International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 21 (2018) pp. 15124-15129

Framework for Ranking Big Data Service Providers Using Classical Probability Ranking Principle

Dr. L. Aruna

Associate Professor, Department of P.G & Research Department of Computer Science,
Vivekanandha College of Arts and Science for Women (Autonomous),
Elayampalayam, Namakkal.(Dt), Tamil Nadu– 637205, India.

Abstract

Users can select the optimum cloud provider according to the required levels of services is very difficult and time consuming when data are too big. To select the best cloud service provider with federated cloud architecture in Big Data arena this paper aims to propose a classical probability ranking principle to resolve various problems faced by the user and providers in cloud environment for the big data arena. This proposed principle shortlists the cloud service provider for the big data based on the Quality of Service (QoS) and Service Level Agreement parameters. This proposed principle selects the best possible service provider using Service Measure Index (SMI). These SMI parameters are designed and implemented by Cloud Service Measurement Index Consortium (CSMIC) to shortlist the service providers. This proposed federated architecture can also be used to the new ranking mechanism for ranking the providers using classical probability ranking on the basis of the SMI parameters. All the shortlisted providers are examined and then ranking is done on the basis of the present and past values of SMI. This proposed principle is simulated and tested, results shows that the proposed algorithm works better than the existing model.

Keywords: Federated architecture, Cloud ranking, SLA parameters, Broker manager and Service Provider.

INTRODUCTION

Cloud computing and Big Data are very fast growing field providing all computation resources to the end users. The number of cloud service providers and data sets are rising day by day. Dataset in big data are characterized by Variety, Velocity and Volume. Cloud computing is an emerging paradigm that deliver resources on demand and pay per use approach. The divergent characteristic of big data makes this pay per use difficult and hurdles in providing quality [1,3] services to the users. Cloud computing offers more business benefits due to that reason many organizations have started developing applications on the cloud infrastructure and making their business more profits and flexible. Some cloud service providers guarantee the quality of their services by defining a set of Service Level Agreements (SLAs) with their customers for different dataset. These SLAs naturally lack any procedural means of enforcement which leaves the users data and software process under the total control of the cloud service provider. Any failure to meet the Service Level Agreements(SLA)[2] terms and obligations have terrible effects on the cloud customer and cloud provider, such as losing reputation and client trust and legal or financial penalties that may lead to putting an end to the entire business. This fact put the pressure and responsibility on the various cloud customers when selecting a particular cloud service provider for a particular dataset for running their service. It difficult to evaluate service levels of different cloud provider on their user, big data and QoS requirements on some attributes such as quality, reliability and security of an application. Hence, it is a demanding task to measure the performance of the cloud providers. Cloud Service Measurement Index Consortium (CSMIC) has identified metrics that are combined in the form of the Service Measurement Index (SMI), offering comparative evaluation of Cloud services.

These amount of indices can be used by various customers to compare different Cloud services. Several challenges are attempted in understanding a model for evaluating QoS and ranking Cloud providers. The following are the task of evaluating the providers is how to measure various SMI attributes of a provider and how to rank the providers based on the SMI attributes. In this paper, classical probability ranking principle technique is applied to rank among the shortlisted cloud providers, select and assign the optimum to the service.

PROPOSED DESIGN MODEL OF FEDERATED ARCHITECTURE

Efficient Framework for Ranking Big Data Service Providers in load balancing is used to find out the effective solution for a particular problem. The proposed the assignment approaches for the effective and efficient utilization of available virtual machine. [4,5]The assignment approach is mostly used for the cost minimization and effective utilization of cloud resources. The following algorithm explains the load balancing condition among different virtual resources which are available in the cloud environment.

Algorithm: Resource Selection and Monitoring provision using the assignment method.

Input: BigTask 'bti' in the cloud task pool, cti Є T = {1, 2,….n}, CloudResource 'crj' in the cloud data center, crj Є R = {1,2,…..m}, Cloud Cost Matrix (Cn x m) table.

Begin

1. for cti Є T do
2. for crj Є R do
3. Consider the Cost Matrix table (Cn x m)
4. Check the resource availability for all the tasks
5. for each task bti calculate min(ek) between resources do
6. $La \leftarrow bti(ek) - min(ek)$
7. for each resource crj calculate min(ek) between the tasks do
8. $Lb \leftarrow$ c$ti(ek) - min(ek)$
9. Find c$rj(ek) = 0$
10. for each task bti re-compute $min(ek)$ among resources do
11. Prepare List (Lc) with execution times in resources except cti(ek) is zero
12. Prepare List (Ld) with execution times consider by both tasks and in resources
13. Calculate $Lc - min(ek)$.
14. Calculate $Ld + min(ek)$.
15. Reconstruct cloud cost matrix ($C1n$ x m)
16. If tasks are in ready queue then
17. Task scheduler re-compute all tasks and then repeatedly do: (apply Round Robin method scheduling algorithm).
18. Specify the time quantum value based on execution time of tasks upon assigned resources.
19. Allocate the task to the resources which is best fitted based on assignment mechanism.
20. Insert the next cloud task into next round of queue, if the time quantum value is expired.
21. Maintain slot table for local mapping to record execution schedule of resources.
22. else
23. break
24. Endif
25. Calculate average execution time of cloud tasks .
26. Calculate throughput.
27. Calculate resource utilization rate.
28. Calculate scheduling success rate .
29. endfor
30. endfor
31. endfor
32. endfor

End

The main objective of the above algorithm efficient framework for ranking big data service providers in load balancing algorithm in IaaS cloud environment that aims to utilize the cloud resources of the virtual machine efficiently. [6]This approach is to identify the virtual cloud resources that must be suitable for all the applications and minimize the cost of applications. The algorithm minimizes the average execution time, maximizes the throughput, increases the resource [7]utilization rate and increases the scheduling success rate using these resources are supported to framework for ranking model.

The following assumptions considered in this ranking model are given as follows.

Table 1. Relative Importance.

Equal weight / quality	1
Somewhat more weight / better	3
Definitely more weight / better	5
Much more weight / better	7
Extremely more weight / better	9

Table 2. Category of QoS parameters.

QoS Parameters	**Higher the better**	**Lower the better**
Security	Yes	**No**
Integrity	Yes	No
Capacity	Yes	No
Scalability	Yes	No
Response Time	No	Yes
Mean Time Between Failure	No	Yes
Exception handling	Yes	No
Failure masking	Yes	No
Accountability	Yes	No
Failure semantics	Yes	No
Latency	No	Yes
Incomplete Transactions	No	Yes

Focus on the functional level of dependency: The same function may require different effort, depending on the situation. [8,9]This aspect may affect the values of certain attributes in the model. Users evaluate choices in linear order: This means that there is a linear order in which providers are assigned for service. In some case, user interested provider may also occupy the top slot.

ALGORITHM FOR CLOUD SERVICE PROVIDER SELECTION

Step 1: User submits the Service requirements and QoS preferences to the Broker Manager

Step 2: User requirements are identified to list the Service Providers

Step 3: Service providers are shortlisted

Step 4: Shortlisted service providers are ranked based on the proposed CPRP.

Step 5: CPRP considers and measures user requirements, service providers and feedback

Step 6: If user's required QoS parameters are not specified then rank the providers on standard weighing schemes and calculate ranking using CPPR.

Step 7: The best provider is selected using following formula.

$$\sum Ci\ x\ Ri \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.. (3.1)$$

C_i represents the performance, availability, reliability, feedback and security is normalized value of considered QoS parameters and R_i is relative importance of selected QoS parameter.

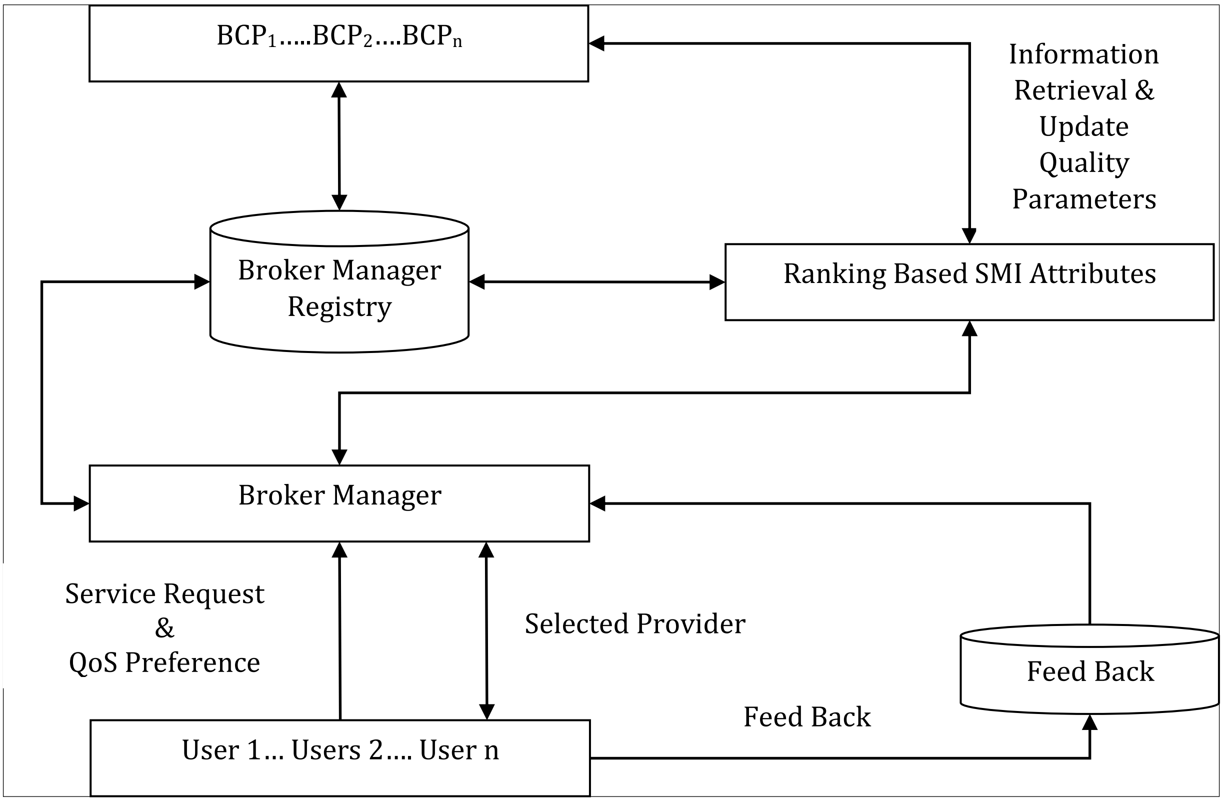

Figure 1: Customized Ranking Federated Architecture.

EXPECTED SELECTION LIST.

Now assume that the set of cloud providers C_i shortlisted for processing the services$_i$, the selected providers are given in the list L_i =< c_{i1}; c_{i2}; : : : ; $c_{i;ni}$>. For computing,[10] the expected benefit for the service by assigning the provider in list, Assume that the user considers the choices in the selected order, and the first provider in the list may be choice of assigning to the provider.

Cloud Resource Utilization Rate (c*ruj*): Resource utilization rate is the subtraction of starting execution time from finishing execution time of tasks on each resources.

$$cruj = \Sigma bti \text{ where } bti \text{ will be executed on } crj\ (btei - ctsi) \qquad \text{--(4.1)}$$

where, b*tei* is the finishing time and c*tsi* is the start time of task b*ti* on resource c*rj*.

Scheduling Success Rate (*SSRi,j*): It is based on [11]effective resource utilization of resources by the tasks.

SSRi,j → Σ*ti* = 1(c*ruj* / *m*) ---------------------------------(4.2)

where, c*ruj* is the cloud resource utilization rate of resource c*rj* and *m* is the number of tasks in each jobs.

OPTIMUM RANKING FROM THE SELECTED LIST

For discussing, the optimum ranking of selections from[12,13] the selected ordered list, the expected benefit for the service by ranking the provider and assign it $E(r_i)$ is given as follows

$$E(r_i) = \sum_{j=1}^{n}\left(\prod_{k=1}^{j-1}(1| - p_{ik})(e_{ij} + p_{ij}a_{ij}\right) + t_i^{l,i+1} \ldots (5.1)$$

$i \neq j \neq i+1$

Where

$$t_i^{i,i+1} = (e_{il} + p_{il}a_{il})\prod_{k=1}^{i=1}(1 - p_{ik}) + (e_{i,i+1} + p_{i,i+1}a_{i,i+1})\prod_{k=1}^{i}(1 - p_{i,k})$$

In the following, the probability of choice the selected provider accepted is Pij< 1 for j = 1… *I-1;* otherwise, choices of order of selected providers c_{il} and $c_{i;l+1}$ would never be reached, and their sequence would not matter. The order of selection between[14] first and second spot in the optimum list is given in the term $t_i^{i,i+1}$. It shows the difference [14,16]between the expected benefits of providers in the occupied position in the final ranking list. In order to simplify the derivation of the difference by the probability, that the user did not select any of the providers before the corresponding provider available from the ranking list. This simplified difference can be transformed as follows:

$$d_i^{i,i+1} = \frac{t_i^{i,i+1} - t_i^{i+1,i}}{\prod_{k=1}^{i-1}(1 - p_{ik})} e_{il} + p_{il}a_{il} + (1 - p_{il})(e_{i,i+1} + p_{i,i+1}a_{i,i+1}) - (e_{i+1} + p_{i,l+1}a_{i,i+1} + (1 - p_{i,i+1})(e_{i,l} + p_{il}a_{il})$$

$$p_{i,i+1}(e_{il} + p_{il}a_{il}) - p_{il}(e_{i,l+1} + p_{i,l+1}a_{i,i+1} \ldots\ldots (5.2)$$

Since $\prod_{k=1}^{i-1}(1 - p_{ik})$is positive, the expected benefit [18]of the original list is not less than that of the modified list iff $d_i^{i,i+1} \geq 0$.Formulate the effort of probability ranking principle for ranking and assign the providers is given as below. The effort of this ranking is between the average benefit if the provider is selected and the ratio between the users accepted provider and probability of the provider selected.

$$e(c_{i,l}) = a_{il} + \frac{e_{il}}{p_{il}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.3)$$

SIMULATION RESULT AND DISCUSSIONS.

With the increasing popularity of Cloud computing, many researchers are studied in the recital of clouds for special types of applications such as systematic computing, e-commerce and web applications. Simulation experiments were implemented on the JADE 4.3.0 platform [19,20] and on a computer whose configuration was an Intel Core i5-3337UCPU 1.80 GHz, 4.0GB RAM, Windows 10 (64 bits) operating system with Service Pack 2. Throughput and Average response time was computed for the experimental setup and the performances were also analyzed. The parameters that are taken into consideration for the simulation are based on the number of users, number of cloud service providers, task deadlines and others etc. The execution time for each task is assigned randomly between 0.5ms to 1.0ms. Numbers of users considered are 100, 500 and 1000 at a time. Number of service providers available is fixed as 100, and deadline for each request is fixed as 0.5ms. Every cloud service provider has 10 computing hosts and a time-shared VM scheduler. Cloud broker on behalf of user request consist of 128MB of memory, 512 MB of storage, 1 CPU, and time-shared Cloudlet scheduler. The broker requests instantiation of 5 VMs and associates one Cloudlet to each VM to be executed.

There are two experiments were conducted and performance is analyzed with existing approaches for structured data, unstructured data and semi structured data. To rank the matched providers using CPRP algorithm and the performance is revealed using[15] throughput and execution time in Table 3 and Figure 2.

Execution time of the ranking mechanism depends on its implementation and the overhead of the ranking mechanism is considerably reduced if the number of providers increased. Execution time denotes in Seconds. 1000 requests are submitted at time for testing the simulation. Throughput is defined as the number of messages exchanged between users and providers per second. Result shows that the execution time is decreased when the number of providers increased. Throughput performance is linear propositional to the number of providers is shown in Table 4 and Figure 3.

Table 3 Execution time of the ranking mechanism

Number of Cloud Providers	Executing Time		
	Structured Data	Semi Structured data	Unstructured Data
1	80	128	204.8
2	60	96	153.6
3	40	64	102.4
4	25	40	64
5	20	32	51.2
6	15	24	38.4
7	13	20.8	33.2
8	12	19.2	30.7
9	11	17.6	28.1
10	10	16	25.6

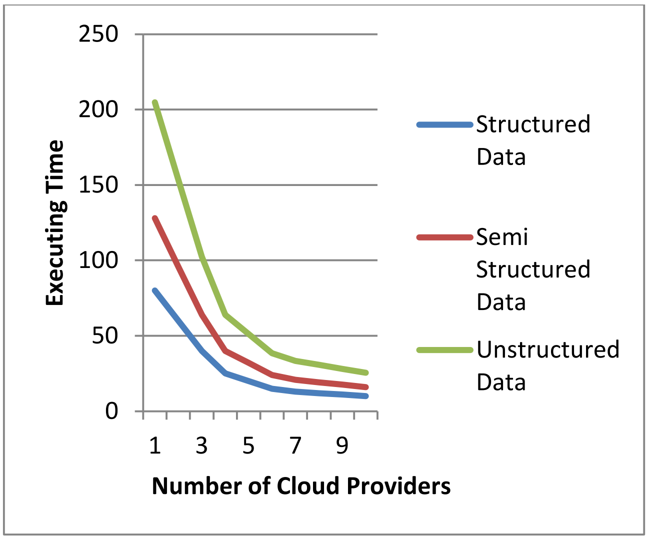

Figure 2 Graph 1: Execution time of the ranking mechanism

Table 4 Analysis the throughput of the ranking model

Number of Cloud Providers	Throughput (Kbps)		
	Structured Data	Unstructured Data	Semi Structured Data
1	10	6	9
2	40	24	36
3	60	36	54
4	75	45	67.5
5	85	51	76.5
6	95	57	85.5
7	100	60	90
8	105	63	94.5
9	110	66	99
10	115	69	103.5

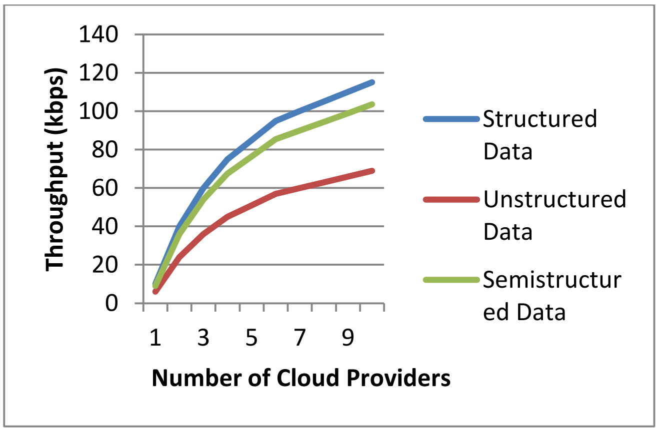

Figure 3. Graph 2: Analysis the throughput of the ranking model.

CONCLUSION AND FUTURE WORK

It has also becomes testing for users to find the best Cloud services which can satisfy their user and QoS requirements in terms of SMI parameters such as performance and security. To choose suitably between different services, users need to have a way to identify and measure key performance criteria that are important to their applications. This paper is discussed the new ranking mechanism for ranking the providers using new classical probability ranking on the basis of the SMI parameters. All the shortlisted providers are examined and then ranking is done on the basis of the present and past values of SMI. The future work on ranking cope with variation in QoS attributes such as performance by adopting evolutionary algorithms.

REFERENCES

[1]. Gartner Highlights Five Attributes of Cloud Computing. *Gartnercom*. 2016. Available at: http://www.gartner.com/newsroom/id/1035013. Accessed March 28, 2016.

[2]. Co je to Cloud computing a proÄ□ se o nÄ›mmluvÃ- - BusinessVize.cz. *Businessvizecz*. 2016. Available at: http://www.businessvize.cz/software/co-je-to-cloud-computing-a-proc-se-o-nem-mluvi.

[3]. A history of cloud computing. *ComputerWeekly*. 2016. Available at: http://www.computerweekly.com/feature/A-history-of-cloud-computing. Accessed March 28, 2016

[4]. Nwaoha T, Yang Z, Wang J, Bonsall S. Adoption of new advanced computational techniques to hazards ranking in LNG carrier operations. *Ocean Engineering*. 2013;72:31-44. doi:10.1016/j.oceaneng.2013.06.010. Available at: http://computer.howstuffworks.com/cloud-computing/internet-cloud-security-. Accessed March 28, 2016.

[5]. Almulla M, Yahyaoui H, Al-Matori K. A new fuzzy hybrid technique for ranking real world Web services. *Knowledge-Based Systems*. 2015;77:1-15. doi:10.1016/j.knosys.2014.12.021.

[6]. Rhoton, John. *Cloud Computing Explained*. [London?]: Recursive Press, 2010. Print.

[7]. Cloud Computing Manifesto. *Wikipedia*. 2010. Available at: https://en.wikipedia.org/wiki/Cloud_Computing_Manifesto. Accessed March 28, 2016.

[8]. Lu K, Yahyapour R, Wieder P et al. Fault-tolerant Service Level Agreement lifecycle management in clouds using actor system. *Future Generation Computer Systems*. 2016;54:247-259. doi:10.1016/j.future.2015.03.016.

[9]. Villegas D, Bobroff N, Rodero I et al. Cloud federation in a layered service model. *Journal of Computer and System Sciences*. 2012;78(5):1330-1344. doi:10.1016/j.jcss.2011.12.017.

[10]. Aljawarneh S. Cloud Security Engineering. *International Journal of Cloud Applications and Computing*. 2011;1(2):64-70. doi:10.4018/ijcac.2011040105.

[11]. Brennan M, Palaniswami M, Kamen P. Do existing measures of Poincare plot geometry reflect nonlinear features of heart rate variability?.*IEEE Transactions on Biomedical Engineering*. 2001;48(11):1342-1347. doi:10.1109/10.959330.

[12]. Buyya R, Garg S, Calheiros R. SLA-oriented resource provisioning for cloud computing: Challenges, architecture, and solutions. *2011 International Conference on Cloud and Service Computing*. 2011. doi:10.1109/csc.2011.6138522.

[13]. Grewal R, Pateriya P. A Rule-Based Approach for Effective Resource Provisioning in Hybrid Cloud Environment. *Advances in Intelligent Systems and Computing*. 2013:41-57. doi:10.1007/978- 3-642-35461-8_5.

[14]. Rajarajeswari C, Aramudhan M. Ranking Model for SLA Resource Provisioning Management. *International Journal of Cloud Applications and Computing*. 2014;4(3):68-80. doi:10.4018/ijcac.2014070105.

[15]. Ganghishetti P, Wankar R, Almuttairi R, Rao C. Rough Set Based Quality of Service Design for Service Provisioning in Clouds. *Rough Sets and Knowledge Technology*. 2011:268-273. doi:10.1007/978-3-642-24425-4_36.

[16]. Garg S, Versteeg S, Buyya R. SMICloud: A Framework for Comparing and Ranking Cloud Services. *2011 Fourth IEEE International Conference on Utility and Cloud Computing*. 2011. doi:10.1109/ucc.2011.36.

[17]. Bathla P, Vashist S. SLA Aware Cost based Service Ranking in Cloud Computing. International Journal of Application on Innovation in Engineering and Management. 2014;Vol.2(Issue 7).

[18]. Buyya R, Ranjan R, Calheiros R. Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities. *2009 International Conference on High Performance Computing & Simulation*. 2009. doi:10.1109/hpcsim.2009.5192685.

[19]. Buyya R, Ranjan R, Calheiros R. InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services. *Algorithms and Architectures for Parallel Processing*. 2010:13-31. doi:10.1007/978-3-642-13119-6_2.

[20]. Buyya R, Yeo C, Venugopal S, Broberg J, Brandic I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*. 2009;25(6):599-616. doi:10.1016/j.future.2008.12.001.