follows:

- Trust derivation is developed for the active user, which uses the rating of the neighbor user who has similar tastes as an active user for the particular item. By threshold value, the Jaccard and Mean Square Difference (MSD) is estimated.

- Here trust derivation is calculated for an item basis, the rating or preference of items that have purchased past by active user is analyzed and calculated the Jaccard and Mean Square Difference (MSD).
- A combination of both item trust and user trust provides a hybrid recommendation approach. The accuracy is evaluated by performance metrics called MAE, coverage.

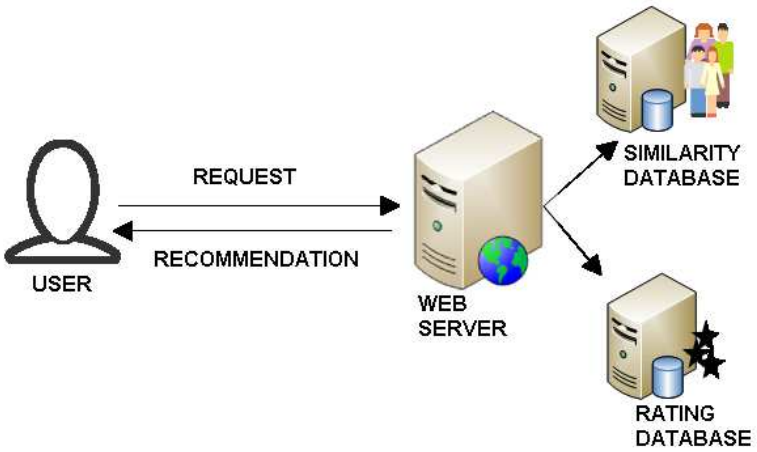

Figure 1. Recommendation system concepts

2. Related Work

Collaborative filtering

The Collaborating filtering (CF) based methods use the rating value given by the active user to various products in the application mainly to determine the neighbors of the active user and create a list of recommendations on unnoticed items based on the preference of neighbor. So the sparse data issue the related work [13] to predict unknown rating in the matrix used a technique matrix factorization method as pre-processing and [4] imputation of absent value to rating matrix by the framework of matrix factorization. By that analysis, the sparse data problem is reduced by the proposed idea called reliability-based trust-aware collaborative filtering method [13]. This method provides the trust propagation, which is the nearest neighbor by the shortest number of hops in the network path. That improves the accuracy of trust by increasing the coverage in a collaborative filtering recommendation system. To calculate the accuracy, the evaluation metrics used are User Coverage (UC), Rate coverage (RC), Mean Absolute User Error (MAUE) and Mean Absolute Error (MAE).

Content-based filtering

The content-based filtering method uses the past activities of the user to recommend the items i.e., use the content of the particular user. So it tries to recommend the product similar to the user's past liked items. There is an approach called semantic-based filtering, which uses the review provided by the user and recommends the product. The related work [17] analyzed a various set of a feature such as user expertise, sentiment, information type, and quality and it is observed that customer post the detailed comparative opinion about particular product or service for making a good decision for this recent suggestion like qualitative aspect analysis applicable to review helpfulness by [17]. So this the paper proposed four-set i.e., a concept in the review, the average number of ideas in each sentence, and review type.

Hybrid filtering

A hybrid recommender system is the one that combines one or more recommendation techniques to achieve synergy between them. Here the technique used [9] is a cross-domain hybrid system that finds the neighbor similarity of the users and items. That is beneficial to overcome sparsity problem even it has a sparse data the recommendation will be accurate by the similarity basis of neighbor. The proposed technique, called hybrid user-item trust-based recommendation (HUIT). It observes the user correlation between various user and items correlation between various items by the past preference of the active user. By that hybrid, the trust-based recommendation is derived and analyzed the metrics Mean Absolute Error (MAE) & Coverage.

Here the approaches of each paper are summarized with the performance metrics in Table 1. The various approaches in Collaborative filtering are the User-Based Algorithm (UBT), Item Based Algorithm (IBT), Dimensionalized Reduce Algorithm (DRA), Generative Model Algorithm (GMA), Spreading Activation Algorithm (SAA) and Link Analysis Algorithm (LAA). There Semantic-Based Algorithm (SBA) approach used in content-based filtering. In remaining, approaches are the Content-Aware Algorithm (CAA), Cross Domain Algorithm (CDA), and Peer-Peer Algorithm (PPA).

3. TRUST-BASED RECOMMENDATION FRAMEWORK

Through the related work, the problem addressed is data sparsity and scalability in the recommendation system. That problem is resolved in the greater approximation by the proposed idea called the Cloud-based Trustable Recommendation system. As mentioned earlier, the scalability problem always occurs when it deals with big data. Although the proposed work is completed in a cloud setup, the scalability issue already exists in a recommendation system that can also be overcome. Here the existing idea of user-level is combined with item level trust to form much more obvious performance. This collaboration of user-level and item level trust is known as a Hybrid filtering recommendation. As the dataset is large, the platform incorporates is Apache Hadoop, in which mapreduce is implemented. Hadoop is the platform that runs on the cloud infrastructure to provide distributed data mining since the rate of data is growing these days. Then K-nearest neighbor mining framework is used to address the sparsity issue is an effective way. In this model, the collected dataset through data repository is analyzed and processed.

Dataset: This dataset involves of reviews from amazon online store. The data duration is a period of 18 years that contains 35 million reviews. Reviews take account of product and user information, ratings, and a plaintext review.

Trust Computation

In trust computation, there are two level trusts building in proposed work. First is building trust by a user-user level where the users of similar taste is correlated to predict the likes and dislikes of the active user. Second is building trust by a user-items level where an earlier series of active user preferences is correlated to predict the likes of the active user. For that recommendation java, GUI is built connected with the dataset collected to predict the accurate preference of the active user. To calculate both levels of trust, the Resnick prediction technique is utilized. The result of the prediction in both the level should be in limit specified that is cross-checked by setting the threshold value. If that does not exceed the threshold value, then MSD and Jaccard are calculated. Using this user, repudiation is computed to determine the highest similarity of users on both levels. At last, both predicted trust is combined with the approach hybrid filtering where the prediction of item and user is set to be a true or false statement to determine the certain hybrid similarities in higher accuracy.

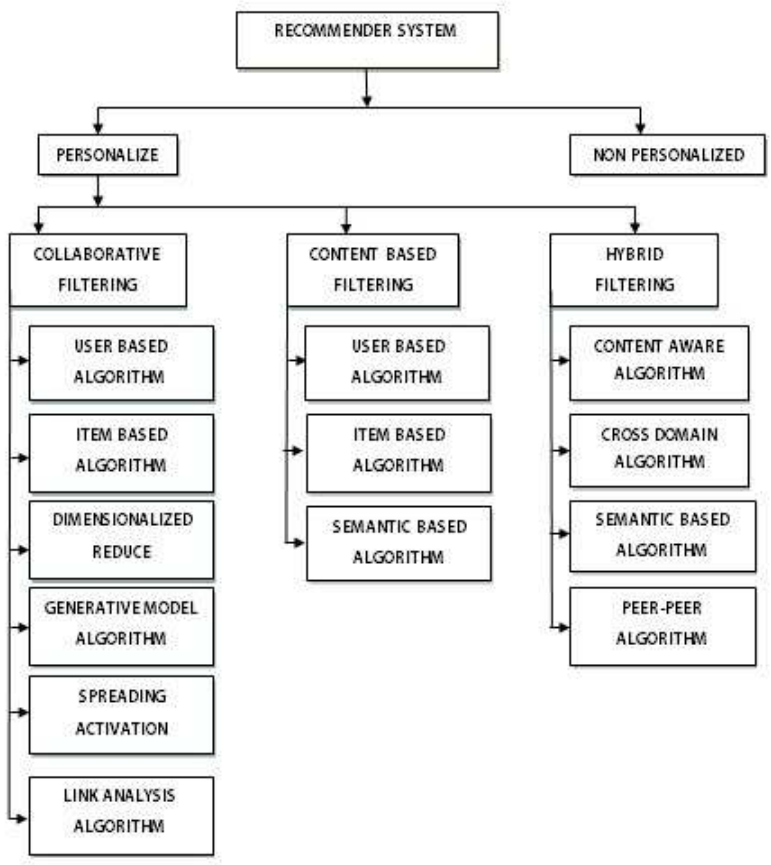

Figure 2. Recommendation system methodology taxonomy

Trust computation for User

Trust is built with a network of similar users. The algorithm checks for similar taste users with the active user to determine the item's preference of the active user. The matching user is correlated, and users within the threshold limits are used in further prediction using the MSD for rated items and Jaccard for unrated items.

Algorithm 1: Calculate the $UTrust_{U1,U2}$

Input: User U1 and U2 ratings for item I1

Output: $UTrust_{U1,U2} \in [0,1]$

1: for User U1 and U2 rated to item I1 do
2: if U1, U2$\in$ U, and I1$\in$ I then
3: get $P_{U1,I1}$
4: end if

5: if $P_{U1,I1} \in [0, 1, 2]$ then
6: get $MSD_{U1,U2}$ and $UJaccard_{U1,U2}$
7: end if
8: end for
9: get $UTrust_{U1,U2} = MSD_{U1,U2} * UJaccard_{U1,U2}$
10: return User-based Trust value between U1 and U2

Then the same function is derived for all the similar users in the network to predict accurately.

Algorithm 2: Calculate the UR_{U1}

Input: User-based trust value between U1 and U2

Output: Reputation score of $UB_{U2} \in [0, 1]$

1: for User U1 and U2 rated to item I1 do
2: if U1, U2$\in$ U, and I1$\in$ I then
3: get $UR_{U1,I1}$
4: end if
5: end for
6: return Overall Reputation score of the user U2

Trust computation for Item

The trust is built with the preferred items of the active user in the network. The algorithm checks for previous user preference taste to determine the likes of the active user to recommend. The matching items are correlated, and items within the threshold limits are used in further prediction using the MSD for rated items and Jaccard for unrated items by the particular user.

Algorithm 3: Calculate the $ITrust_{U1,U2}$

Input: User I1 and I2 ratings for item U1

Output: $ITrust_{I1,I2} \in [0, 1]$

1: for User U1 rated to item I1 and I2 do
2: if $I1, I2 \in I$ and $U1 \in U$ then
3: get $P_{U1,I1}$ 4: end if
5: if $P_{U1,I1} \in [0, 1, 2]$ then
6: get $MSD_{I1,I2}$and $UJaccard_{I1,I2}$
7: end if
8: end for

9: get $ITrust_{I1,I2} = MSD_{I1,I2} * UJaccard_{I1,I2}$
10: return Item-based Trust value between I1 and I2

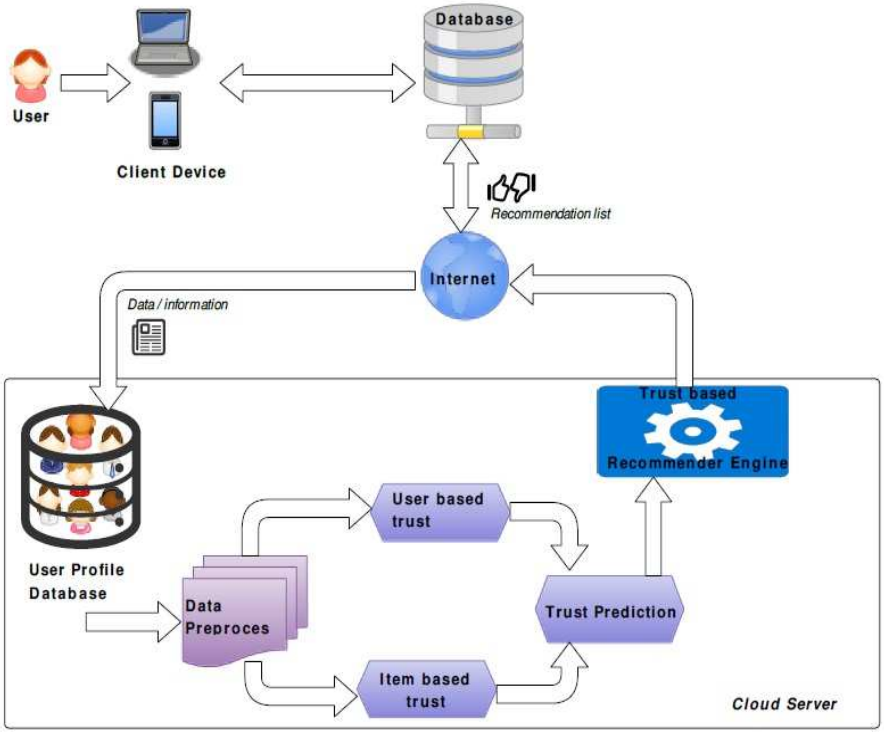

Figure 3. Trust-based Recommendation Framework

The same procedure is followed to find all preferred items by the active user in the network. It is found by reputation method i.e. called iteration for multiple preferred by the active user.

Algorithm 4: Calculate the $IR_{U1,I1}$

Input: $ITrust_{I1,I*}$

Output: Reputation score of $IR_I1 \in [0, 1]$

1: for User U1 and U2 rated to item I1 do
2: if I1,I*$\in$ U and U1$\in$ U then
3: get $IR_{U1,I1}$
4: end if
5: end for
6: return Overall Reputation score of user I1

Highest coefficient derivation for User

In the user-user level, calculate user trust and reputation of user trust is utilized for finding the higher similarity of users. If the user trust derivation for U1 and U2 is null value, then the prediction is estimated with the reputation trust of the user. Otherwise, the user trust derivation is used to estimate the higher similarity users.

Algorithm 5: Calculate the $P^{UT}_{(U1,I1)}$

Input: Rating value of U1 and U2 on item I1, $UTrust_{U1,U2}$ and UR_{U1}

Output: $P^{UT}_{(U1,I1)} \in [0,5]$

1: if $UTrust_{U1,U2} = 0$ then

2: get $P^{UT}_{(U1,I1)} = r^{-}_{U1} + \frac{\sum_{U2=1}^{N^{UT}} UR_{U2}(r_{U2,I1} - r^{-}_{U2})}{\sum_{U2=1}^{N^{UT}} UR_{U2}}$

3: else

4: get $P^{UT}_{(U1,I1)} = r^{-}_{U1} + \frac{\sum_{U2=1}^{N^{UT}} UTrust_{U1,U2}(r_{U2,I1} - r^{-}_{U2})}{\sum_{U2=1}^{N^{UT}} UTrust_{U1,U2}}$

5: end if

6: return User-based Trust Prediction

Highest coefficient derivation for Item

Here the user-item level higher similarity is calculated by the item trust and reputation of item trust. If the item trust derivation for I1 and I2 is null value, then the prediction is estimated using the reputation trust of the item. Otherwise, the item trust derivation is used to estimate the higher similarity items for the particular user.

Algorithm 6: Calculate the $P^{IT}_{(U1,I1)}$

Input: Rating value of I1 and I2 by user U1, $ITrust_{I1,I2}$ and $IR_{U1,I1}$

Output: $RP^{IT}_{(U1,I1)} \in [0,5]$

1: if $ITrust_{I1,I2} = 0$ then

2: get $P^{IT}_{(U1,I1)} = r^{-}_{I}1 + \frac{\sum_{I2=1}^{N^{IT}} IR_{U1,U2}(r_{U1,I2} - r^{-}_{I2})}{\sum_{I2=1}^{N^{IT}} IR_{U1,I2}}$

3: else

4: get $P^{IT}_{(U1,I1)} = r^{-}_{I}1 + \frac{\sum_{I2=1}^{N^{IT}} ITrust_{I1,I2}(r_{U1,I2} - r^{-}_{I2})}{\sum_{I2=1}^{N^{IT}} ITrust_{I1,I2}}$

5: end if

6: return Item-based Trust Prediction

Hybrid prediction

The hybrid prediction is to simulate the high correlated items for the active user as the preferable one. This satisfaction of accuracy produces the recommendation system as a valuable one, even though the sparse data exist. Mainly hybrid is incorporated to reduce the sparsity issue. If anyone's prediction is not accurate, another level of prediction will be close to accuracy. This leads the system to be an effective recommendation engine. If the prediction of user and item is not equal to null, then both prediction is combined to form the hybrid prediction in a more accurate preference of the active user.

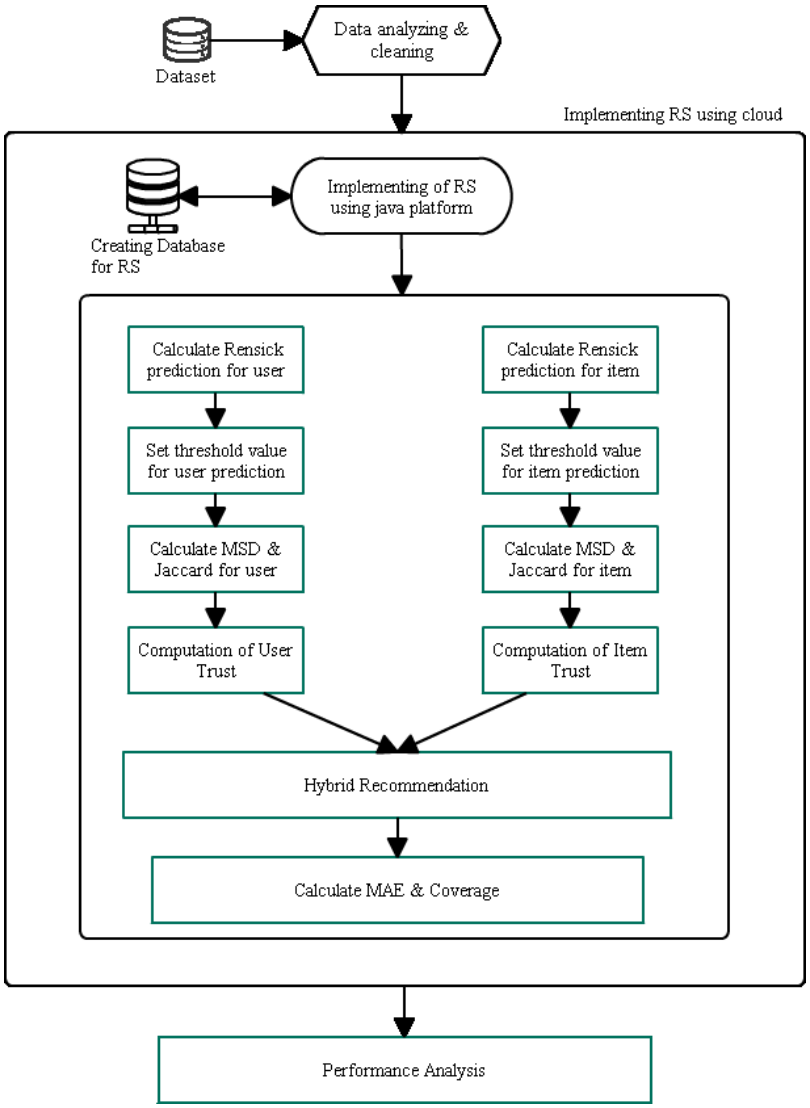

Figure 4. Trust-based Recommendation Process

Algorithm 7: Calculate the $HP_{U1,I1}$

Input: $P^{UT}_{(U1,I1)}$ and $P^{IT}_{(U1,I1)}$

Output: $HP_{U1,I1} \in [0,5]$

1: if $P^{UT}_{(U1,I1)} \neq 0 P^{IT}_{(U1,I1)} \neq 0$ then
2: get $HP_{U1,I1} = \frac{2 * P^{UT}_{(U1,I1)} * P^{IT}_{(U1,I1)}}{P^{UT}_{(U1,I1)} + P^{IT}_{(U1,I1)}}$
3: else
4: if $P^{UT}_{(U1,I1)} = 0 P^{IT}_{(U1,I1)} \neq 0$ then
5: get $HP_{U1,I1} = P^{IT}_{(U1,I1)}$
6: else
7: if $P^{UT}_{(U1,I1)} \neq 0 P^{IT}_{(U1,I1)} = 0$ then
8: get $HP_{U1,I1} = P^{UT}_{(U1,I1)}$
9: else
10: get $HP_{U1,I1} = 0$
11: end if
12: end if
13: end if
14: return Hybrid based Trust Prediction for user U1 and I1

4. Experimental results

In this section, the proposed technique performance is evaluated by the metrics. The proposed technique is implemented in the Apache Hadoop on a cloud platform. Here as we use Apache Hadoop, an open-source project to develop the scalable, reliable, and distributed process platform on the cloud infrastructure. Here the prediction of the recommendation system is evaluated by different metrics. The standard metrics used in the recommendation system is Mean Absolute Error (MAE), Mean absolute user error (MAUE), and coverage. In these metrics, MAE and MAUE are used to determine the accuracy of the proposed recommendation system. Then in coverage metrics, there are two level rate coverage and user coverage of determining the hidden similarities in the network. Predicted rating is evaluated by the iterative calculation of the difference between the actual rating and predicted rating. The fraction of difference and the average error value is defined as a mean absolute error:

$$MAE_{U1} = \frac{\sum_{I=1}^{K} r_{U1,I} - HP_{U1,I1}}{K},$$

where $r_(U1, I)$ is the actual rating, $HP_{U1,I1}$ predicted rating for user U1 on item I1 and K is the total number of ratings in the dataset:

$$MAUE = \frac{\sum_{U1 \in U} MAE_{U1}}{K_U}.$$

The mean absolute user error is determined using the calculated mean absolute error fraction with the total number of users. In which U is the users in the dataset, U1 is the active user whose preference is determined, MAE_{U1} is the found mean absolute error value for the user U1 and K_U is a number of users in the dataset. It estimates the accuracy level of the technique proposed. The other important metric is coverage to find the maximum extent of data regarding the similarity of a particular user in the dataset. Even if the dataset contains sparse data coverage will lead to an increase in the accuracy of determination to recommend

$$Coverage = \frac{HP_{U1,I1}}{K}.$$

The coverage is defined as a fraction of predicted item for the user from the total number of users in the dataset where $HP_{U1,I1}$ predicted user-item, and K is the total number of users.

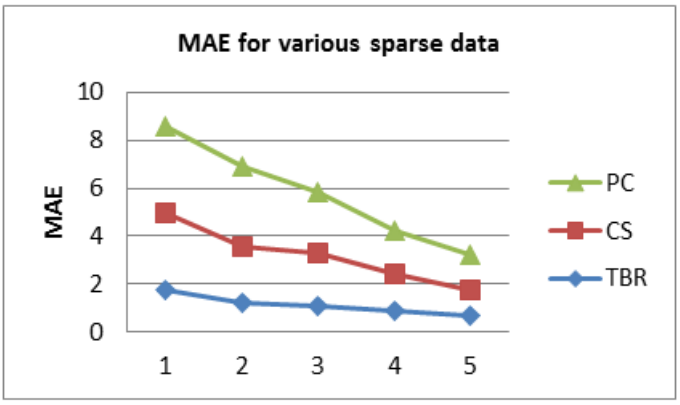

Figure5. Accuracy for different sparse data

5. CONCLUSION

In cloud computing, it is easier to simplify the large scale system. This paper, the most general limitation in recommendation system, is data sparsity, which

is necessary for the accurate recommendation, and the capability as the application uses the large scale database has been addressed. Several case studies are analysed to show the feasibility and efficiency in each model with its pros and cons in real-world applications. The proposed trust-based recommendation for the cloud solves the sparsity issue by the neighbour correlation rule and deploy in the cloud environment for performance investigation. The algorithm of each part is derived, so that hybrid prediction is calculated. Finally, the trust-based recommender system is derived hybrid prediction using MapReduce concept. The MapReduce application framework proposed is deployed on the top of cloud environments such as Eucalyptus and open nebula. The working of MapReduce becomes a better way by using the cloud environment. In the near future, the effective and quality of recommendation system in the cloud is calculated as per the proposed algorithm. Finally, the performance of the proposed trust-based recommendation system in cloud measures by any of metrics prediction metrics, Set Recommended metrics, Rank Recommended metric.

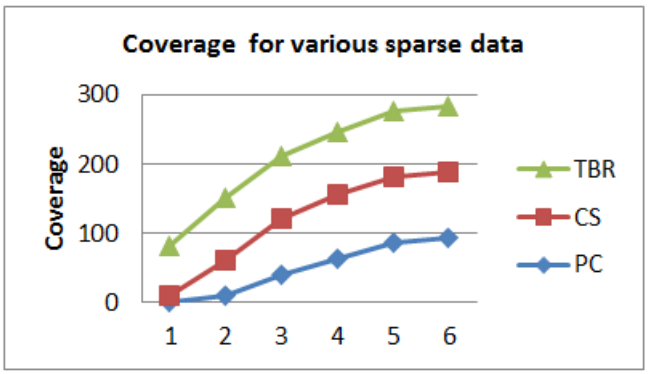

Figure6. Extensity of coverage for different sparse data

REFERENCES

[1] L. JIANG, Y. CHENG, L. YANG, ET AL.: *A trust-based collaborative filtering algorithm for E-commerce recommendation system*, J Ambient Intelligent Human Computing **10** (2019), 3023-3034.

[2] A. GUPTA, B.K. TRIPATHY: *A Generic Hybrid Recommender System based on Neural Networks*, IEEE International Advance Computing Conference, (IACC), Gurgaon, 2014, 1248-1252, doi: 10.1109/IAdCC.2014.6779506.

[3] F. DONG, J. LUO, X. ZHU, Y. WANG, J. SHEN: *A personalized hybrid recommendation system oriented to e-commerce mass data in the cloud*, IEEE International Conference on Systems, Man and Cybernetics, Manchester, 2013, 1020-1025, doi: 10.1109/SMC.2013.178.

[4] L. SHARMA, A. GERA: *A Survey of Recommendation System: Research Challenges*, International Journal of Engineering Trends and Technology (IJETT), **4**(5) (2013), 1989-1992.

[5] F.O. ISINKAYE , Y.O. FOLAJIMI , B.A. OJOKOH: *Recommendation systems: Principles, methods and evaluation*, Egyptian Informatics Journal, **16**(3) (2015), 261-273.

[6] T. ZAIN, M. ASLAM, M.R. IMRAN, A.M. MARTINEZ-ENRIQUEZ: *Cloud Service Recommender System Using Clustering*, 11th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Campeche, 2014, 1-6, doi: 10.1109/ICEEE.2014.6978334.

[7] S.S. LAKSHMI, T. ADI LAKSHMI: *Recommendation Systems: Issues and challenges*, International Journal of Computer Science and Information technologies, **5**(4) (2014), 5771-5772.

[8] S. BAGCHI: *Performance and Quality Assessment of Similarity Measures in Collaborative Filtering Using Mahout*, 2nd International Symposium on Big Data and Cloud Computing, **50** (2015), 229-234.

[9] R. YERA TOLEDO, YAILE CABALLERO MOTA , LUIS MARTINEZ: *Correcting noisy ratings in collaborative recommender systems*, Journal on Knowledge-Based Systems, **76** (2015), 96-108.

[10] P.B. THORAT, R.M. GOUDAR, S. BARVE: *Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System*, International Journal of Computer Applications, **110**(4) (2015), 31-36.

ASSISTANT PROFESSOR,
DEPARTMENT OF INFORMATION TECHNOLOGY,
THIAGARJAR COLLEGE OF ENGINEERING,
MADURAI

PROFESSOR,
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
THIAGARAJAR COLLEGE OF ENGINEERING,
MADURAI