EAI Endorsed Transactions on Collaborative Computing Research Article **EAI.EU**

A Hybrid Service Ranking Based Collaborative Filtering Model on Cloud Web Service Data

Suvarna S. Pawar[1,*], Y. Prasanth[1]

[1]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Vaddeswaram - 522502, Guntur District, Andhra Pradesh, India

Abstract

INTRODUCTION: Trust is an important indicator in the cloud computing environment for service selection and recommendation. It is a difficult task to create a composite value-added service from several candidate services for the desired objectives due to the dramatic growth in services that have similar functionalities.
OBJECTIVES: This research aims to design a hybrid service feature ranking; cloud service ranking are computed using the advanced contextual service ranking measures. A hybrid collaborative approach is totally based on confidence to the QoS web service prediction.
METHODS: A new service ranking similarity computation is optimized for the cloud-based service selection. This collaborative filtering measure is used to check the top k customer selection by performing the top-k customer selection estimation on the cloud service ranking
RESULTS: The proposed method is useful in the prediction of QoS values and helps with optimal service ranking. As a result, similar/ relating cloud services are increasing, making it extremely complex to select the best cloud service among the relevant / similar services available
CONCLUSION: The state-of the-art approaches are proposed and tested on a mathematical QoS-Aware assessment framework. The use of semantic matching technique and QoS for web service ranking satisfies user requirements for web service recommendations. In addition, users require a web service not only based on functionality, but also based on high quality.

Keywords: Cloud web services, service ranking, collaborative filtering.

Received on 10 June 2021, accepted on 18 October 2021, published on 26 October 2021

doi: 10.4108/eai.26-10-2021.171599

1. Introduction

In present scenario, web services that implements a set of open applications focusing at interoperability along with compatibility with existing infrastructure support, appear to be the most efficient technology that solely relies upon SOC. Here a web service can be perceived to be a public interface for a specific application that can be invoked in order for performing a business function or a group of functions. The QoS properties can be used in order to evaluate the degree of conformance of the desired service to a specific quality requirement. Such properties are split into two categories such as technical and managerial. The technical properties necessarily elaborate the properties those are associated with the operation of the service incorporating availability, security and reliability. The managerial properties are related to management of the service integrating contract, cost, payment as well as ownership. In course of time, the computing resources have become less expensive, more powerful resulting from innovations brought into the processing as well as storage technologies.

Eventually, fast changing computing technology has given rise to "Cloud computing", an amazing computing environment where the resources such as storage, CPU, platform etc. are made available to users via global internet to users on demand basis. i)Providers of the infrastructure those hold the responsibility of managing the cloud platform

*Corresponding author. Email: sspawar.scoe@sinhgad.edu

thereby leasing the resources as per a usage-based pricing model, ii) Service providers those hold the responsibility of renting the resources from the providers of the respective infrastructure in order to serve the end users requirements. The impact of cloud computing in IT industry has led the large enterprises such as Amazon, Google, Microsoft etc. to compete among themselves in providing cost effective, more reliable and powerful cloud platform. Due to this growth and the widespread use of the Internet, the network traffic generated by web content requests and response has been an unusually large growth. If traffic increases so far as either the processing capacity or storage space of the server can easily max the bandwidth available on the Internet, user requests are dropped and access delays are increased and requests (i.e. lower throughput) are answered. Ever since the beginning of the internet, efforts have been undertaken to ensure not just that they provide users with web content but also that such content satisfies user's service quality expectations, such as higher performance and a minimum delay in responding to requests.

An initial solution was proposed by [1] to the problem of ensuring web content met the user expectations of QoS. This way, performance improved, server load reduced, and at the same time bandwidth usage reduced by using the proxies to serve the user request, especially for narrowband users [2][3]. It contributed to meeting growing Internet demands through improved speed, performance and accessibility. Copies of frequently asked documents from the server to the closer cache of the client successfully migrated speed. Speed was improved. In so doing, customer requests have experienced shorter delays. The use of server farms is another approach to improved performance. The system involves a set of replacement servers (distributed worldwide) which cache the contents of original Servers, routers and network elements which delivery contents to the optimal location and replacement server. Today, cloud computing is an affordable way for companies and content providers to expand their infrastructure by using a common pool of configurable computer resources which can be used by the same or different service providers [4].

The computer infrastructure of a cloud computing provider is built to efficiently supply cloud data centers based on the costs already incurred by the core companies using them. Therefore, [5] proposed a scheduling algorithm called Multiple QoS Restricted Multiple Workflow Scheduling Strategy to tackle the challenge arising from the unique QoS requirements of the multiple customers. In order to solve this problem of task planning, a mixed integer non-linear programming problem [6] was formulated.

They assumed that their model was multi- heterogeneous and parameters cost / performance computing and storage providers, as well as limitations on the highest number of cloud resources. This task reduced the total cost of the completion of work flow under deadlines. However, this paper is distinguished by the focus on tasks and flows optimization, whereas the focus of this paper is on web content delivery based on a QoS basis. A complete QoS demand for Big Data using cloud computing is a challenge, while minimizing overall costs. To address this challenge [7], heuristic algorithms have been proposed that were developed based on the assumption that reducing resource waste is directly associated with cost minimization. Those algorithms come with tuning parameters to find minimized solutions for the allotment of dynamic resources, but they don't take account of metrics like delays, jitters and delays. The Recommender System is an extensive technique which ensures that users receive valuable advice and results. In the CC environment several cloud services are launched, as the number of cloud users is growing.

2. Related work

The data sets of the research include the QoS dataset and synthetic dataset attributes. On the basis of various studies, the result provided by the algorithm ensures that the algorithm rating is computationally attractive and scalable [9].[10] Points out that there is increasing demand and popularity for the CC environment in the cloud services selection. As many cloud services resources exist in a dynamic cloud environment, choosing the best cloud service for their applications, especially with regards to online real-time applications, becomes complicated. It offers low quality, increasing computing and high processing times as well as the lack of the service selection framework.

The results for the cloud service selection, which represents the adaptive features, are dynamically adaptive learning techniques. The method is designed to dynamically optimize the Cloud Selection Service, providing the user with the best output [5]. However, because it takes few parameters, the ranking of the service level is very low and not regarded as a suitable method for cloud service providers selection. To identify the most reliable and appropriate cloud service providers, a reliable approach of HBFFOA (Hybrid Binary Fruit Fly Optimization Algorithm) for the service ranking has been developed. A mutational probability feature is used in HBFFO to ignore local optima.

A cloud environment assessment of the trust can be done using the QoS (Qualities of Service) attributes by using a WSDream#2 data set, user needs recognition and the use of compatible CSP, service rankings, data credibility etc. [11] Many cloud services exist in a cloud scenario for real-world applications. In addition to the authentication service offering for cloud service providers and sensor network providers, It provides three types of functions proposes the approach to enhancing service confidence evaluation through the adoption of a trustworthy cloud service providers selection framework called 'TRUSS.'

As a result, the Gaussian cloud transformation frames Multi-granularity Selection Standard of trust standards. The calculation model of user preferences is then developed on the basis of cloud analytical hierarchy. Ultimately, an experimentally authenticated two-step, fluidized evaluation of the confident cloud service selection algorithm [13] is recommended. From the advantage point of view, a CSP must propose to maximum users its services and thus increase the precision of the advice by using integrated data for the service recommendation.WS- DREAM validates the proposed policy of scalability, accuracy of recommendations, and ability to preserve the privacy of the services [14]. A virtual network of different services is created by Cloud Computing to numerous customers worldwide. A reimbursement is required and based on the quality of service provided by the cloud. These services vary according to the services and resources involved [15]. The authors proposed a novel framework for ranking and advanced cloud services with Quality of Service (QoS) features in the study submitted in [16].

The differences between these frameworks and the context [16] give cloud providers healthy competition. As QoS requirements are dependent on the applications used for evaluating these suppliers. A minor drawback is that QoS attributes such as cost, service validation, safety etc. can be used only for quantifiable purposes. This means that the reaction time, transparency and interoperability cannot be worked out. The authors took a different course in [17] in order to prevent the costly call for real-world services. Rather, during the decision-making phase they incorporate previous service use experience in their QoS prediction frameworks. Only when it comes to cloud can the collaborative approach used to prediction QoS services be used. For example, the coefficient of Pearson correlation is used for determining the similarity of the users. A generic Cloud Workflow Systems QoS framework has been proposed in [18].

Recent research [19] shows that most service selection strategies are developed on the basis of the weighted combination of various aspects of the cloud service QoS parameters to identify the best service according to user preference [20]. Studies under [21- 23] specifically examine service composition models in which the quality-of-service group models have been demonstrated in order to evaluate the QoS in the form of an optimized composite service for individual candidates. In assessing the QoS parameters and selecting the best service, the number of applicant services and time is not reduced significantly. The QoS values of Cloud services are initially clustered using Master Component Analysis to reduce overhead discovery and a number of candidate services. In accordance with our knowledge nothing has been done to develop a multi-level model for the selection of Cloud services with modified with Master Component Analysis.

This paper utilizes an algorithm for the Term Frequency and Inverse Document Frequency (TF-IDF) for the filtering of the nuclear services obtained from the multi-cloud environment in accordance with the service request. In [25], a model using a major component analysis to analyze service quality parameters as a multi-media network. The method is proven effective in their studies and experiments, but this method has not been used for the selection of cloud services. In addition, the author proposed in [26] the effective and efficient selection approach to the composition of the QoS cloud service. In this paper, the researcher adopted the cloud model for reliable services to calculate the value of the incertitude of the cloud services that are redundant. In [27] the author implemented a service selection strategy based on the QoS ratings of cloud service applicants without taking into consideration the context.[28] QoS requirements, based on their QoS parameters like price, reliability, accessibility and time of computation, are clustered into various classes and detailed. A discriminant analysis model based on the service success rate was developed in [29].

This paper recommends an integrated method of confidence assessment, with the objective and subjective trust assessment in order to build an efficient trust assessment model of TRUSS. The performance of the proposed framework is assessed by simulation-based testing, although the method suggests that the main users are honest and that the dishonest users are given a larger number of unfair ratings [2].

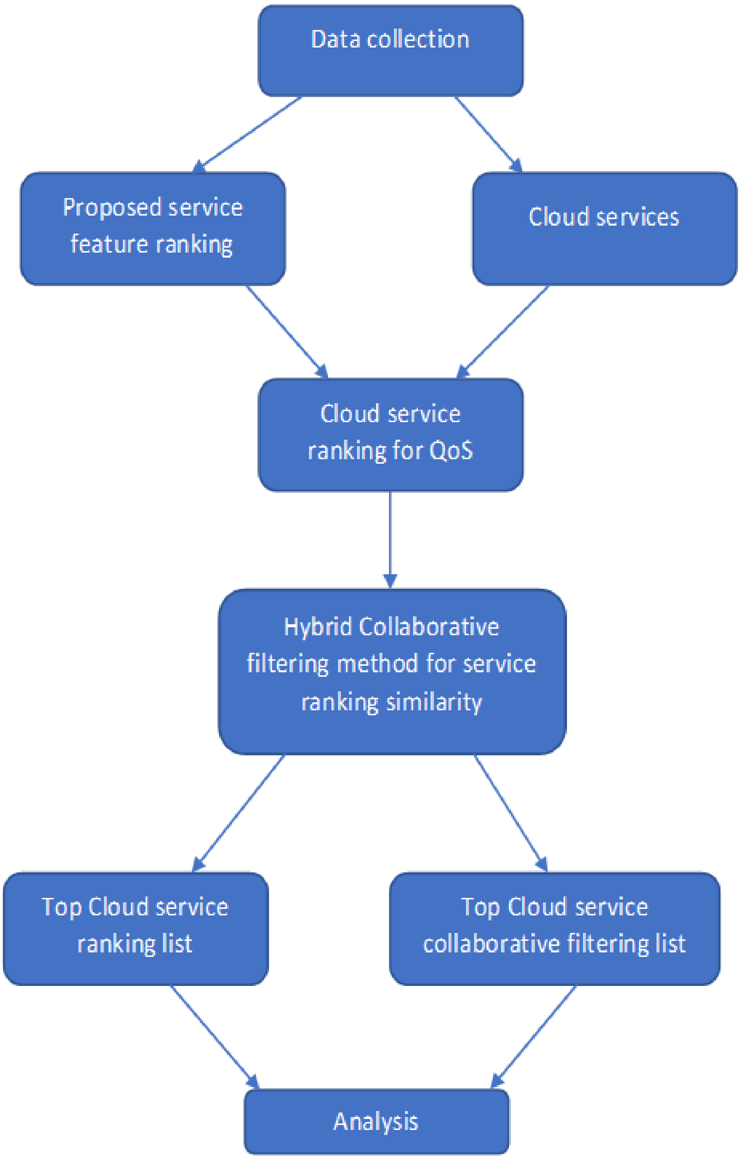

Figure 1. Hybrid Collaborative Filter model

3. Methodology

In this work, a hybrid service feature ranking, cloud service ranking is computed using the advanced contextual service ranking measures. In the proposed framework, initially an advanced cloud service ranking and its similarity is computed using the novel collaborative filtering measure. In the hybrid collaborative filtering method, a new service ranking similarity computation is optimized for the cloud-based service selection. This collaborative filtering measure is used to check the top k customer selection by performing the top-k customer selection estimation on the cloud service ranking. Here, probabilistic cloud service selection is used to compute the cloud service ranking similarity based on the attribute utility measure and the top-k customer selection estimation as shown in fig 1. A proper service selection framework is developed to help users select the best cloud provider, while at the less time encouraging the cloud service providers to comply with and fulfil the Service Level Agreement. The selection framework assigns random weights to the QoS attributes and randomly replaces missing data that do not accurately rank the cloud service providers. Therefore, the minimum distance property algorithm is suggested in order to accurately rank cloud service provider.

4. Results and Discussion

Experimental results are simulated in java environment with third party libraries. In this work, the cloud web service training data is taken as input for service ranking and collaborative filtering process. In this experimental results, different simulation results are performed on the training dataset with different cloud services and its associated tasks. Experimental results proved that the present model has high computational efficiency in terms of cloud service ranking and collaborative filtering. They concentrated on the problem of the expansion of the requests faced by the provider continuously. The expansion number of requests makes it difficult for the cloud, within the requested time, to supply or at least to recognize requests. Few of the QoS properties are used by its proposed framework to solve this critical problem. Again, these frameworks measured cloud services quality and priority.

Table 1. Cloud service-based ranking and collaboration filtering

Initializing input matrices (e.g. exec time & communication time matrices

Best Runtime at iteration (0): 1890.305603
Best Runtime at iteration (10): 1890.305603
Best Runtime at iteration (20): 1890.305603
Best Runtime at iteration (30): 1890.305603
Best Runtime at iteration (40): 1890.305603
Best Runtime at iteration (50): 1890.305603
Best Runtime at iteration (60): 1890.305603
Best Runtime at iteration (70): 1833.339252
Best Runtime at iteration (80): 1833.339252
Best Runtime at iteration (90): 1833.339252
Best Runtime at iteration (100): 1779.142432

28	SUCCESS	30	30	371.27
10	381.27	0.68525	0.96057	0.96949
34	SUCCESS	36	36	374.72
10	384.72	0.72107	0.93486	0.9679
54	SUCCESS	56	56	379.1
10	389.1	0.86069	0.92174	0.96484
81	SUCCESS	83	83	380.54
10	390.54	0.96166	0.9607	0.97947
65	SUCCESS	67	67	386.3
10	396.3	0.92479	0.95486	0.96456
97	SUCCESS	99	99	386.96
10	396.96	0.77089	0.95956	0.96308
64	SUCCESS	66	66	388.18
10	398.18	0.8819	0.96971	0.96974
53	SUCCESS	55	55	392.67
10	402.67	0.80103	0.9651	0.96377
31	SUCCESS	33	33	397.43
10	407.43	0.72075	0.96012	0.96764
13	SUCCESS	15	15	409.37
10	419.37	0.82709	0.9615	0.96716
74	SUCCESS	76	76	413.01
10	423.01	0.65879	0.92214	0.97279
68	SUCCESS	70	70	417.27
10	427.27	0.93004	0.94135	0.96394
89	SUCCESS	91	91	424.21
10	434.21	0.95428	0.94306	0.97357
88	SUCCESS	90	90	438.67
10	448.67	0.75823	0.92015	0.96332
30	SUCCESS	32	32	445.46
10	455.46	0.62972	0.96662	0.96815
45	SUCCESS	47	47	451.14
10	461.14	0.80506	0.93465	0.96215
12	SUCCESS	14	14	453.42
10	463.42	0.96028	0.95118	0.97522
15	SUCCESS	17	17	471.24
10	481.24	0.63061	0.95642	0.9724
43	SUCCESS	45	45	475.49
10	485.49	0.88709	0.95474	0.96146
39	SUCCESS	41	41	478.95
10	488.95	0.951	0.96972	0.97622
77	SUCCESS	79	79	481.5
10	491.5	0.6484	0.9338	0.97444
98	SUCCESS	100	100	481.96
10	491.96	0.8112	0.94032	0.97056
16	SUCCESS	18	18	506.04
10	516.04	0.85178	0.92493	0.96569
71	SUCCESS	73	73	510.11
10	520.11	0.96441	0.94854	0.96542
4	SUCCESS	6	6	514.43
10	524.43	0.68468	0.96932	0.97673
36	SUCCESS	38	38	518.16
10	528.16	0.77131	0.92579	0.96557
92	SUCCESS	94	94	521.39
10	531.39	0.95266	0.94067	0.97623
17	SUCCESS	19	19	546.99
10	556.99	0.84798	0.9499	0.97425
90	SUCCESS	92	92	548.27
10	558.27	0.78049	0.92812	0.97029
72	SUCCESS	74	74	553.95
10	563.95	0.6871	0.92853	0.97983
5	SUCCESS	7	7	558.07
10	568.07	0.71649	0.94674	0.96591
73	SUCCESS	75	75	562.34
10	572.34	0.80142	0.94318	0.96175
94	SUCCESS	96	96	563.82
10	573.82	0.85392	0.95138	0.9672

86	SUCCESS	88	88	565.76
10	575.76	0.72494	0.92566	0.97791
19	SUCCESS	21	21	569.79
10	579.79	0.83263	0.95024	0.97251
48	SUCCESS	50	50	570.11
10	580.11	0.96067	0.92505	0.97495
83	SUCCESS	85	85	573.39
10	583.39	0.87647	0.96539	0.97191
22	SUCCESS	24	24	587.33
10	597.33	0.77939	0.95805	0.97211
69	SUCCESS	71	71	589.05
10	599.05	0.85174	0.92784	0.96495
37	SUCCESS	39	39	593.54
10	603.54	0.95107	0.96595	0.96424
50	SUCCESS	52	52	594.92
10	604.92	0.71322	0.94527	0.966
80	SUCCESS	82	82	601.16
10	611.16	0.82145	0.93506	0.97197
62	SUCCESS	64	64	620.98
10	630.98	0.80532	0.92959	0.96118
41	SUCCESS	43	43	626.7
10	636.7	0.75259	0.96888	0.97418
57	SUCCESS	59	59	627.1
10	637.1	0.90059	0.93017	0.96421
8	SUCCESS	10	10	636.05
10	646.05	0.86439	0.92621	0.96395
93	SUCCESS	95	95	638.01
10	648.01	0.70391	0.95421	0.9708
70	SUCCESS	72	72	640.96
10	650.96	0.94336	0.96019	0.973
52	SUCCESS	54	54	645.75
10	655.75	0.94666	0.96055	0.96691
3	SUCCESS	5	5	645.81
10	655.81	0.73012	0.9675	0.97846
23	SUCCESS	25	25	650.15
10	660.15	0.82937	0.95125	0.96753
67	SUCCESS	69	69	651
10	661	0.87106	0.95199	0.97335
59	SUCCESS	61	61	655.07
10	665.07	0.68378	0.96912	0.97962
84	SUCCESS	86	86	662.87
10	672.87	0.67548	0.92217	0.97887
18	SUCCESS	20	20	681.05
10	691.05	0.8832	0.92031	0.97017

25	SUCCESS	27	27	706.53
10	716.53	0.89709	0.933	0.96661
60	SUCCESS	62	62	708.59
10	718.59	0.73799	0.94576	0.97368
85	SUCCESS	87	87	709.29
10	719.29	0.79638	0.94614	0.96729
55	SUCCESS	57	57	718.02
10	728.02	0.73625	0.94437	0.96706
42	SUCCESS	44	44	718.44
10	728.44	0.74963	0.95973	0.96977
82	SUCCESS	84	84	728.74
10	738.74	0.90637	0.93131	0.96504
66	SUCCESS	68	68	741.71
10	751.71	0.90527	0.96377	0.96196
49	SUCCESS	51	51	758.3
10	768.3	0.83643	0.95724	0.97754
20	SUCCESS	22	22	758.72
10	768.72	0.78457	0.92451	0.96925
61	SUCCESS	63	63	784.63
10	794.63	0.72561	0.96374	0.97519
95	SUCCESS	97	97	791.3
10	801.3	0.88599	0.94611	0.97446
87	SUCCESS	89	89	801.51
10	811.51	0.80883	0.9241	0.96876
58	SUCCESS	60	60	803.53
10	813.53	0.74504	0.95998	0.97969
24	SUCCESS	26	26	807.98
10	817.98	0.65087	0.95794	0.97642
96	SUCCESS	98	98	818.79
10	828.79	0.64209	0.92093	0.97281
63	SUCCESS	65	65	822.24
10	832.24	0.81775	0.92688	0.96898
14	SUCCESS	16	16	826.9
10	836.9	0.76234	0.92733	0.97636
21	SUCCESS	23	23	842.01
10	852.01	0.62985	0.94468	0.97388
76	SUCCESS	78	78	864.33
10	874.33	0.89061	0.92015	0.96174
99	SUCCESS	101	101	872.97
10	882.97	0.81173	0.93545	0.96948
35	SUCCESS	37	37	880.97
10	890.97	0.61835	0.96486	0.96071
91	SUCCESS	93	93	889.81
10	899.81	0.80546	0.94575	
0.965110.1:				

Table 1, describes the experimental results of different cloud instances and its associated ranking and collaborative filtering trust measure for cloud service selection. In the above table, each cloud service is selected based on the ranking and collaborative filtering measure for decision making process. This resource allocation strategy ensures that only when requested and until use is made for resources by the provider. The QoS-based resource assignment model, as referred to, assumed a multiple- competitive system, each with their own system resource based QoS levels. The QoS-based resource assignment model aims to allocate resources to each application so that the total system utility is maximized in accordance with the requirements that each application can be made available with regard to each QoS dimension. All applications must be added to the total system utilities to be maximized.

Table 2. Comparative analysis of different cloud web service ranking measures and its runtime(ms)(T=0.8)

#VM	IBGSS	IBGSS Rank2	IBGSSRank2+QPred	ProposedModel
VM-0	7477	6837	6401	5802
VM-1	7665	6869	6435	5674
VM-2	7650	7659	6339	5935
VM-3	7224	7576	6365	5876
VM-4	7380	6647	6348	5751
VM-5	7550	7293	6459	5670
VM-6	6600	7148	6309	5721
VM-7	7764	7752	6415	5582
VM-8	7465	6703	6407	5636
VM-9	7206	7508	6363	6024
VM-10	6962	7234	6381	5938

Table 2, describes the comparative study of proposed model to the conventional models on runtime analysis of proposed cloud web service ranking to the conventional models. In this table, the threshold of 0.8 is used to find the runtime computations. This process has four key components: Quality requirements, QoS service selection, QoS consistency monitoring and QoS violations. In this framework, the process has been divided into four main components. This generic QoS framework, however, does not solve difficult problems. The generic QoS framework lacks communication and the exchange of knowledge between the various components.

Table 3. Comparative analysis of proposed model to the conventional models on runtime analysis (T=0.9)

#VM	IBGSS	IBGSSRank2	IBGSSRank2+QPred	Proposed
VM-0	6803	6792	6436	5622
VM-1	6599	7428	6327	5941
VM-2	7467	7011	6440	5756
VM-3	7553	7688	6329	5911
VM-4	7681	6622	6268	5959
VM-5	6790	6728	6406	6025
VM-6	7486	7392	6399	5655
VM-7	7519	7594	6314	5823
VM-8	7240	7011	6461	6010
VM-9	7305	6807	6391	5747
VM-10	6569	7036	6434	5760

Table 3, describes the comparative study of proposed model to the conventional models on runtime analysis of proposed cloud web service ranking to the conventional models. In this table, the threshold of 0.9 is used to find the runtime computations. In his work, instead of taking the QoS values proposed by the cloud service provider, to enhance confidence in the service composition model by taking into account previous QoS records of cloud services. This approach is specifically based on the summary weighted by QoS.

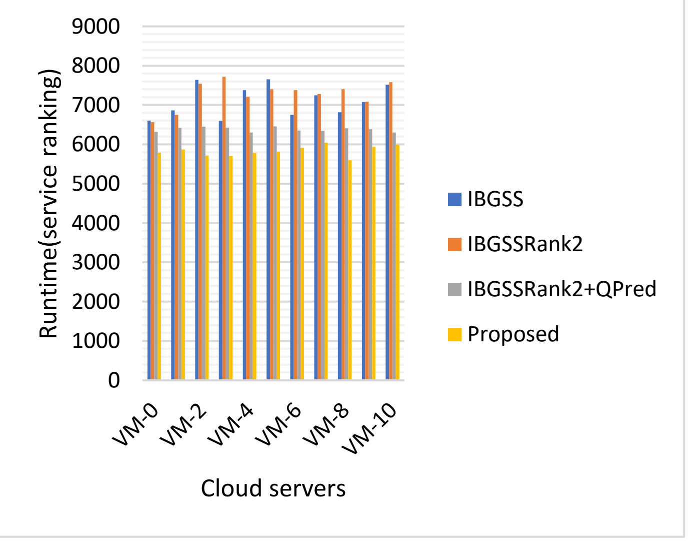

Figure 2. Comparative analysis of proposed cloud web service ranking to the conventional models on training dataset.

It describes the comparative study of proposed model to the conventional models on runtime analysis of proposed cloud web service ranking to the conventional models. In this figure, the threshold of 0.95 is used to find the runtime computations. Workflows with different QoS requirements can be started at any time and are scheduled with a high level of success on arrival. The results of this algorithm experiments have produced better planning results. QoS restrictions such as availability and reliability have not been taken into account.

Table 4. Comparative analysis of proposed model to the conventional cloud service collaborative filtering measure to the proposed model (T=0.9)

#VM	IBGSS	IBGSS RANK 2	IBGSS RANK2 +QPRE D	PROP OSED
VM-0	0.856	0.901	0.904	0.95
VM-1	0.861	0.892	0.909	0.961
VM-2	0.859	0.894	0.899	0.947
VM-3	0.882	0.853	0.906	0.961
VM-4	0.882	0.893	0.882	0.95
VM-5	0.86	0.879	0.865	0.963
VM-6	0.92	0.867	0.864	0.944
VM-7	0.9	0.9	0.906	0.969
VM-8	0.867	0.864	0.892	0.959
VM-9	0.912	0.9	0.885	0.961
VM-10	0.883	0.918	0.901	0.967

Table 4, describes the comparative study of proposed model to the conventional ranking models using cloud web service data. In this table, the average ranking measure is computed based on the available cloud web services. A new framework that reduces the computer complexity and the correlations of the QoS attributes is therefore needed. In a model using a major component analysis to analyse service quality parameters as a multi-media network. The method is proven effective in their studies and experiments, but this method has not been used for the selection of cloud services.

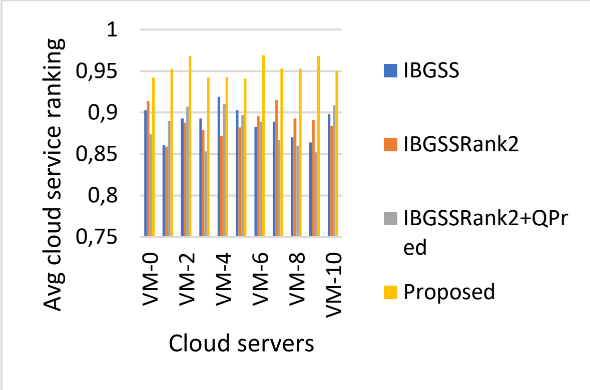

Figure 3. Comparative analysis of proposed model to the conventional cloud service collaborative filtering measure to the proposed model (T=0.95) using cloud web service data.

In this Figure, the average ranking measure is computed based on the available cloud web services. The main purpose of this algorithm is to detect the correlation between QoS attributes, which not only causes high computer complexity but also leads to computer errors. A new framework that reduces the computer complexity and the correlations of the QoS attributes is therefore needed.

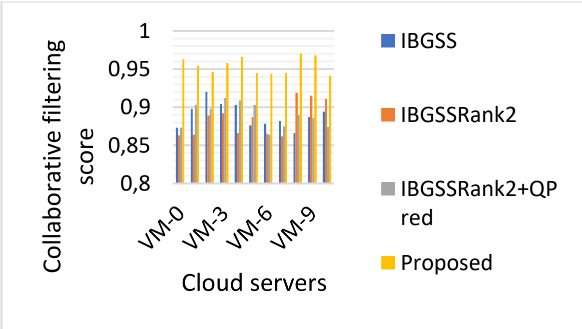

Figure 4. Comparative analysis of proposed model to the conventional cloud service collaborative filtering measure to the proposed model (T=0.95)

It describes the comparative study of proposed model to the conventional collaborative filtering measures using cloud web service data. In this Figure, the average ranking measure is computed based on the available cloud web services. This work uses a modified Master Component Analysis to analyses QoS and to further classify the selected cloud services according to user preferences. The key contributions of this work include a significant reduction in the overhead and calculation rate of service discovery because this reduces the number of applicant services, thus guaranteeing optimum selection of the best service on the basis of the service application.

6. Conclusion and Future Work

As the number of similar web service features increases, the problem of service selection becomes more important. Web service QoS is considered to be the secondary method for service selection. Using QoS, service recommendations help users to choose high quality service. Continuous monitoring of the web service is required for accurate parameter value. This directly affects the accuracy of the value of a parameter. There is therefore a need for a method of QoS computation that considers all aspects of the web service as distinct. The use of semantic matching technique and QoS for web service ranking satisfies user requirements for web service recommendations. In addition, users require a web service not only based on functionality, but also based on high quality. A variety of Web Service Composition (WSC) approaches have now been implemented in order to address this challenge and have an important impact on composition efficiency. However, the effects on service composition processes of such approaches are not known. The state-of-the-art approaches are proposed and tested on a mathematical QoS-Aware assessment framework. In this work,, a hybrid cloud service ranking and collaborative filtering model is designed and implemented on the training data for better .

References

[1] Dasari R., Prasanth Y., NagaRaju O. (2017),'A framework for multipurpose cloud based data centre network security',International Journal of Applied Engineering Research,12(22),PP.12189-12196.

[2] Z. Zheng , Y. Zhang , M.R. Lyu , Investigating QoS of real- world web services, IEEE Trans. Services Comput. 7 (1) (2014) 32–39.

[3] Zheng, Z., Ma, H., Lyu, M.R., & King I.(2011).Qos-aware web service recommendation by collaborative filtering.IEEE Transactions on Services Computing, 4, 140-152.

[4] Q. Shambour , J. Lu , A hybrid trust-enhance collaborative filtering recommendation approach for personalised government-to-business e-services, Int. J. Intell. Syst. 26 (9)(2011)814–843.

[5] Kumar M.R., Yosepu C., Prakash A., Kamesh D.B.K. (2017),'Data safety in the service cloud computing using today's scenario',Journal of Engineering and Applied Sciences,12(Specialissue12),PP.9382-9385.

[6] L. Yao , Q. Sheng , A. Ngu , J. Yu , A. Segev , Unified collaborative and content-based web service recommendation, IEEE Trans. Services Comput. 8 (3)(2015) 453–465 .

[7] Amarendra, C., &Harinadha Reddy, K. (2019)." PSO algorithm support switching pulse sequence ISVM for six- phase matrix converter-fed drives", doi:10.1007/978-981-13- 1921-1_55.

[8] Z. Zheng , H. Ma , M. Hao , M.R. Lyu , I. King , QoS- aware web service recommendation by collaborative filtering, IEEE Trans. Services Comput.4(2)(2011)140–152.

[9] Balaji P., Nagaraju O., Haritha D. (2017),'CommTrust: Reputation based trust evaluation in E-Commerce applications',Proceedings of the 2017 International Conference On Big Data Analytics and Computational Intelligence, ICBDACI2017,(),PP.318-323.

[10] J. Xu , Z. Zheng , M.R. Lyu , Web service personalized quality of service prediction via reputation-based matrix factorization, IEEE Trans. Reliability 65 (1) (2016)28–37.

[11] C. Wu , W. Qiu , Z. Zheng , X. Wang , X. Yang , QoS prediction of web services supported two-phase K-means clustering, in: IEEE International Conference on Web Services,2015,pp.161–16.

[12] Anandakumar H., Arulmurugan R., Roshini A. (2019), 'Analysis of Cloud Based Simulation Methods and Optimization Algorithms', Proceedings of the 2nd International Conference on Smart Systems and Inventive Technology, ICSSIT 2019, (),PP.1270-1274.

[13] Ding, S., Wang, Z., Wu, D., & Olson D.L. (2017). Utilizing customer satisfaction in ranking prediction for personalized service selection, Decision Support Systems, 93, 1-10

[14] Zhou, T., Kuscsik, Z., Liu, J.G., Medo, M., Wakeling, J.R., &Zhang,Y.C.(2010). Solving the apparent diversity- accuracy dilemma of recommender systems. Proceedings of Natioanl Academy of u.s. of America, 107,4511-4515.

[15] Patil Abhijit J., Syam Prasad G. (2018),'Trust based security model for IoT and fog based applications',International Journal of Engineering and Technology(UAE),7 (),PP.691-695

[16] Gouse, S. M., & Krishna Mohan, G. (2018). "A survey on challenges and future perspectives in big data using cloud- based technologies". Journal of Advanced Research in Dynamical and Control

Systems, 10(9 Special Issue), 623- 631.

[17] Kiranbabu M.N.V., Satyanarayana K.V.V. (2019), 'A perusal inspection on ranking the cloud service provider in cloud computing', International Journal of Recent Technology and Engineering, 7(6), PP.531-536

[18] Herlocker, J., Konstan, J., &Terveen, L.(2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems, 22,5-53.

[19] Marden, J.I.(1995). Analyzing and modeling rank data. London: Chapman and Hall (chapter 4).

[20] Sasi Bhanu J., Lakshmi Prasad M., Sastry J.K.R. (2018),'A combinatorial particle swarm optimization (PSO) technique for testing an embedded system',Journal of Advanced Research in Dynamical and Control Systems,10 (7),PP. 321-336

[21] Kulkarni, N., Lalitha, S. V. N. L., &Deokar, S. A. (2019). "Real time control and monitoring of grid power systems using cloud computing". International Journal of Electrical and Computer Engineering, 9(2), 941-949. doi:10.11591/ijece.v9i2.pp.941-949.

[22] Sujatha M.M., Prasada Rao P.V.R.D., Sastry J.K.R. (2019), 'Metrics for assessing quality of a web site', International Journal of Innovative Technology and Exploring Engineering,8(8),PP.1710-1714.

[23] Gavvala, S. K., Jatoth, C., Gangadharan, G. R.,&Buyya, R. (2019). "QoS-aware cloud service composition using eagle strategy". Future Generation Computer Systems, 90, 273-290.doi:10.1016/j.future.2018.07.062.

[24] B.H. Li, L. Zhang, S.L. Wang, F. Tao, J.W. Cao, X.D. Jiang, X. Song, X.D. Chai,Cloud manufacturing: a replacement service-oriented networked manufacturing model, Comput. Integr.Manuf. Syst. 16 (1) (2010) (in Chinese)X. Xu, From cloud computing to cloud manufacturing, Robot. Comput.Integr. Manuf. 28 (1) (2012)75–86.

[25] Naresh T., Lakshmi A.J., Reddy V.K. (2017),'An efficient resource allocation strategy based on improved particle swarm optimization (IPSO)',Pakistan Journal of Biotechnology,14(Special Issue2),PP.125-128.

[26] Venkata Raghavarao Y., Sasidhar K., Sastry J.K.R., Chandra Prakash V. (2018),'Quantifying quality of WEB sites based on content',International Journal of Engineering and Technology(UAE),7 (2),PP.138-141

[27] X. Yao, Y. Lin, Emerging manufacturing paradigm shifts for the incoming age, Int. J. Adv. Manuf. Technol. 85 (5–8)(2015),1665–1676.

[28] Potluri S., Rao K. (2017),'Efficient hybrid qos driven task scheduling algorithm in cloud computing using a toolkit: Cloudsim',Journal of Advanced Research in Dynamical and ControlSystems,9(),PP.1270-1283.

[29] Wu, J., Chen, L., Feng, Y., &Zheng, Z.(2013). Predicting quality of service for selection by neighborhood-based collaborative filtering. IEEE Transactions on Systems, Man & Cybernetics, 43,428-439

[30] Senthil Kumar A.M., Venkatesan M. (2019), 'Multi- Objective Task Scheduling Using Hybrid Genetic-Ant Colony Optimization Algorithm in Cloud Environment', Wireless Personal Communications, 107(4),PP.1835-1848.

[31] Phani Sheetal A., Ravindranath K. (2018),'Software metric evaluation on cloud based applications',International Journal of Engineering and Technology(UAE),7 (0),PP. 13- 18

Appendix A

Service Ranking Similarity:

The following equation is used to find the service ranking similarity of each cloud service wrt the users. Here, U, V represents the users. Users access cloud web services by using the quality of service q. Uth user will access the ith cloud web service by using the quality of service measure $q_{u,i}$. Simlarly, V th user will access the ith cloud web service by using the quality of service measure $q_{v,i}$. Here, $\tilde{I}(x)$ is the web service allocation indicator which is used to enable and disable the user's cloud web service.

$$\mathrm{Sim}(u,v) = 1 - \frac{4 \times \sum_{i,j \in I_u \cap I_v} \tilde{I}((q_{u,i} - q_{u,j})(q_{v,i} - q_{v,j}))}{|I_u \cap I_v| \times (|I_u \cap I_v| - 1)}$$

where $|I_u \cap I_v|$ is the commonly accessed services by the users u and v. $q_{u,i}$ is the ith service accessed by the uth user and $\tilde{I}(x)$ is an status indicator

$$\tilde{I}(x) = \begin{cases} 1, & \text{if } x < 0 \\ 0, & \text{if } x \geq 0 \end{cases}$$

Collaborative Filter Measure:

Once the cloud web serivice ranking of each user towards the available cloud services are completed. Next step is to find the collaborative filtering of each web service by using the user's service ranking and its quality of service metrics.

Let $CWS_{ij} = \{\alpha_1, \alpha_2, \ldots, \alpha_k\}$ represents the service ranking list of user ur_i and ur_j which are computed by using the service ranking measuere .

Let $EV_i = \{\beta_{i1}, \beta_{i2}, \ldots, \beta_{ik}\}$ and $EV_j = \{\beta_{j1}, \beta_{j2}, \ldots, \beta_{jk}\}$ are ith and jth user qos metrics which are related to selected cloud web services from CWS_{ij}.

The mutual collaborative filtering measure used to find the

$$\mathrm{ProposedSIM} = \frac{\max\{\alpha_u(r(u,i)), \alpha_v(r(u,i))\} \times \sum_{e=1}^{k} (\beta_{ie} - \overline{\beta_i}) \times (\beta_{je} - \overline{\beta_j})}{\sqrt{\sum_{e=1}^{k} (\beta_{ie} - \overline{\beta_i})^2 \times \sum_{e=1}^{k} (\beta_{je} - \overline{\beta_j})^2}}$$

where $\max\{\alpha_u(r(u,i)) * \phi_u(r(u,i)), \alpha_v(r(u,i) * \theta_v(r(v,i)))\}$ represents the maximum cloud service ranking value of uth user and vth user in the CWS list.

β_{ie} : qos value of the uth user

$\overline{\beta_i}$: mean value of all the uth user qos values

β_{je} : qos value of the vth user

$\overline{\beta_j}$: mean value of all the vth user qos values

$$\phi_u(r(u,i)) = \bar{u} + \frac{\sum` u_a \in Sim(u)}{\sum_{u_a \in S(u)} \log(sim((i_a,i)).\frac{\Sigma_{ia} \in Sim(i)\, sim'(i_a,i)(r(u,i_a)-\bar{i}_a)}{\sum_{i_a \in S(i)} sim\,(i_a,i)}}$$

$Sim(u)$: Similarity rank of the uth user web service.
$sim((i_a,i))$: Similarity score of the uth user with assigned service rank i_a
$r(u,i_a)$: QoS rate of the uth user with the assigned service rank i_a.
$\bar{i}_a$: mean of all assigned service ranks to the uth user.

$$\theta_v(r(v,i)) = \bar{v} + \log(\cos(i_a,i)).\frac{\sum_{u_a \in S(u)} sim(v_a,v)(r(v_a,i)-\overline{v_a})}{\sum_{v_a \in S(v)} sim\,(v_a,v)}$$

where $S(u)$ is the similar web service users of user v, $\overline{v_a}$ denotes the average QoS values of web user v_a .