Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CLOUD SERVICE BROKER: SELECTION OF PROVIDERS USING DTRFV EVALUATION

[1]ANUPRIYA KONERU, [2]SREELATHA M

[1]Research Scholar, Acharya Nagarjuna University, Department of Computer Science and Engineering, Guntur, India

[2]Professor, RVR & JC College of Engineering, Department of CSE, Guntur, India

E-mail: [1]smartykoneru@gmail.com, [2]lathamoturi@rediffmail.com

ABSTRACT

Multicloud environment possess new challenges to the Cloud Service Requesters. Cloud Service Requesters are not aware of the quality of service offered by Cloud Service Providers. To select the suitable provider for a service the requester meets the Cloud Service Broker. Broker finds the best suitable provider for every request. Broker identifies the previous service ratings to evaluate the Reputation Factor Value of provider. In this paper, we present Dynamic True Reputation Factor Value Evaluation algorithm to find the best providers. By using this algorithm Broker can identify the unfair ratings and reduce the effect of unfair ratings in the calculation of True Reputation Factor Value of provider. Results are presented to appraise the success of the proposed model.

Keywords: *Multicloud, Cloud service requester, Broker, Reputation, Provider*

1. INTRODUCTION

A Multi cloud environment consists of huge number of cloud service providers to serve the needs of different organizations which are scaled up. It is really a great opportunity to the Cloud Service Requesters for choosing the Cloud service Providers in this kind of environment. In this place, the Cloud Service Requesters can take different services from different Cloud Service Providers. If a particular service is available with multiple Cloud Service Providers then the selection of a single Cloud Service Provider is really a difficult task to the Cloud Service Requester. So, in the industry most of the organizations are contacting a mediator to select the best suitable Cloud Service Provider. The responsibility of the mediator is to choose the suitable Cloud Service Provider on behalf of Cloud Service Requester. In the IT Industry, the mediator is termed as a CLOUD SERVICE BROKER. The Cloud Service Providers and Cloud Service Requesters are interested to communicate through Cloud Service Broker for simplifying their work.

In the Multi Cloud Environment, the cloud service providers and cloud service requesters register with the Broker. Cloud Service Broker maintains all the services offered by each registered Cloud Service Provider and accepts the request from the Cloud service Requesters with all the required specifications. Based on the specifications of each Cloud Service Requester, Cloud Service Broker selects that particular Cloud Service Provider. The advantage of using Cloud Service Broker is to get the maximum services at minimum cost from the best suitable Cloud Service Provider to the Cloud Service Requester. This advantage really attracts the entire IT Industry towards Cloud Service Brokering. There are Number of Cloud Service Brokers in the real environment which are running profitably. In this environment, the Cloud Service Requesters have to pay the amount to the Cloud Service Broker when it takes suggestion from the Broker. The Cloud Service Broker use different mechanisms to map the requirements specified by the CSRs to the services offered by CSPs. One of the mechanism is to use different filters to select the Cloud Service Provider like cost, availability, security, QoS etc. The Cloud Service Requester specifies the cost, level of availability, level of security and also the level of quality in the requirement specification form. The Broker apply these values to the filters and find out the suitable Cloud Service Provider among the registered Providers. Another mechanism is by applying a service mapping using mining algorithm. There are different approaches to select.

This paper is mainly concentrating on the evaluation of fair and unfair ratings by Cloud Service Broker (CSB) when the Cloud Service Requester rated the Cloud Service Provider (CSP). After identifying the unfair ratings CSB reduce the weightage of those ratings to reduce the effect of unfair ratings during the CSP reputation evaluation. This could be done by using Dynamic True Reputation Factor Value Evaluation Algorithm.

This paper consists of introduction in the first section, problem statement in the second section, survey in the third section, methodology is explained in the fourth section, experimental work in the fifth section, Results and Discussions along with two case studies in the sixth section, and concluded with implications of research in the seventh section.

2. PROBLEM STATEMENT

In the previous work [15], Reputation Factor Values (RFVs) are calculated based on the response form submitted by the Cloud Service Requester. But in the real environment the most of the Cloud Service Requesters are no longer interested to fill the lengthy response form. Some of the Cloud Service Requesters interested to fill response form when they were not satisfied by that Cloud Service Provider Service. To reduce the complexity of filling lengthy response forms, in this paper consider star ratings which are very popular in most of the e-commerce applications. In this work the complete Cloud Service Provider selection depends on the feedback given by the used CSR. If the Cloud Service Requester is not fairly rated the CSP then that impacts the total Broker system. So, it shows the necessity of evaluating the fair and unfair ratings given by the CSRs.

3. LITERATURE SURVEY

3.1 Survey On Multicloud Adoption

To improve the cloud availability most of the organizations migrating from single to multicloud environment. Many of the researchers worked on this to postulate the effectiveness of this migration. In [1] this paper, author specified a novel scheme to store data among multiple clouds without using any encryption scheme. It works on the basis of log range interpolation algorithm. When they compared this with existing methods, it has given good performance through availability, confidentiality and lock-in issues. In the paper [2], author raised an issue regarding the complexity in deploying multicloud applications and proposed CYCLONE architecture. This method merged number of existing solutions to provide multicloud application for research organizations. In [3], author proposed a model called multicloud searchable encryption which enables the individual providers to collaborate to search in their own data. The process is carried out at three levels: key generation, keyword identification, and obfuscation. In this the multicloud architecture searches over obfuscated keyword. In [4], author explained the way how this multicloud environment provides data privacy and availability. As per this, data is portioned into slices and each provider does not have all the data slices to see the content. This way of utilizing multicloud really improves the privacy of data. Along with privacy author introduced data duplication to improve the availability also. Many researchers [5, 6, 7], worked towards this at different applications.

3.2 Survey on Broker

In [8], author specified the significance of Broker, role of Broker, and the services provided by the Broker. In this work the Brokers are classified based on the service they provide towards the user and the provider. In [9], author specified the Broker services. As per this work, Broker reserves the services from different providers and lease their services to the users at cost benefit. So, this model motivates the organizations to take services from providers only through Broker to get at profitable cost. In [10] this work, Broker could show the benefit to the user when they take services through Broker. In this Broker guides the users to get volume discount pricing offered by providers using cost efficient online scheduling algorithm.

3.3 Survey on Trust Model

Trust is an important factor which is needed in between the providers and users. The factors that impact the trust are scalability, availability, security, and usability. In [11] this paper, author proposed a fuzzy based trust model which evaluates the performance of providers. To describe this feature IaaS based e-learning system is considered. The main important steps in this model are defining the fuzzy sets to all possible factors and also for the output which is the trust value of the provider, fuzzification process, designing the rules based on the neural network, use of Sugeno's method, and defuzzification for generating a single output value. In [12] this paper, author proposed a method for CSR to identify trustworthy cloud providers. It deals with different kinds of threads against the system and methods to handle these threads. [13], proposed a method called RLM for reputation evaluation. This reputation value is mainly depending on the feedbacks. To identify the motivated feedbacks author used EM algorithm. [14], describes the threads that compromises the reliability of the trust system. In the previous work, RFV [15] is evaluated for every CSP and selected the CSP which has high RFV. But in RFV evaluation, the CSB is not evaluating the trust worthiness of the ratings. It effects the future business of CSP.

4. METHODOLOGY

Cloud Service Broker is evaluating the performance of Cloud Service Provider using Cloud Service Requesters response. If the response given by the CSR is not fair and motivated by any of the other sources then the entire Broker system will be collapsed and the future business of that Cloud Service Provider will be get affected. So, the Cloud Service Broker should maintain a mechanism to detect the unfair and fair raters. This mechanism is termed as Dynamic True Reputation Factor Value Evaluation (DTRFVE).

The Cloud Service Broker architecture consists of a local repository with list of registered Cloud Service Providers and list of Cloud Service Requesters. This repository is maintained by the Broker by assigning the local unique identity numbers to all the providers and requesters. The purpose of these IDs is to recognize the individual providers and requesters and also to hide their original identity which may intern overcome the Sybil attacks. The internal components of the Cloud Service Broker are discussed in [15].

4.1 Dynamic True Reputation Factor Value Evaluation

The Dynamic Trust Reputation Factor Value Evaluation algorithm takes list of Cloud Service Providers and their star ratings as an Input. All the ratings are normalized using min-max normalization [16] method. Normalized values are stored for further process in the dataset which is named as Norm_Data. Norm_Data is sorted in ascending order using an order ( ) function and stored as Order_Data. Find the first quartile [17] and third quartile of the Order_Data and store these values in Quartile_1, Quartile_3. These values are used to calculate the Inter Quartile Range (IQR).

IQR = Quartile_3 - Quartile_1

Find the upper and lower boundary values using IQR.

Lower_Quartile = Quartile_1 – 0.5*IQR

Upper_Quartile = Quartile_3 + 0.5*IQR

Now, every rating against each Cloud Service Provider in Order_Data is considered and compared their range.

If the rating falls below the Lower_Quartile or above the Upper_Quartile then consider it is as unfair and assign the very low weightage score to its rating.

If the rating falls above the Lower_Quartile and below the Quartile_1 then consider it is as fair and assign the average weightage score to its rating.

If the rating falls below the Upper_Quartile and above the Quartile_3 then consider it is as fair and assign the average weightage score to its rating.

If the rating falls in between the Quartile_1 and the Quartile_3 then consider it is as fair and assign the high weightage score to its rating.

After assigning weightage scores to their ratings find the mean of all these weighted ratings for every provider in the data set.

$$TRFV = \sum_{i=1}^{n} (Ws * Rating\, i) / n$$

Calculate the True Reputation Factor Value of Cloud service Provider by using the above equation. In that Ws indicates the Weightage Score, Rating is the normalized rating given by the Cloud Service Requester, n is the number of Ratings. This process gives the True Reputation Factor Value of every Cloud Service Provider.

When the CSR gives the feedback immediately the CSB find out the Divergence Value of that rating.

Divergence = Normalized (New_rating) – True RFV of CSP

If the Divergence is less than a Threshold value, then that rating will be added to the Data set.

Divergence <= 0.5 then True Rating

Divergence > 0.5 then False Rating

Otherwise, CSB allots minimum rating to the CSP by specifying that as the CSR rating as unfair.

4.2 Algorithm

Import the data
Replace non-numeric cells with NaN
Create output variable
Create dataset array
Allocate imported array to column variable names
Clear temporary variables
Take size of data in to Row_Count and Coloumn_Count
Loop 1: Repeat 1to Coloumn_Count

Loop 2: Repeat 1 to Row_Count
Norm_Data = Normalization (Data)
Order_Data = Sort (Norm_Data)
Quartile_1 = quantile(Order_Data,0.25)
Quartile_3 = quantile(Order_Data,0.75)
IQR = Quartile_3 - Quartile_1
Lower_Quartile = Quartile_1 – 0.5*IQR
Upper_Quartile = Quartile_3 + 0.5*IQR
Unfair [ ] = Order_Data ((Order_Data < Lower_Quartile) | (Order_Data > Upper_Quartile))
Weighted_Score = 0.2
Unfair [ ] = Unfair [ ]* Weighted_Score
Fair[] = Order_Data (Order_Data < Quartile_1 & Order_Data > Lower_Quartile)| Order_Data (Order_Data > Quartile_3 & Order_Data < Upper_Quartile)
Weighted_Score = 0.5
Fair [ ] = Fair [ ]* Weighted_Score
Fairest[]=Order_Data (Order_Data >= Quartile_1 & Order_Data <= Quartile_3)
Weighted_Score = 0.8
Fairest [ ] = Fairest [ ]*Weighted_Score
End Loop 2
True RFV= Average of Unfair [ ], Fair [ ], and Fairest [ ]
End Loop 1

5. EXPERIMENTAL WORK

In this section, we investigate the effectiveness of our proposed method DTRFVE Initially this is simulated using Java. We compare the performance of DTRFVE with RFV [15]. This investigation is done by applying our algorithm in the simulated cloud application. The following figure 1 shows the flow of work. It consists of the data collection, cleaning the data, Normalization of data and apply the Dynamic True Reputation Factor Evaluation Algorithm.

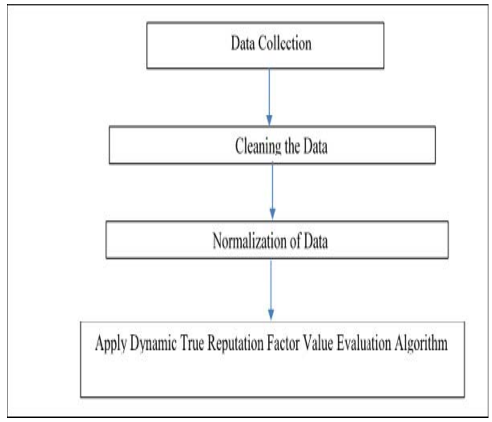

Figure 1: Work Flow

5.1 Data Collection

Dataset named Cloud Service Providers is used in experiment presented here. Table 1 shows the list of Cloud Service Providers considered in the Dataset. This data is collected from www.g2crowd.com. This dataset consists of ten Cloud Service Providers in the Market.

Table 1: List of Providers With Their Ids

IDs of the CSP in the Dataset	Name of the Cloud Service Provider
1	AmazonVPC
2	Digital Ocean
3	Amazon S3
4	Softlayer Virtual Servers
5	Linode
6	Cloud stack
7	Microsoft Azure
8	Compute Engine
9	Open Stack
10	Rackspace Managed Cloud

The initial star ratings of the Cloud Service Provider are shown in figure 2. In the dataset, the ratings of each provider varies based on the number of Cloud Service Requesters who rated the Providers after taking the service.

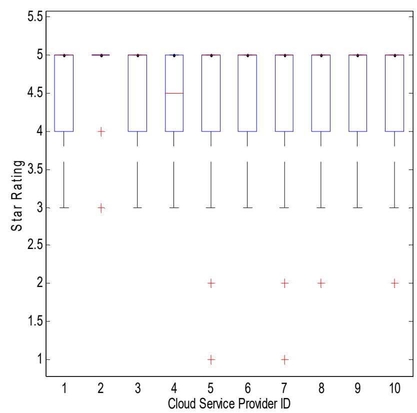

Figure 2: Cloud Service Provider 1-5 Ratings

In figure 3 the X-axis shows the Ids of Cloud Service Providers and the Y-axis shows number of ratings for each of the Cloud Service Provider.

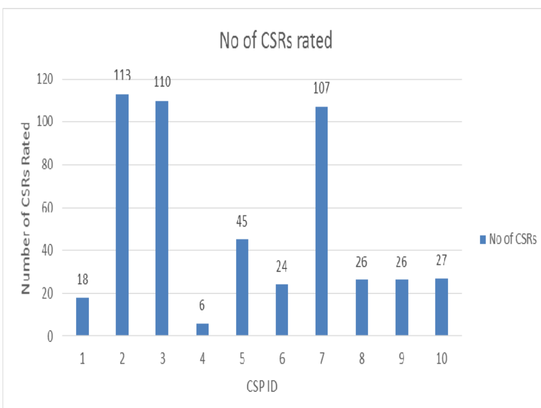

Figure 3: Number Of Ratings For Each CSP

5.2 Cleaning

Cleaning is the process making the data ready for the task. In this dataset all the Cloud Service Providers are not having the same number of Cloud service Requesters. The first row consists of the Providers names. So, to make it ready for the task we apply replacement of non-numeric elements with NaN.

5.3 Normalization

The ratings in the Dataset are 1-5 star ratings. These are normalized in the range of 0-1. When we multiplied these range of ratings with the weighted score in some cases where rating is 0 the product will become 0. To avoid this situation, we are normalizing the ratings in between 0.1 - 0.9. Here we used min-max Normalization [16] method.

5.4 Sorting

The ratings which are normalized in the range of 0.1 - 0.9 are need to be sorted in ascending order to apply the Dynamic Reputation Factor Value Evaluation Algorithm.

5.5 Dynamic True Reputation Factor Value Evaluation

Apply the DTRFV algorithm on the data set to find the TRFV of ten Cloud Service Providers in the data set.

6. RESULTS & DISCUSSIONS

Cloud Service Broker received a request from Cloud Service Requester for IaaS. Broker search for the Providers who offer IaaS in its local repository. The result of this search has given ten providers who offer this service. Now Broker has to select one best Provider among ten. Broker applied the Dynamic True Reputation Factor Value Evaluation Algorithm on the Cloud Service Provider Data set which has those ten providers with ratings.

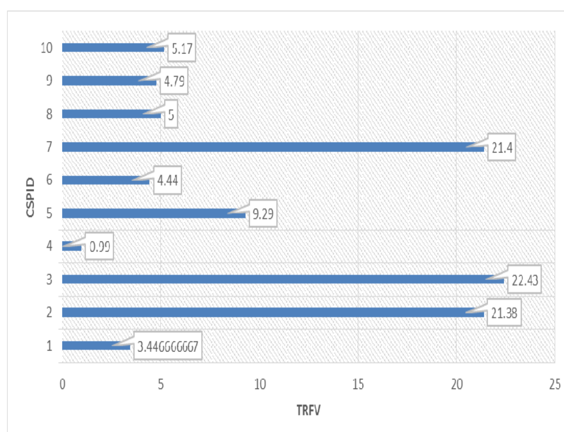

Figure 4: TRFV of CSPs

The above figure shows the True Reputation Factor Values of all the ten providers in the Dataset using Dynamic True Reputation Factor Value Evaluation Algorithm. In this Graph X-axis shows the TRFV of CSPs and Y-axis shows the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CSP Ids. Now the Broker decides that CSP ID 3 as a suitable provider to serve the present request.

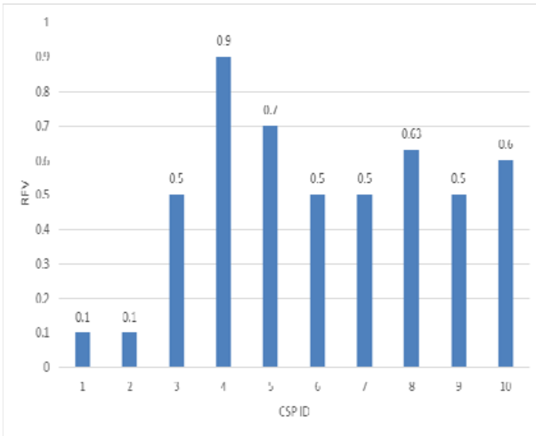

Figure 5: RFV of Cloud Service Providers

The above figure shows the Reputation Factor Values of ten Cloud Service Providers. As per the previous model [15] the Broker selected the CSP ID 4 as a service provider to serve the request. In this method, equal priority has been given to each rating. The presence of unfair ratings effects the complete Broker model and leads to effect the business of the Cloud Service Providers. This is the main limitation of RFV.

As per the above observation, the number of ratings effects the True Reputation Factor Value. If the Number of ratings are more that Cloud Service Provider is getting maximum TRFV. Then it creates load imbalance problem. To solve it, we consider only the past 'n' transactions. This 'n' value is decided by the Cloud Service Broker. For example the CSB decided n=7, then past seven ratings of each provider is considered and apply the Dynamic True Reputation Factor Value Evaluation Algorithm. The accuracy of this algorithm is helpful to the Cloud Service Broker to improve Brokering Service among the Cloud Service Requester and Cloud Service Provider. The effect of number of ratings is shown in figure 6 when we consider Cloud Provider Dataset.

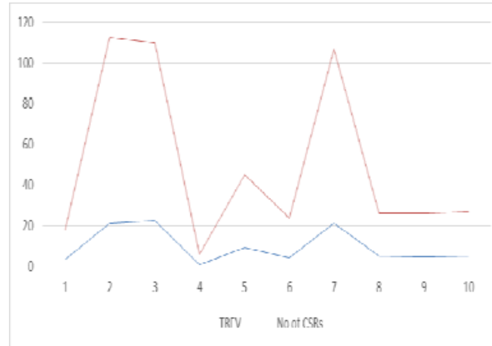

Figure 6: Effect of Number of Ratings on CSP Dataset

6.1 Case Study I

To understand the usage of Dynamic Reputation Factor Value Evaluation Algorithm we consider a simple situation where there are 10 normalized ratings for a Cloud Service Provider1 and 15 normalized ratings for a Cloud Service Provider2. Their ratings are as follows.

Rating [ ] = [0.8 0.6 0.5 0.5 0.4 0.4 0.3]

As per our algorithm the sorted array of Ratings

Rating [ ] = [0.3 0.4 0.4 0.5 0.5 0.6 0.8]

First quartile range Q1 = 0.4

Third quartile range Q3= 0.6

IQR = 0.2

Lower Limit= 0.3

Upper Limit =0.7

Consider all ratings which falls under the range of above Upper Limit and below Lower Limit. Those are unfair ratings. In this case only one ratings falls under this category. Weighted Score is 0.2.

$$0.8*0.2 = 0.16 \quad (1)$$

Consider all ratings which falls under the range of Q3 and Upper Limit or Q1 and Lower Limit. Those are fair ratings. In this case only one ratings falls under this category. Weighted Score is 0.5.

$$0.3*0.5 = 0.15 \quad (2)$$

Consider all ratings which falls under the range of Q1 and Q3. Those are also fair ratings. In this case five ratings fall under this category. Weighted Score is 0.8.

$$0.6*0.8 = 0.48 \quad (3)$$
$$0.5*0.8 = 0.40 \quad (4)$$
$$0.5*0.8 = 0.40 \quad (5)$$
$$0.4*0.8 = 0.32 \quad (6)$$
$$0.4*0.8 = 0.32 \quad (7)$$

TRFV of the provider = (Eq1+ Eq2+ Eq3+ Eq4+ Eq5+ Eq6+ Eq7)/n

TRFV = (0.16+0.15+0.48+0.40+0.40+0.32+0.32)/7 = 0.31

Like this the Cloud Service Broker evaluates the True RFV of all Cloud Service Providers.

Whenever the Cloud Service Broker receives a request for a Cloud Service Provider from the Cloud Service Requester immediately it search in its local repository for its service match. If there are multiple providers who could offer the same required services then CSB choose the one which has maximum TRFV. That Cloud Service Provider will serve the request. After taking the service from the Provider the Cloud Service Requester rate the service by giving a Star rating. This is not directly included in the local repository of the CSB. The CSB finds the Divergence value of this rating by subtracting the normalized value of this rating from the TRFV of that CSP. If the Divergence is less than the 0.5 then consider the rating as valid and insert it in the repository. Whenever it is needed CSB finds the TRFV of all the required providers. For example the new normalized rating is 0.6. Then Divergence = 0.3 which is less than 0.5. So, this rating is added to the dataset because it is Valid.

6.2 Case Study II

Consider two Cloud Service Providers and assigned Identity Numbers are as CSP 1 and CSP 2. Ratings are as follows:

CSP 1: 5 3 2 1 3 5 5 2 1 3 3

CSP 2: 1 5 3 2 4 2 4 4 3 3 3 2 3 4 2 5 5

The CSP 1 has 11 Ratings whereas CSP 2 has 17 Ratings. These are normalized to 0.1-1. After applying Min- Max Normalization:

CSP 1: 1 0.5 0.3 0.1 0.5 1 1 0.3 0.1 0.5 0.5

CSP 2: 0.1 1 0.5 0.3 0.7 0.3 0.7 0.7 0.5 0.5 0.5 0.3 0.5 0.7 0.3 1 1

The sorted list of ratings are

CSP 1:0.1 0.1 0.3 0.3 0.5 0.5 0.5 0.5 1 1 1

CSP 2: 0.1 0.3 0.3 0.3 0.3 0.5 0.5 0.5 0.5 0.5 0.7 0.7 0.7 0.7 1 1 1

The RFV values of CSP 1 and CSP 2 are

RFV (CSP1) = 0.549

RFV (CSP2) = 0.60

Apply Dynamic True Reputation Factor Value Evaluation Algorithm on CSP 1 and CSP2

TRFV (CSP 1) = 0.416

TRFV (CSP 2) = 0.344

By considering RFV values the Broker would select the CSP 2 which has high RFV value. But when the Broker uses the TRFV of Cloud Service Providers, selects CSP 1.

Table 2 shows the RFV and TRFV of CSP 1 and CSP 2. In this table, initially we consider 11 ratings of CSP 1 and 17 ratings of CSP 2.

Table 2: Number Of Ratings For Each CSP

CSPID	No of Ratings	RFV	TRFV
1	11	0.549	0.416
2	17	0.6	0.344
1	5	0.48	0.25
2	5	0.52	0.33

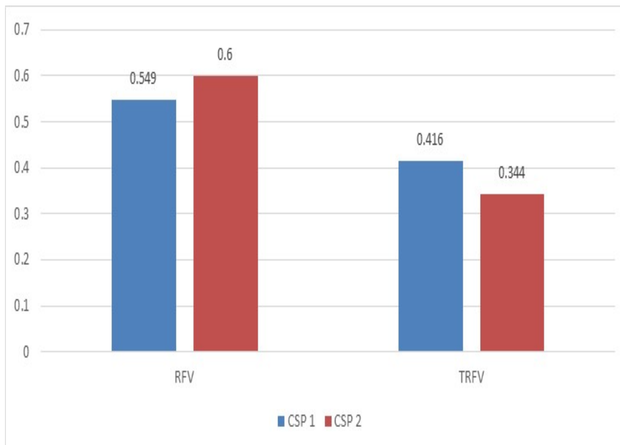

Figure 7: RFV and TRFV of CSPs

The above figure shows the RFV and TRFV of CSP 1 and CSP2. When we consider only first 5 ratings of CSP1 and 17 ratings of CSP2 then TRFV (CSP1) is 0.25 and TRFV (CSP2) is 0.344. Then Broker selects CSP2 which has high TRFV value. Figure 8 shows these values. Based on the above result we could say if the number of ratings are more that CSPs TRFV value would be high. To overcome this problem we consider equal number of past 'n' ratings from each provider.

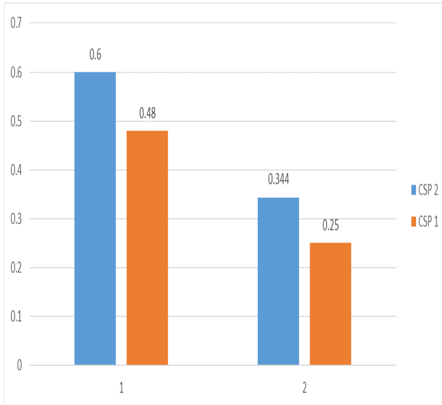

Figure 8 : Effect Of Number Of Ratings

In this case consider n value as 5.

CSP 1: 5 3 2 1 3

CSP 2: 1 5 3 2 4

TRFV (CSP 1) = 0.25

TRFV (CSP 2) = 0.33

The Broker selects CSP 2. Figure 9 shows the RFV and TRFV of CSP 1, CSP 2 by considering past five ratings in both the providers.

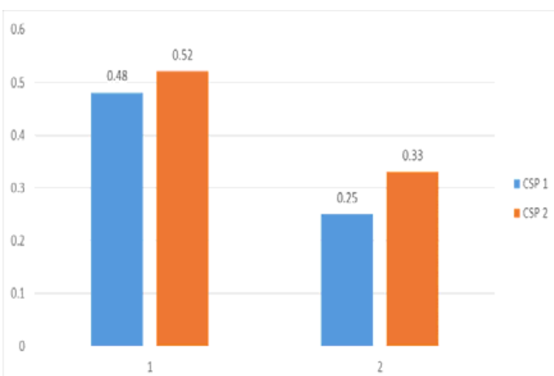

***Figure 9 :** RFV and TRFV of Cloud Service Providers when n = 5*

7. CONCLUSIONS

This paper defines the effect of false reputation in the Cloud Service Broker system where the Cloud Service Requesters depend on the decision of Cloud Service Broker in the Multicloud environment. To identify the false ratings, we proposed an algorithm called Dynamic True Reputation Factor Value Evaluation. This algorithm helps to reduce the effect of unfair ratings by assigning low weightage to those ratings and encourage the fair ratings by giving high weightage in the TRFV evaluation process. We applied this algorithm on the Cloud Service Provider rating data set to show the efficiency of this algorithm. We compared these results with RFV and identified the effectiveness of DTRFV evaluation algorithm by reducing the effect of unfair ratings over RFV.

We identified, however, that ratings given by the Cloud Service Requester indicates the degree of their satisfaction with the service they received which really implies to that service provider also. In future we plan to improve the Broker services by taking necessary actions on the Providers who got poor TRFV and any SLA violations.

REFRENCES:

[1] Jin Y, Wang Y, Xia W, Deng L, He H. A Data Hiding Scheme Based on Lagrange Interpolation Algorithm and Multi-clouds. In: Seventh International Symposium on Parallel Architectures, Algorithms and Programming (PAAP); 2015; doi:10.1109/paap.2015.47. pp. 210-216.

[2] Slawik M, Zilci BI, Demchenko Y, Baranda J, Branchat R, Loomis C, Lodygensky O, Blanchet C. CYCLONE Unified Deployment and Management of Federated, Multi-cloud Applications. In: Utility and Cloud Computing (UCC) 2015 IEEE/ACM 8th International Conference; 7-10 Dec. 2015; doi: 10.1109/UCC.2015.81, Limassol. pp. 453-457.

[3] David S.L.W, Rajasekaran S, Cheng Z, Naik K, Kuo SY. Efficient Selection and Sorting Schemes Using Coteries for Processing Large Distributed Files. J PARALLEL DISTR COM 2002; ISSN 07437315, vol. 62, pp. 1295-1313.

[4] Peng Xu, Xiaqi Liu, Zhenguo Sheng, Xuan Shan, Kai Shuang. SSDS-MC: Slice-based Secure Data Storage in MultiCloud Environment. In: 11th EAI International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QSHINE 2015); 19-20 Aug. 2015; doi:10.410S/eai.19-S-2015.2260679, pp. 304-309.

[5] Li J, Lin D, Squicciarini A, Li J, Jia C. Towards Privacy-Preserving Storage and Retrieval in Multiple Clouds. In: IEEE Transactions on Cloud Computing; 01 October 2015; vol.PP, Issue.99, doi: 10.1109/ TCC.2015.2485214, pp.1-11.

[6] Hussain SK, Sreenivasulu. An efficient and economical multi-cloud storage in cloud computing. In: IET Chennai 3rd International on Sustainable Energy and Intelligent Systems (SEISCON 2012); 2012, Tiruchengode, doi: 10.1049/cp.2012.2212, pp. 1-7.

[7] Alsolami F, Boult TE. CloudStash: Using Secret-Sharing Scheme to Secure Data, Not Keys, in Multi-clouds. In: 2014 11th International Conference on Information Technology; 2014; New Generations, Las Vegas, NV, doi: 10.1109/ITNG.2014.119, pp. 315-320.

[8] Wadhwa B, Jaitly A, Suri B. Cloud Service Brokers: An Emerging Trend in Cloud Adoption and Migration. In: 2013 20th Asia-Pacific Software Engineering Conference (APSEC); 2013; Bangkok, doi: 10.1109/APSEC.2013.129, pp. 140-145.

[9]Wang W, Niu D, Liang B, Li B. Dynamic Cloud Instance Acquisition via IaaS Cloud Brokerage. IEEE T PARALL DISTR; 1 June 2015; vol. 26, no. 6, doi: 10.1109/TPDS.2014.2326409, pp. 1580-1593.

[10] Zhang R, Wu K, Li M, Wang J. Online Resource Scheduling Under Concave Pricing for Cloud Computing. IEEE T PARALL DISTR; April 1 2016; vol. 27, no. 4, doi: 10.1109/TPDS.2015.2432799, pp. 1131-1145.

[11] Alhamad M, Dillon T, Chang E. A Trust-Evaluation Metric for Cloud applications. International Journal of Machine Learning and Computing; 2011; vol. 1, no. 4, pp. 416-421.

[12] Chong S, Abawajy J, Ahmad M, Hamid I. Enhancing Trust Management in Cloud Environment. Procedia Soc Behav Sci; 2014; vol. 129, pp. 314-321.

[13] Wang X, Liu L, Su J, RLM: A General Model for Trust Representation and Aggregation. IEEE Trans Services Comput; Jan.-March 2012; doi: 10.1109/TSC.2010.56, vol. 5, no. 1, pp. 131-143.

[14] Chong S, Abawajy J, Hamid I, Ahmad M. A Multilevel Trust Management Framework for Service Oriented Environment. Procedia Soc Behav Sci; 2014; vol. 129, pp. 396-405.

[15] Koneru A, Latha MS. Three tier architecture: To select CSP through CloudServiceBroker in multicloud environment. In: 2016 International Conference on Inventive Computation Technologies (ICICT); 2016; Coimbatore, doi: 10.1109/INVENTIVE.2016.7823236, pp. 1-6.

[16] Chmielnicki W, Stąpor K. Investigation of Normalization Techniques and Their Impact on a Recognition Rate in Handwritten Numeral Recognition. Schedae Informaticae; 2010; vol. 19, no. -1, pp. 53-77.

[17] Benjamini Y. Opening the Box of a Boxplot. The American Statistician; 1988; vol. 42, no. 4, pp. 257-262.