Journal of Computer Science and Applications.
ISSN 2231-1270 Volume 6, Number 2 (2014), pp. 99-105

http://www.irphouse.com

Ranking of Cloud Services Using Cumulative Sum Method

Syed Riyazul Haq[1] and S.H Shabbeer Basha[2]

M.Tech, CSE Department, Madanapalle Institute of Technology & Science, JNTUA, Madanapalle, AP, Iindia. riyazulhaq.syed@gmail.com
Asst. Professor, CSE Department, Madanapalle Institute of Technology & Science, JNTUA, Madanapalle, AP, Iindia. shabbeermtech@gmail.com

Abstract

Cloud Computing is Internet-based computing, it had accepted now-a-days around the globe, due to the enormous diversity in the available services. It has become difficult to find out which service user may choose from the available services. Quality of Service (QOS) rankings by which one can choose whose services they should use. There are many services existing in the cloud, different services have different level of qualities based on their capabilities. QOS gives ranking based on the quality of service, from which we can make optimal selection. For making optimal selection of cloud service from a set of services, QOS gives valuable information. It is very expensive and time consuming for invoking real-world services, so we need to look at different techniques to find QOS values. With the help of past usage experiences of the consumers, QOS ranking prediction can be done for cloud services. But there is a chance of malicious QOS ratings which are included in feedback series. To overcome from this problem, we proposed a method called cumulative sum method. By using this method we can detect malicious QOS ratings.

Keywords: QOS; malicious.

1. INTRODUCTION

Cloud computing is a new prototype for delivering on-demand resources (e.g., infrastructure, platform, software) for the customers. The cloud removes the need for you to be in the same physical location as the hardware that stores your data. There are number of services in the cloud which are similar in their functions. Due to changeable internet connections different cloud applications users may receive different levels of quality for same cloud service. Hence, there is a need to select the best possible services. Now a days we saw a great improvement in cloud services

offered by different cloud providers, it leads to great confusion in the minds of the users, which service they may choose. There are different cloud provider existing in the market they are providing similar kinds of services. One provider may provide cheaper infrastructure services but platform service provided by him is expensive. It become a great challenge to the user to select a service which full fill his QOS requirements, it also affect the success of the business. The main challenge is how to rank the Cloud services based on these QOS values. Normally users have two kinds of requirements functional and non functional. Some of the QOS Properties can't be measured easily like user experience and security. Here, we were facing Decision making problem, which functional and non functional requirements matches the best. Due to the changing nature of cloud services which result on-demand service provision, there is a substantial variations in the Quality of Service (QOS) levels of each service. QOS is defined as a collection of properties like response time, throughput, availability etc. Among these QOS properties, values of some properties like response time, user-observed availability, essential to be measured at the client side. It is difficult to get such quality of services information from service providers, since these QOS values are depends on Internet environment. Therefore, different users may observes a quite variable QOS values for the same cloud service. Optimal Service Selection from the set of services is difficult. It is impractical for end users to obtain quality of services information by examine all services by themselves, it is also very difficult to invoke real world services because it takes much more time and expensive.

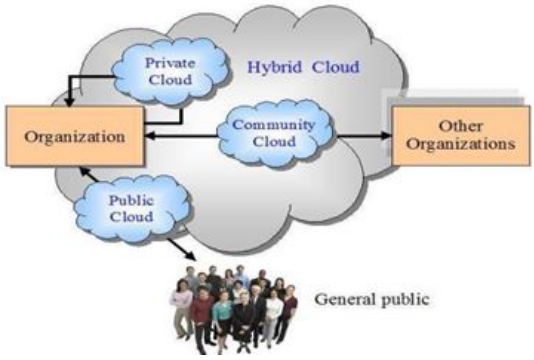


Fig.1. Cloud computing Example

The rest of this paper is organized as follows 2 related work, 3 proposed system, 4 system architecture, 5 proposed methodology, 6 conclusions and future enhancements and references.

2. RELATED WORK

Quality-of-Service (QOS) is normally used to determine the performance of the web services. Users in different geographical locations collaborate with each other to

evaluate the target Web services and share their observed Quality of services values information.

In this paper [2], it will examine the accompanying two-stage methodology to figuring out how to request. In stage one, it take in a inclination capacity, a two-contention capacity Pref(u; v) which gives back a numerical measure of how certain it is that u should be positioned before v. In stage two, it utilizes the educated inclination capacity to request a set of new occasions X; to finish this, it assess the scholarly capacity Pref(u; v) on all sets of cases u; v □ X, and pick a requesting of X that concurs, however much as could reasonably be expected, with these pair savvy inclination judgments. For stage one, it portray a particular calculation for taking in an inclination capacity from a set of positioning masters". The calculation is an on-line weight distribution calculation, much like the weighted larger part calculation (Littlestone & Warmuth, 1994) and Winnow (Littlestone, 1988), and, all the more straightforwardly, Freund and Schapire's (1997) Hedge" calculation. For stage two, it demonstrates that discovering an aggregate request that concurs best with such an inclination capacity is NP-complete. All things considered, it demonstrates that there are proficient eager calculations that dependably discover a decent estimate to the best requesting. This paper then present some trial brings about which these calculations are utilized to consolidate the aftereffects of a few inquiry masters," each of which is an area particular question development method for a web search engine.

Ardagna and Pernici [3] proposed a modeling approach to service selection problem, but trust is just one of those considered quality criteria. Although this approach is effective to select a service for a consumer, it did not focus on the detection of malicious consumers. Then if malicious consumers exist, it may not select appropriate services for consumers. In the best Service Selection proposed Exact and approximated algorithms for best service selection based on service requests made by the users or services which matches user QOS requirement. It leads to computational complexity problem. The Best service selection process can be solved by applying greedy approach. This algorithm seems to be much faster when compare to other process.

Ranking of cloud services provides important information for making best service selection from the set of similar services. To obtain QOS values information from the web services we need to invoke the services. This kind of invocation is impractical, because some time services may be open source but executing these services are expensive. In the present system we have problem with malicious QOS values. In the present system we are facing the problem with malicious QOS ratings. As we are going to provide ranking of the services based on QOS values provided by the users and by monitoring the service. It leads to inaccurate ranking, By detecting and discarding malicious ratings we can overcome from this problem.

3. PROPOSED WORK

Existing system try to predict the effective ranking in the services but fails in it just follow the recommender system and also follow the rating-oriented approach. So it does not provide that much accurate ranking prediction in the services. It is very time

consuming and resource consuming for the larger cloud services in the cloud server. In order to overcome the disadvantages in the previously proposed system, we proposed work launches a modified ranking, QOS grade of a position of cloud services without requiring additional real world service invocations from the intended users. Our proposed approach takes benefits of the previously proposed system's experiences for making be spoke ranking prediction for the current client of the cloud. Quality-of-service can be measured either at the server side or at the client side. Client-side QOS properties provide more accurate measurements of the user usage experience. The commonly used client-side QOS properties include response time, throughput, failure probability, etc. This paper mainly focuses on ranking prediction of client-side QOS properties, which likely have different values for different users (or user applications) of the same cloud service. To avoid the time-consuming and expensive real-world service invocations, QOS ranking is provided based on QOS values provided by the users and by monitoring the cloud services. Here, there is a chance of malicious QOS values, to overcome this problem Cumulative Sum Method (CUSUM) is proposed, by which we can detect the malicious QOS values. By eliminating these values, our Ranking accuracy will be enhanced.

In CUSUM method we can detect small mean variations occurs in QOS values. By detecting these changes we can know the malicious QOS ratings given by the users. Ultimately we can enhances the ranking of the system.

4. SYSTEM ARCHITECTURE

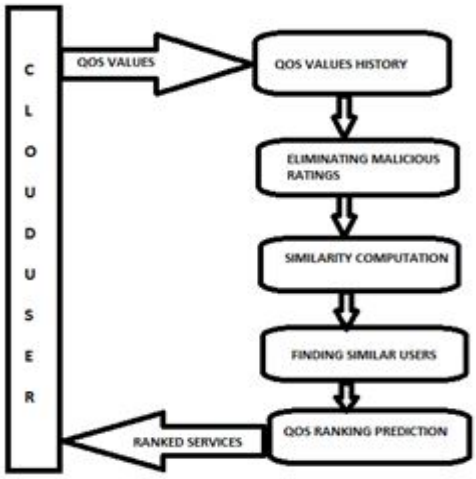


Fig. 2. System Architecture

5. PROPOSED METHODOLOGY

5.1CUSUM METHOD:

Cumulative Sum Method (CUSUM) to find out malicious Quality of services values.

This method comes under the variation detection algorithms that are based on hypothesis testing. This algorithm mainly include two hypothesis namely θ_0 and θ_1, which having probabilities $P\theta_0$ and $p\theta_1$, where Pθ_0 refers to the statistical distribution before to a change and the $p\theta_1$ refers to the distribution after a change. The likelihood ratio, defined by

$$S_n = \sum_{i=1}^{n} Si \quad (1)$$

Where $S_i = \ln \frac{p\theta 1(y)}{P\theta 0(y)}$

For each user feed backs, CUSUM monitor R_s a set of n rating sample range $\{y_1 \ldots . y_n\}$.

$$y_n = \sum_{I=1}^{n} r_i \quad (2)$$

Where r_i is the sum of all ratings in the n^{th} sample interval. μ_0 and μ_1 are the mean rating values before and after the change. Then CUSUM will be described as below

$$s_n = s_{n-1} + \frac{\mu 1 - \mu 0}{\sigma^2}\left(yn - \frac{\mu_1 + \mu_0}{2}\right) \quad (3)$$

In order to simplify approach to apply CUSUM to $x\sim_n$

$$x\sim_n = x_n - \mu_n \quad (4)$$

Here x_n refers to sum of all feedback ratings in the n^{th} sample, and μ_n is an estimate of the mean rate at the n-th sample.

For the $\mu_n$of equation (4), we can obtain it by using an exponential weighted moving average as follows

$$\mu_n = \lambda\mu_n - 1 + (1-\lambda)\, x_n \quad (5)$$

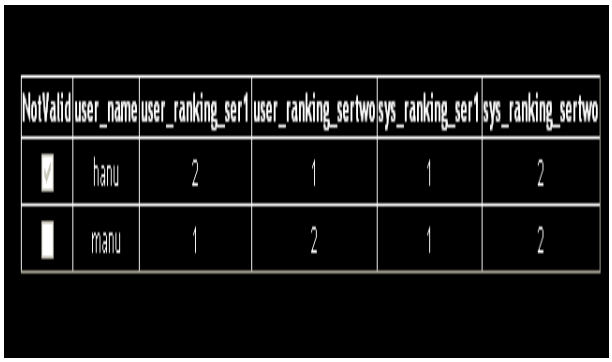
Where λ is the exponential weighted moving average factor.

Here the rating mean of $x\sim_n$ prior to a change is zero, hence, the mean in is $\mu 0 = 0$.Where as $\mu 1$, mean rate after the change. We don't know the exact value of $\mu 1$, hence, we take approximate value $\alpha\mu_n$, where α is amplitude percentage parameter, which corresponds to the most probable percentage of increase of the mean rate after a change. Then the CUSUM equation can be written as:

$$s_n = s_{n-1} + \frac{\alpha\bar{\mu}_{n-1}}{\sigma^2}\left(x_n - \bar{\mu}_{n-1}\frac{\alpha\bar{\mu}_{n-1}}{2}\right) \quad (6)$$

6. EXECUTION AND RESULT

Response-time refers to the time duration between the users sending out a request to a service and receiving the Response. In our system, ranking is provided based on the response time of the service. In the following figure(3) checked one malicious ranking given by the users. Table 1 shows the response time of the services and ranking information.



NotValid	user_name	user_ranking_ser1	user_ranking_sertwo	sys_ranking_ser1	sys_ranking_sertwo
☑	hanu	2	1	1	2
☐	manu	1	2	1	2

Fig. 3 malicious rating

Table 1. Ranking Description

Service Name		Response time of the service	User Ranking	System Ranking	Malicious rating (yes/no)
Weather service	Service provider1	69	2	1	**YES**
	Service provider2	72	1	2	
	Service provider3	103	3	3	
Currency service	Service provider1	22	3	3	**NO**
	Service provider2	13	2	2	
	Service provider3	10	1	1	

7. CONCLUSION

As we know now days, different organizations providing similar kind of cloud services. It may create great confusion in the user mind which service he may choose, To full fill his QOS requirements, By enabling this ranking mechanism, users are able to select their services as per there requirement. Here, there is a chance of malicious QOS values, to overcome this problem we proposed Cumulative Sum Method (CUSUM) by which we can detect the malicious QOS values. By eliminating these values, our Ranking accuracy is improved. By using this method malicious quality of services values are identified and discarded.

Scope of future enhancement

In future we would like to improve ranking accuracy by applying different techniques. In addition to this rating adjustment technique will be introduce

REFERENCES

[1] Zibin Zheng, Xinmiao Wu, Yilei Zhang, Michael R. Lyu and Jianmin Wang, "QoS Ranking Prediction for Cloud Services," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 6, June. 2013.

[2] K.J. arvelin and J. Kekalainen, "Cumulated Gain-Based Evaluation of IR Techniques,"ACM Trans. Information Systems,vol. 20, no. 4, pp.422-446, 2002.

[3] D. Ardagna and B. Pernici, "Adaptive service composition inflexible processes," Software Engineering, IEEE Transactions on, vol. 33, no. 6, 2007, pp. 369–384.

[4] J. R. Douceur. The Sybil Attack. *In Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS'01)*, pages 251-260, 2002.

[5] H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman. Sybil-Guard: Defending against sybil attacks via social networks. *IEEE/ACM Transactions on Networking*, 16(3): 576-589, 2008.

[6] Z. Yanzhen, G. Liang, L. Ge, X. Bing, and M. Hong. Rectifying prejudicial feedback ratings in reputation based trust management. *In Proceedings of the IEEE International Conference on Services Computing (SCC'07)*, pages 530-535, 2007.

[7] T. Yu, Y. Zhang, and K.-J. Lin, "Efficient Algorithms for Web Services Selection with End-to-End QoS Constraints," ACM Trans. Web, vol. 1, no. 1, pp. 1-26, 2007.