INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

ENDORSE THE TRUSTFUL QoS TO CRAFT MASHUP/BUSINESS ORIENTED WEB SERVICES AND RECTIFY MALICIOUS ATTACKS

[1]V.Devi, [2]R.Kanagaselvi

[1]Student, Department of computer science and engineering, Sree Sowdambika College of engineering, TamilNadu, India
[2]Assistant Professor, Department of computer science and engineering, Sree Sowdambika College of engineering, TamilNadu, India
devi17lakshmi@gmail.com, rkanagaselvi@gmail.com

Abstract: - Web services are standardized way of integrating web based applications. In the recent days, the web domain is augmented with new types of services. Due to the growing number of web services available in online that makes difficulty for user to select an appropriate web services among a large number of services for their business interest. In this paper, we propose a web service recommender system that fit user's interest based on the quality-of-service properties such as response time, throughput, availability and develop a fault based testing method to improve the robustness of the web service when there is a need for the users to rectify the bugs without finding no risk in web services while accessing the service. A malicious attack is also important to get trustworthy QoS properties from the users and it is used to filter the malicious information about the web services.

Index Terms—Web Service, Qos, Fault based testing, Robustness, Malicious Attack, Throughput, Response Time, Availability.

1. INTRODUCTION

Web services is a method of communication between devices over a network; it integrates the web based applications because of each web application provide a specific service so all of the services are integrated within

the enterprise. Web services can convert existing applications into web applications so it is mainly used for business process. Web services are defined by WSDL (Web Service Description Language) and XML (Extensible Markup

Language) is used for exchanging data between devices and SOAP (Simple Object Access Protocol) for exchanging structured information. Web services have a cross-platform, cross-language characteristics so it is mainly used by Both enterprises and individual developers for edifice service oriented applications. When building service oriented applications, developers try to find and reuse existing services to build their process.

The predominance of service oriented architectures and cloud computing leads to enormous amount of services are deployed on web so that reason service users acquire difficulties in finding a preeminent matching service to reuse existing service for their business needs from huge volume of services Currently many of service recommender systems are available but no one recommend the quality of service to end users because more number of the web services under the same category so none of them does not know which one is the quality of the web service.

Since selecting a high quality web service among a large number of web services is a non-trivial errand so many of the developers implement their own service instead of reuse existing service but that takes too much time and resources. Quality-of-Service is widely engaged to represent the non-functional characteristics of web services and also that has been considered as the key feature in service selection. QoS is described as a set of properties such as response time, throughput, availability but response time and availability is measured at the client-side.

A malicious attack is also important to get trustworthiness Qos properties from the users because training users also give the Qos properties. Sometimes user give the malicious details about the web services so it is affect the accuracy of the recommender system. Classification technique is used to filter the malicious information about the web services.

Fault recognition method is also used to improve the quality of services because of some times, users get erroneous service when accessing the service for the reason of server error, error in ping of service etc.., This method is implemented by processing the WSDL that had details about which time which type of errors are facing by user for what reasons.

Collaborative filtering (CF) such as item and user based methods are prevailing technique to endorse the services from overwhelming services but it take too much time to compute similarity between services. A greenful solution is to decrease the number of services that need to be processed in real time. Clustering techniques decrease the data size by consortium ideal services together. Since the number of services in a cluster is much less than the total number of services so significantly reduces the amount of time of CF algorithm. At the same time, ratings of similar services are grouped within a cluster since the recommendation accuracy based on user's rating enhanced.

A Big Table is the repository of Big Data applications. It is designed for storage requirement for service relevant data in distribute and scalable manner. It give the unique key values for every data in repository so easily retrieve and process the tremendous amount of service relevant data. Big table has achieved several goals that are wide applicability, scalability, higher performance and high availability. Big table does not sustain a full relational data model instead of it supports simple data model that supports dynamic control over data layout and format.

The rest of this paper is structured as follows. In section 2 have the back ground techniques of the proposed system. In section 3 covers the analysis of the problems with existing system. In section 4 discusses the description about the concepts of proposed system with their architecture and its techniques. The experimental background and result analysis is in section 5. In section 6 gives the conclusion and future work of this paper.

2. BACK GROUND AND RELATED WORK

Our proposed system utilizes and enhances the following techniques and approaches to endorse the quality of web services for developer's business needs.

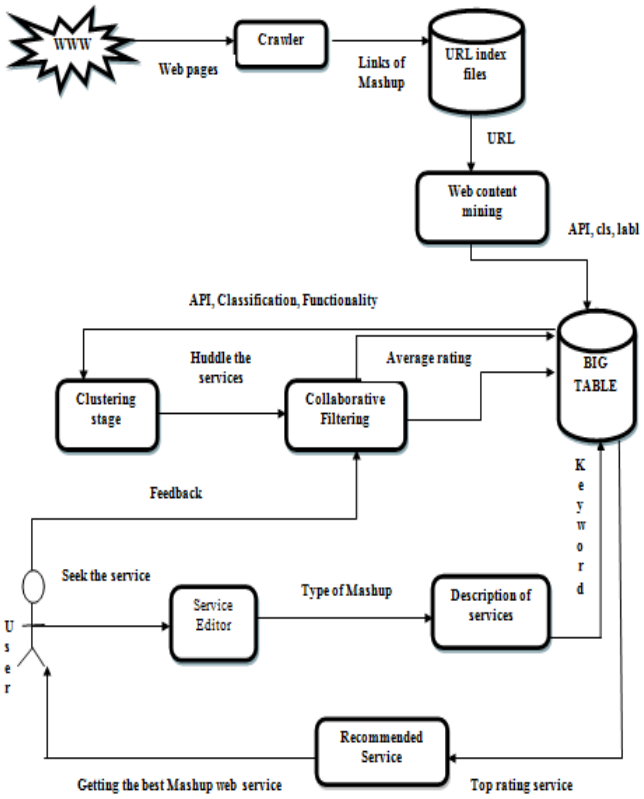

Figure 1: Architecture of background work

Web Crawler is a simple program or automated script that scans through internet pages to create an index of web pages. we crawl all the publicly available Web services from World Wide Web that are all web service relevant web pages' links are stored in to the database. Now, database contains the links of all the web service relevant web pages but each web page contains a lot of web pages that are hyperlinked inside a single web page then it also crawls each hyperlinked web page. The hyperlinked web page's links are also stored into the database.

Web Content Mining is used to extract the content of web services related Web page's links because of all the hyperlinked Web pages have superfluous text, images and also graphical data but we need only label of the web services, relevant APIs, classification, functionality from their links by using this web content mining technique. Web content mining is further divided into Web page content mining and search result mining. Web page content mining is traditional searching of Web pages with the help of content while search results mining is further search for pages found from previous search.

Clustering is the task of discovering homogenous group of data items. It is the partition of a set of data item into subsets. Data clustering is based on the similarity or dissimilarity measurements between data items. In data mining, hierarchical clustering is the most important method of cluster which builds the hierarchy of clusters. Service users have difficulties to finding our desirable services because of tremendously increasing the service relevant data so we will group the similar services based on APIs, functionality, classification by using clustering algorithm.

Generally, cluster analysis algorithms have been used for huge amount of data stored. Clustering algorithms can be either hierarchical or partitional [13]. Partitional algorithms affected from several limitations that are (i) the result of clustering depends on the choice of number of clusters; (ii) cluster size is not monitored during execution of algorithms. In this proposed system, hierarchical clustering algorithm is chosen because of it does not affected by the above partition algorithm limitations.

Our proposed system exercise bottom-up hierarchical clustering. Figure 1 shows bottom-up hierarchical clustering that means presume n number of services, each service is initiated into a single cluster. Then it repeats integrating the similar data items until all of the data items are in one cluster.

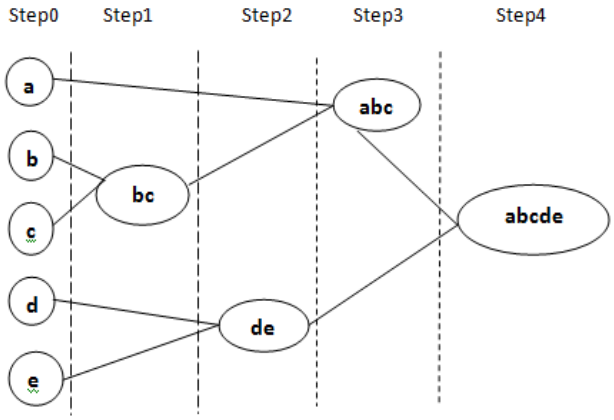

Figure: 2 Bottom-up hierarchical cluster

Algorithm 1: Bottom-Up hierarchical clustering algorithm for service clustering.

Input: A set of services S = {S_1… Sn},
a functionality similarity matrix F = $[f_{i,j}]_{n\times n}$,
Number of clusters K.
Output: $Dendrogram_k$ for k=1 to |S|.
1. Fi= {s_i}, ∀i;
2. $d_{Ci, Cj} = d_{i,j}$, ∀i,j;
3. **for** K= |S| **down to** K
4. $Dendrogram_k = \{F_1,\ldots,F_K\}$;
5. l,m = $\text{argmax}_{i,j}\, d_{Ci,Cj}$
6. F_l = join (F_l, F_m)
7. **for each** $F_h \in S$
8. **if** $F_h \neq F_l$ and $F_h \neq F_m$
9. $d_{fl,fh}$ = Average ($d_{fl,fh}$, $d_{fm,fh}$);
10. **end if**
11. **end for**
12. S=S− {F_m};
13. **end for**

Collaborative Filtering (CF) is a popular machine learning algorithm to recommend the services, which make automatic prediction about the interest of user by collecting preferences or taste information from many users. We also finding the rating of the services by users based on the description, usage of the service and also how many users access this particular service for their Mashup so collaborative filtering provide the accurate predication to endorse, which type of service integration provide the best Mashup.

3. ANALYSIS OF PROBLEM

Service oriented application is a collection of services. These services communicate with each other via data passing using extensible mark-up language. Booming of service oriented applications, huge number of services is available on internet so services users facing many difficulties to finding quality of services because of most of the developers reuse the existing web services for their business needs according to their requirements. Currently many of service recommender systems are available but most of the recommender system mainly used Collaborative filtering to

endorse the service to users but it has difficulties because of increasing the number of services on web and also most of the service recommender system endorse the services based on only the ranking of the services but the ranking was only base on the service's functionality similarities and application programming interface. This type of service recommender system only provide service endorsement based on service similarities and feedbacks about the service but they did not focus on the quality of service and also does not focus on fault recognition method. In this type of service recommender system, sometimes, users get erroneous service when accessing the service for the reason of server error, error in ping of service etc.., and also most of the recommender system does not focus on clustering so they did not produce the accurate endorsement and also This approach did not provide the recommendation to users within the acceptable time because it processes the overwhelming services. Latterly, some recommendation system uses K-means algorithm to cluster the relevant services because of decreasing the amount of available service but this algorithm needs the number of clusters before starting this algorithm

Traditional databases like as a relational databases are used in existing system but more number of services are emerging on the web, such huge volume of service relevant elements are generated and distributed across the network, which cannot be effectively accessed by traditional database management system. And also service relevant data elements are frequently change the structure of service relevant data items so it take too much time to handle the huge volume of data, and often we change the structure of the whole database .

Above problems are overcome by our proposed approach using the section II background approaches, techniques and algorithms. The mainly focusing of our proposed approach is to endorse the quality of web services within an acceptable time to service users to get the quality of service for their business needs. Our proposed approach primarily concentrating to reduced the time complexity to recommend the quality of web services. To enhance the prediction accuracy, we propose a service recommendation engine based on QoS properties and malicious attacks, fault recognition method.

4. PROPOSED METHODOLOGY

The proposed work aims to provide the recommendation of quality of web services. Endorse the quality of web services based on QoS properties, malicious attack and fault recognition method. Our proposed system mainly focuses on recommending the quality of service to service users.

4.1 Quality of Service

To improve the quality of service based on the following QoS properties (1) Response Time (2) Throughput (3) Availability.

4.1.1 Response Time

Response time is the overall time obligatory to complete a service request. It is a composite quality attribute comprises of latency and delay. It is one of the important QoS properties. Let S be a web service. The response time of S is the time duration from the moment service requester send request to the moment the execution result is received, denoted as $q_{RT}(S)$

$$q_{RT}(S) = T_{RT} - T_{ST} \qquad (1)$$

RT - Response Time
T_{RT} – Moment the service S returns the response
T_{ST} – Moment service requestor sends the request

In our recommendation engine, user hit the service that time is recorded by timer and then time is calculated after user getting the data from the server of service.

4.1.2 Availability

In order to maintain the fame and reputation of a website, the quality of service seeming by users, especially the service availability, is a success factor. Let S be a web service. The availability of S is the probability that S can be accessed during a time cycle, denoted as qa(S).

$$q_a(S) = \frac{N_{AS}}{N_{AT}} \qquad (2)$$

N_{AS} – Number of times the service has been accessed
N_{AT} – Number of times the service has been requested

4.1.3 Throughput

Throughput is a vital contributor in QoS rating with higher better behavior. Throughput is defined as total number of completed transactions by web service over time period.

$$Q_{tp}(S) = \frac{N_{AT}}{N_{AS}} \quad (3)$$

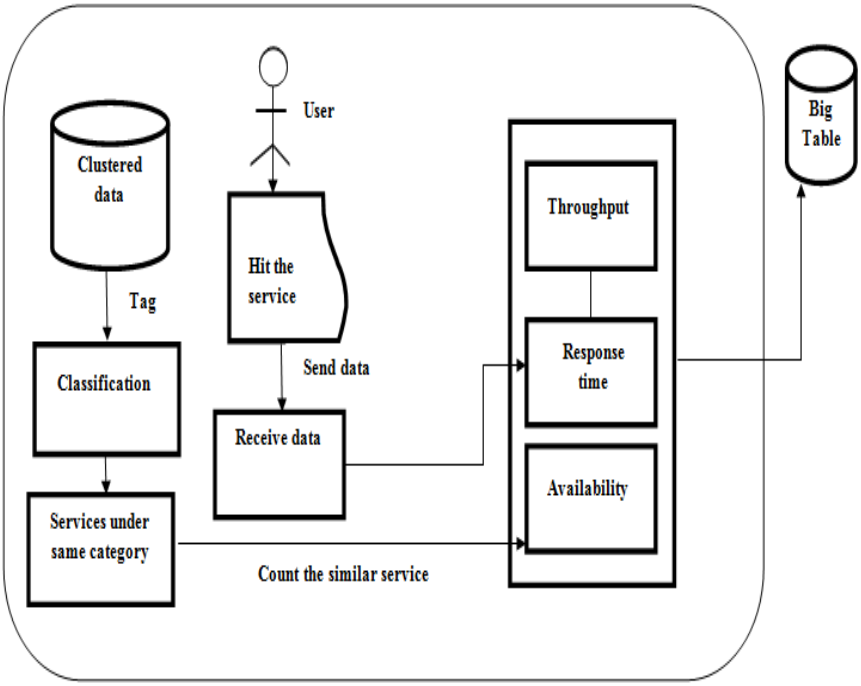

Figure 2: Architecture of method of getting QoS

Algorithm 2: Standardized algorithm for getting QoS value of web service.

For each service S_i in candidate service CS do
I = 1 to m
// calculating QoS value
QoS (S_i) = ($q_{RT}(S)$, $q_a(S)$, $q_{tp}(S)$);
End For
// web services are sort in descending order
For i = 1 to m do
For j = 1 to m do
IF QoS(S_i) < QoS(S_j) then
tmp = QoS(S_i);
QoS(S_i) < QoS(S_j);
QoS(S_j) = tmp;
End IF
End For
End For

4.2.1 Malicious Attack

A malicious attack is also important to get trustworthiness QoS properties from the users because training users also give the Qos properties. Sometimes user give the malicious details about the web services so it is affect the accuracy of the recommender system. Classification technique is used to filter the malicious information about the web services.

A classification based intrusion detection system will classify all the users into either malicious or normal. Classification techniques evaluate and classify the data into known classes. Each data sample is marked with a known class label. Also these techniques are used to learn a model using the training set data sample. This model is used to classify the data samples as anomalous behavior data or the normal behavior data.

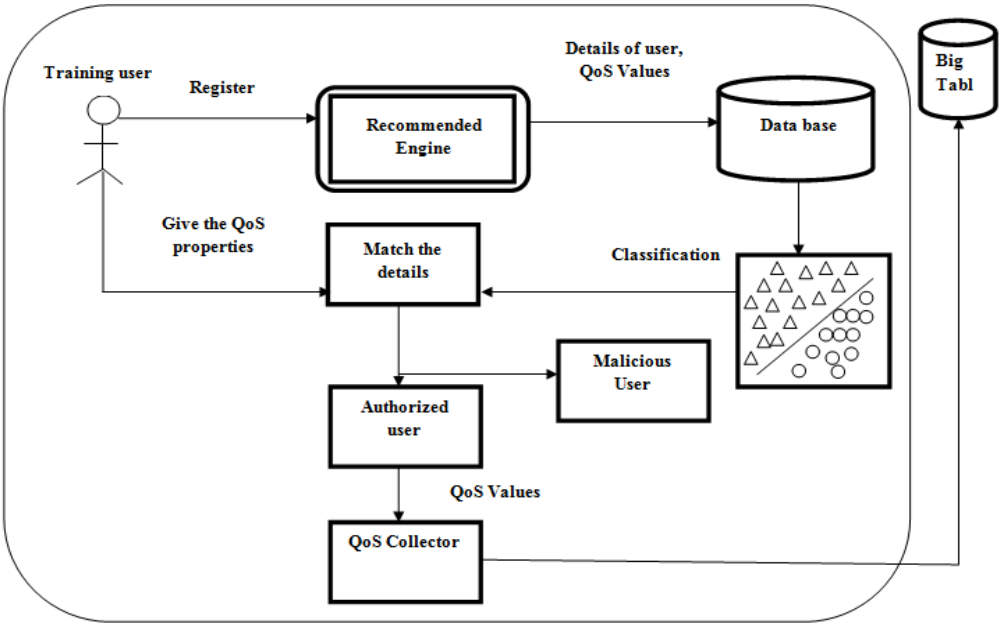

Figure 3: Architecture of malicious attack

Algorithm 3: Apriori algorithm for identifying malicious QoS value

```
Ck: Candidate item set of size k
L1 = {frequent items};
  for (k = 1; Lk !=Æ; k++) do
    Ck+1 = candidates generated from Lk;
      for each transaction t in database do
        increment the count of all candidates in Ck+1    // that are contained in t;
      end for;
    Lk+1 = candidates in Ck+1 with min_support
  end for;
return k U  Lk;
```

4.2.2 Fault Recognition method

It is also used to improve the quality of services because of some times, users get erroneous service when accessing the service for the reason of server error, error in ping of service etc.., This method is implemented by processing the WSDL that had details about which time which type of errors are facing by user for what reasons. Fault recognition method is mainly used to improve the robustness.

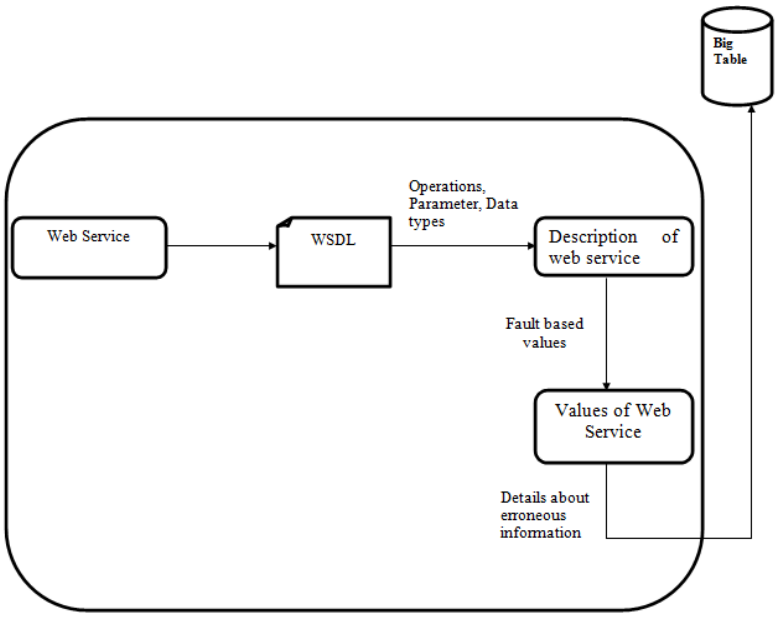

Figure 4: Architecture fault recognition method

Above figure 2, 3, 4's big table is connected in the architecture of background's big table. These all the architectures connected with in one. The above methodologies are used to proposed to give accurately predict the services to service users.

5. EXPERIMENTAL BACKGROUND

To verify our proposed approach, web service relevant data set is used in the experiments. Web service's name, API, category are crawled from the publicly available web sites. Web crawling technique is used to get the web services information. Web content mining is also used in web crawling because of mining the necessary information from huge amount of information. Result of the web crawling and web content mining is shown below.

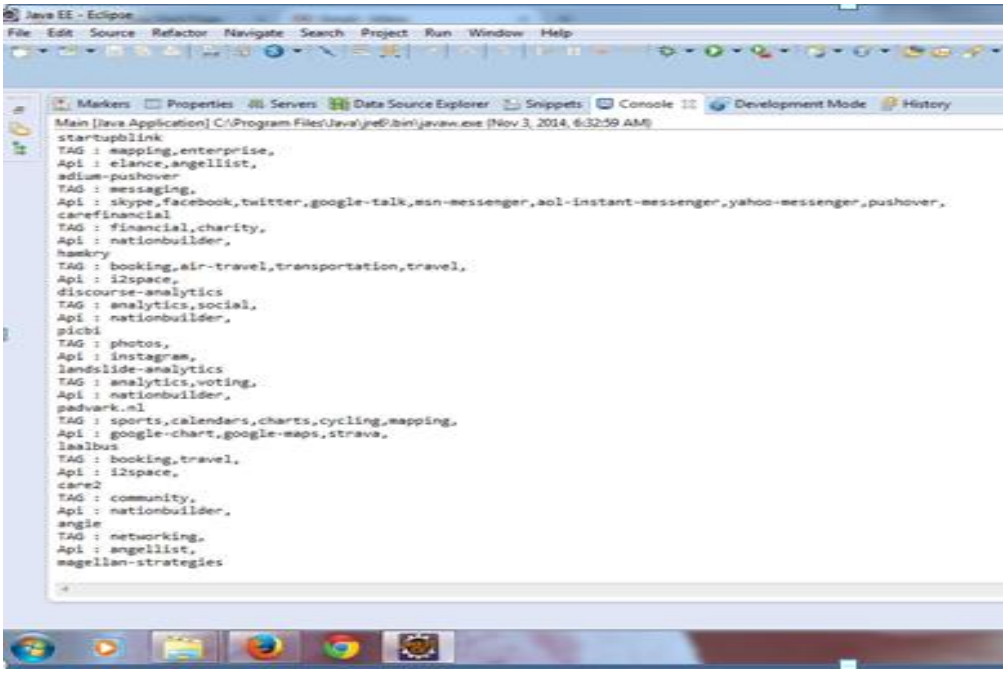

Figure 5: Result of web crawling and web content mining

Then, calculating the QoS properties based on the response time, throughput, and availability formulas. Algorithm 2 is used to getting the QoS value.

Service	Response Time	Availability	Throughput	Success Rate	QoS
S1	47.27	61	6	62	0.6534
S2	63.83	19	1.74	20	0.3924
S3	68.91	19	12	20	0.3856
S4	64.96	18	10	18	0.3637
S5	71.54	18	11	18	0.3550
S6	68.88	17	4	18	0.3455
S7	70.23	17	9	18	0.3437
S8	143.65	19	15.16	19	0.2789
S9	221	18	19	18	0.1562
S10	328.67	21	38	21	0.0668

Table 1: Results of Quality of service of proposed methodology

Table 1 shows the result of calculating the QoS values of web services

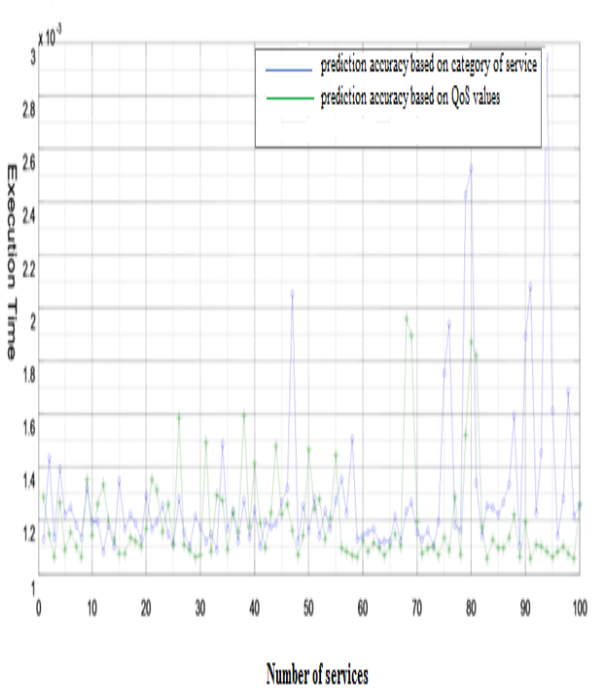

Figure 5: comparison of predication accuracy

The figure 5 shows the comparisons of prediction accuracy of service endorsement. It shows the experimental results of endorse the services to service users based on quality values, malicious attacks, and fault recognition method and also this technique consumes low execution time to give the appropriate services to service users for their business interest.

6. CONCLUSION AND FUTURE WORK

In this paper, we present, endorse the quality of services to service user using the quality values. Malicious attack, fault recognition method also include in this proposed system because of improving the trust worthiness and robustness. In this proposed system provide the following benefits (1) then the identical similarities between relevant services within the same cluster are computed. As the result of number of services in a cluster is much less than that of whole services so recommendation time is less; (2) with the use of QoS value, we quickly retrieve the top ranking services with good results according to user needs. (3) accurately predicate the trustworthiness services to users within the acceptable time These three advantages verified by experiments on real-world data set.

Future research can be done in the following way: In the respect of service similarity, semantic analysis may be performed on the description of services. In this way, services are clustered based on the semantic-similar services. Also we will introduce the provider information of service and study service recommendation from social network and trust aspect.

REFERENCES

[1] Rong Hu and Wanchun Dou, "ClubCF: A Clustering-based Collaborative Filtering Approach for Big Data Application" IEEE Transactions on Emerging Topics in Computing, march 2014.
[2] Xi Chen, Zibin Zheng, R.Lyu, "Web Service Recommendation via Exploiting Location and QoS information" IEEE transactions on parallel and distributed systems, vol.25, no.7, july 2014
[3] Yuqiang Li1, Wei Zhaol and Lei Che2 "Research on the Improvement of Traditional Linear Weighted Algorithm for QoS-based Web Service Selection International Journal of Hybrid Information Technology Vol.7, No.4 (2014), pp.249-258.
[4] Kamini Nalavade, B.B. Meshram, "Finding Frequent Itemsets using Apriori Algorihm to Detect Intrusions in Large Dataset" International Journal of Computer Applications & Information Technology Vol. 6, Issue I June July 2014.
[5] Nuno Laranjeiro, Macro Vieira, Henrique Maderia, "A technique for deploying robust web services" IEEE transactions on services computing, vol.7, no.1, January-march 2014.
[6] V.Devi, R.Kanagaselvi, "An innovative approach to endorse the web services to craft mashup web applications using big table" international journal of embedded and soft computing, issue December 2014.
[7] Buqing, Mingdong Tang, Xing Huang, "CSCF: A Mashup Service Recommendation Approach based on Content Similarity and Collaborative Filtering" international journal of grid and distributed computing, vol.7, No.2 (2014), pp.163-172.
[8] Carsten Radeck, Alexander Lorz, Gregor Blichmann, Klaus MeiBner, "Hybrid Recommendation of Composition Knowledge for End User Development of Mashups" The Seventh International Conference on Internet and Web Applications and Services, 2012.
[9] ProgrammableWeb, http://www.programmableweb.com/, (2014)
[10] Wikipedia, http://en.wikipedia.org/wikilMashup
[11] X. Wu, X. Zhu, G. Q. Wu, et al., "Data mining with big data," IEEE Trans. on Knowledge and Data Engineering, vol. 26, no. 1, pp. 97-107, January 2014.
[12] M. A. Beyer and D. Laney, "The importance of "big data": A definition," Gartner, Tech. Rep., 2012.
[13] Z. Zheng, J. Zhu, M. R. Lyu. "Service-generated Big Data and Big Data-as-a-Service: An Overview," in Proc. IEEE BigData, pp. 403-410, October 2013.
[14] G. Thilagavathi, D. Srivaishnavi, N. Aparna, et al., "A Survey on Efficient Hierarchical Algorithm used in Clustering," International Journal of Engineering, vol. 2, no. 9, September 2013.
[15] F. Chang, J. Dean, S. mawat, et al., "Bigtable: A distributed storage system for structured data," ACM Trans. on Computer Systems, vol. 26, no. 2, pp. 1-39, June 2008.