International Innovative Research Journal of Engineering and Technology

ISSN NO: 2456-1983

International Conference on Emerging Innovation in Engineering and Technology

ICEIET-2017

A Consumption History and QoS based Web Service Ranking Technique

R. Sarala [1], P. Manisha[2], Mukku Vineesha[3], G. Indumathy[4]

[1]Assistant Professor/CSE, Pondicherry Engineering College

[2]Student Member/CSE, Pondicherry Engineering College

[1]sarala@pec.edu

[2]manisha.td21@pec.edu

[2]mukkuvineesha.td23@pec.edu

[2]g.indumathy@pec.edu

Abstract—The number of web services that are available on the internet increases day by day, but the end users get struck in identifying the web services that meet their requirements. Web services are generally ranked and listed by finding how much is a web service functionally similar to the one the user is searching for. But over the time, this methodology failed to satisfy the users due to the large number of web services that became available. Later, approaches like collaborative filtering and ranking based on QoS which were compensatory in nature were adopted. But the results could be made more accurate if consumption history is considered and QoS preferences and QoS constraints were explored at large which is considered in this work. The proposed system ranks the web services by considering the functional relevance, user behavior, QoS and service usage factor. The results from the proposed system were found to be better than the existing ranking systems in satisfying the users.

Keywords— web service ranking, functional relevance, user behavior, QoS, service usage factor.

I. INTRODUCTION

Web services are services which are made available from a business web server for web users or other web-connected programs. These are basically client and server applications that communicate over the World Wide Web's Hypertext Transfer Protocol (HTTP) and are designed to support an interoperable machine-to-machine interaction over the network [1]. They have an interface described in a machine-processable format and other systems interact with the web services in the manner prescribed by their description using simple object access protocol (SOAP) messages. These messages are conveyed using HTTP with an extended markup language serialization in conjunction with other web-related standards. Web services can be used by any application irrespective of the platform in which they are developed. Web service descriptions are provided in a web service description language (WSDL) document, which can be accessed through the Internet using SOAP protocol. The primary aim for web services is to demystify and normalize application interoperability within and across establishments which leads to growth in the operational proficiencies and intimate partner relations [2]. The number of web services available on the internet keeps increasing over the time which confuse the users in selecting the web service that best matches their need. This situation demands in finding a way to help the web users find their web service of interest. One feasible solution in the literature is ranking of web services [2], [3], [4]. Most of the existing ranking techniques use keyword matching and Quality of Service (QoS) factor while ranking. QoS denotes the quality aspects of a web service such as performance, reliability, scalability, availability, etc. These constraints are defined on the QoS, and remaining constraints can be utilized for selecting an optimal service for a requester. For example, a requester can request for an information service with availability of 96%. QoS plays a major role in all service oriented tasks, particularly in the discovery and selection of optimal services [2]. In a situation where multiple services provide similar functionality that can accomplish the user's functional requirement, QoS provides a means for differentiating between the web services. Hence, QoS is an essential factor for choosing an optimal service for requesters. Though these ranking techniques help the users in choosing the web service they are looking for, they do not consider the user satisfaction level and service needs that change dynamically. This motivates to propose a web service ranking technique that incorporates user behavior and their decision while ranking. Ranking the web services by considering the functional relevance and user behavior for better accuracy, QoS factor to cope with the decision of the users that change from time to time and service usage factor to differentiate consumed web services from those web services that are simply selected provides a better user satisfaction level.

The remainder of the paper is organized as follows. Section II reviews the related work. Section III describes the proposed system. Section IV highlights on the experimental setup. Section V discusses the evaluation metrics and results of the proposed work. Section VI concludes the paper. Section VII has the references which were used.

II. RELATED WORK

In a work by G. Kang *et al*, [5] a user explicitly specifies his/her functional interest, i.e. key-words, input, QoS requirement and submits them to the web service discovery system. Then the system matches the user's functional and QoS requirements, and returns web services with the best matching degrees to the user. The service selection scenarios can be classified into two categories. The first category aims to select a set of web services for a composite service driven by a workflow, which is widely studied by existing work on service selection [5], [6], [7]. The second category aims to select a single service for a request, or to select multiple services with the same function for multiple requests from multiple users [8], [9].

Yau et al. [10] propose a QoS-based service ranking approach to help the users to select the service that best satisfies or matches the users' QoS requirements. Chen et al. [n11] recognize the influence of the characteristics of web services QoS. They propose a scalable hybrid Collaborative Filtering (CF) algorithm, which incorporates users' locations to help identify similar users. Zhang et al. [12] observe that the QoS performance of web services is highly related to the service status and network environments which vary against time. They propose a QoS prediction framework to provide time-aware personalized QoS prediction for different service users. Wu et al. present a neighborhood-based CF approach to predict the unknown QoS values for service selection [13]. Some CF based service recommendation approaches employ the matrix factorization theory to improve the accuracy of QoS prediction [14].

CF is widely used for web service ranking techniques. CF algorithms are classified into two categories, memory-based and model-based. Memory-based CF includes user-based and item-based approaches. CF based web service recommendation evolved over the time and started to focus on QoS prediction. Zheng et al. [15] propose a novel hybrid collaborative filtering algorithm for QoS prediction of web services by systematically combining both item-based Pearson Correlation Coefficient (PCC) and user based PCC. Jiang et al. [16] present an improved similarity measurement for users and web services, which takes the personalized characteristics of users and web services into account when calculating the similarity using PCC. Tang et al. [17] recognize the influence of web service locations in web service QoS prediction. A location-aware CF approach is proposed for the web service recommendation. Shao et al. propose a user-based CF algorithm using PCC to compute user similarity. The missing QoS values of a web service can be predicted by considering the corresponding QoS values of web services used by his similar users [18].

Guosheng Kang et al.[19] considers the functional relevance simultaneously with user behavior using CF approach to prevent valid web services from being missed in ranking due to keyword mismatch. It uses a Weighted Additive (WADD) strategy in QoS evaluation, which doesn't produce better results since it is compensatory in nature. This is fixed in the proposed work by finding the QoS score using two approaches. One which prioritizes the QoS constraints and the other that is concerned with the number of QoS constraints that get satisfied. Service usage factor based score also contributes to the final score to prevent both selected as well as consumed web services from getting the same user behavior score.

III. PROPOSED FRAME WORK

The proposed system ranks the web services by considering functional relevance, user behavior, QoS requirement and service usage factor. Fig. 1 represents the architecture of the proposed system.

The end user requests the system for web services by specifying the functional query and the QoS query. QoS preference and QoS constraint together form the QoS query. Once the user keys in the functional and QoS query, the system initially finds the list of all web services that are functionally similar to the web service which the user requests for, with the help of functional query. Keyword frequency is used in computing this functional relevance score. Keywords of a web service operation are abstracted from the descriptions in its associated WSDL file. The keywords specified in the functional query are also extracted and processing is done. The keywords of every web service in the repository are matched against the keywords extracted from the user's functional query. More the number of keywords that gets matched, higher will the score of that web service.

The user similarity score will be calculated for those web services that are found to be functionally relevant to the user expected web service. Retrieving services which were invoked by the users who have a similar taste to the current user would satisfy the current user's request. For calculating the user similarity score log files and functional query are

International Innovative Research Journal of Engineering and Technology
ISSN NO: 2456-1983

used. Keywords used by the users for service retrieval will also be present with other relevant contents in the log file.

The proposed system finds the web services (previously invoked services) that were invoked by the current user till a certain point of time. Then any user who has also invoked those previously invoked web services would have the same taste as that of the current user. The more the number of such services being invoked by a user, the more similar he/she is to the current user. In addition, the keywords saved in the log files helps in doing additional filtering. Users who have invoked the web services with the same keywords as that used by the current user are considered more similar.

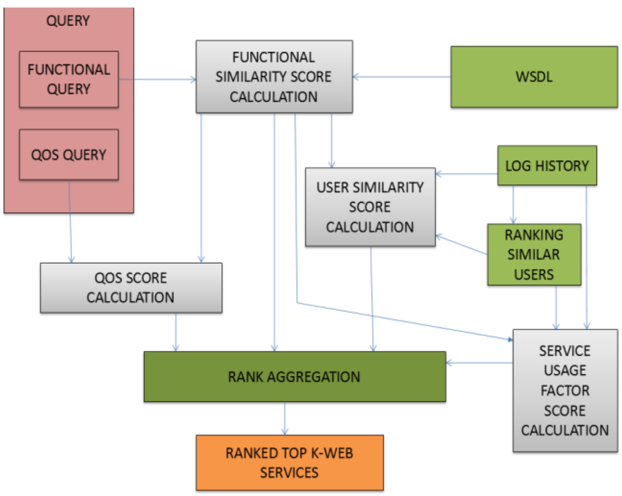


Fig. 1. Proposed System Architecture

The user similarity score is calculated by looking into the log file to find the users who have invoked the web services which were found to be functionally relevant. If those users were ranked as similar users, then the web services invoked by them gets a score corresponding to their similarity rank. If any repetitions of the services are encountered, then those web services that were invoked by the users with a higher similarity score will be considered.

QoS score is calculated with the help of QoS preference and QoS constraint specified by the user. The QoS constraints considered in the proposed system are availability, reliability, throughput, response time and cost. The user specifies the availability and reliability in percentage, throughput in invokes/second, response time in seconds and cost in rupees as per their needs. These values must be normalized for calculation, with the maximum and minimum values they could get which were obtained by evaluating the web services that were created as a prelude. The constraint may belong to either positive or negative criteria. The QoS constraints which are at their best with a maximum value are positive constraints and those which are at their best with a minimum value are negative constraints. Normalization is done using the formulas given in equation 1 and equation 2.

For positive criteria,

$$\text{Normalized Value} = \frac{\text{Actual value} - \text{Minimum value}}{\text{Maximum value} - \text{Minimum value}} \quad (1)$$

For negative criteria,

$$\text{Normalized Value} = \frac{\text{Maximum Value} - \text{Actual Value}}{\text{Maximum Value} - \text{Minimum Value}} \quad (2)$$

Each QoS preference is specified as a number between 1 and 5 for easy understanding of users. They must in turn be converted into weights for calculations, using the formula given in equation 3.

$$W_i = \frac{c_j}{\sum_{i=1}^{n} c_i} \quad (3)$$

where W_i is the weight of the QoS preference of the i^{th} QoS constraint, c_i is the QoS preference of the i^{th} QoS constraint and is the total number of QoS constraints.

Two approaches are used for finding the QoS score. When the user wants the search to be made by giving more preference to QoS constraint, then the QoS score is calculated as follows. Web services are divided into two categories. Web services the satisfy all the QoS constraints fall under Category1 and those web services that satisfy a few QoS constraints fall under Category2. Web services in Category1 will get a score higher than those under Category2. Within each category WADD strategy which finds a summation of the product of each QoS constraint with the QoS weight obtained from the corresponding QoS preference specified is used, which gives the QoS score.

When the users are keen on the number of QoS constraints that get satisfied, then the web services are divided into 5 categories. Web services that satisfy all five constraints fall under category5, and those that satisfy any 4 fall under category4, and those that satisfy any 3, 2, and 1 fall under category 3, 2 and 1 respectively. Web services in Category x will get a score higher than the web services in Category x-1. A category may contain any number of services. Lexicographic (LEX) strategy is used

International Innovative Research Journal of Engineering and Technology

ISSN NO: 2456-1983

for finding the QoS score, wherein the services are ranked first on the QoS constraint which has the highest preference, if there exists a tie, then services are ranked on the QoS constraint which has the next highest preference, and so on till no tie occurs or no more QoS constraints exist.

From the list of services retrieved for a user query, a user may select more than one service. But one or two is going to be consumed by the user. The user behavior score is calculated by considering the invocation history, but not consumption history. Hence, both invoked as well as consumed services get the same user behavior score and it reduces the efficiency of the system. So the proposed system adds the service usage factor score in the final rank which nullifies the drawback of providing the same rank to invoked and consumed services while calculating the user behavior score. This is done similar to user similarity score calculation.

Once the functional relevance score, user similarity score, QoS score and service usage factor score are calculated, the final rank calculation will be done. Each one the computed ranks has different weightage in the final rank calculation which is done using the formula given in equation 4.

$$R = (W_0 * \text{Functional Relevance Score}) + (W_1 * \text{User Similarity Score}) + (W_2 * \text{QoS Score}) + (W_3 * \text{Service Usage Factor Score}) \quad (4)$$

where W0 +W1 +W2 +W3 =1.

The web services are listed in non-increasing order of their final rank as the output to the end user.

EXPERIMENTAL SETUP

Around fifty web services are created for bus ticket reservation. Thousand five hundred user queries were simulated for thirty users. The proposed system is developed in ASP.NET using Visual Studio 2015 and Microsoft SQL 2010.

V. RESULTS ANALYSIS AND COMPARISON

The ranking accuracy of the proposed system is evaluated using Mean Reciprocal Rank (MRR) as a metric. For a list of queries Q, the MRR is the average of the reciprocal ranks of results, calculated using the formula given in equation 5.

$$MRR = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{r_i} \quad (5)$$

where r_i is the position of the web service consumed by the user.

MRR values are calculated for ten users by considering the search results for fifty queries. The comparison of their results for the proposed approaches and the existing system is depicted in Fig. 2 and Fig. 3. From these figures it is evident that the search made by considering service usage factor by giving preference to QoS constraint and, the search made by prioritizing the number of satisfying QoS constraints are more convincing to the users than the existing system. These approaches are found to be 20% and 15% more accurate than the existing system respectively as shown in Fig. 4 and Fig. 5.

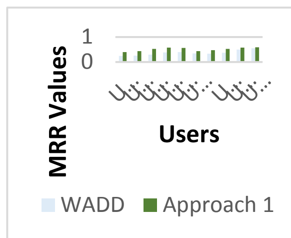
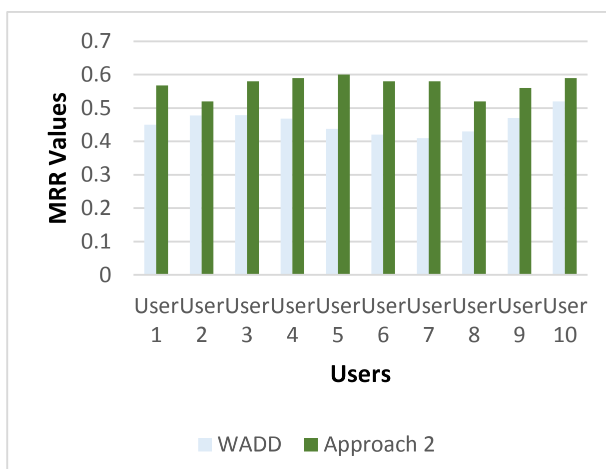


Fig. 2. Comparison between the existing approach and the proposed approach that prioritizes QoS constraints.



International Innovative Research Journal of Engineering and Technology

ISSN NO: 2456-1983

Fig. 3. Comparison between the existing approach and the proposed approach that gives preference to the number of satisfying QoS constraints.

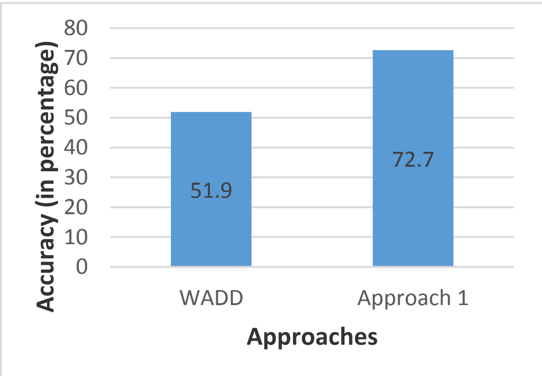


Fig. 4. Accuracy comparison between the existing system and the proposed approach 1.

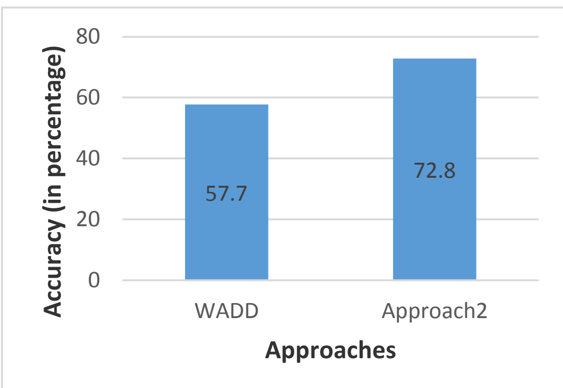


Fig. 5. Accuracy comparison between the existing system and the proposed approach2.

VI. CONCLUSION AND FUTURE WORK

In the proposed work, web services are ranked by considering functional relevance, user behavior, QoS and service usage factor. The consumed services are ranked higher than the selected services with a novel approach where the service usage factor is taken into account. QoS score is calculated by categorizing them based on the number of constraints that get satisfied and then applying WADD/LEX approach. The proposed system ranks the web services with more accuracy that better satisfies the users.

VII. REFERENCES

[1] http://searchmicroservices.techtarget.com/definition/Web-services-application-services, last updated in March 2007

[2] Tajudeen Adeyemi Ajao, Safaai Deris, Isiaka Adekunle Obasa,"QoS-based Web Service Selection Using Filtering, Ranking and Selection Algorithm" in *International Journal of Scientific & Engineering Research, Volume 4, Issue 7, July-2013-184*

[3] Vuong Xuan TRAN and Hidekazu TSUJI, " QoS based Ranking for Web Services: Fuzzy Approaches", in *4th International Conference on Next Generation Web Services Practices, 2008 IEEE DOI 10.1109/NWeSP.2008.41*

[4] Yanan Hao, Yanchun Zhang, Jinli Cao, "Web Service Discovery And Rank: An Information Retrieval Approach" ,in *Future Generation Computer Systems, Volume 26, Issue 8, October 2010, Pages 1053-1062*

[5] Kang *et al.*, "An effective dynamic web service selection strategy with global optimal QoS based on particle swarm optimization algorithm," in *Proc. IEEE 26th Int. Parallel Distrib. Process. Symp. Workshops PhD Forum (IPDPSW)*, 2012, pp. 2274–2279.-4

[6] L. Qi *et al.*, "Combining local optimization and enumeration for QoS- Web service composition," in *Proc. IEEE Int. Conf. Web Serv.*, 2010, pp. 34–41.-11

[7] M. Alrifai, D. Skoutas, and T. Risse, "Selecting skyline services for QoS-based web service composition," in *Proc. Int. World Wide Web Conf.*, 2010, pp. 11–20.

[8] G. Kang *et al.*, "Web service selection for resolving conflicting service requests," in *Proc. Int. Conf. Web Serv.*, 2011, pp. 387–394-5

[9] W. Dou *et al.*, "A QoS-aware service evaluation method for co-selection a shared service," in *Proc. IEEE Int. Conf. Web Serv.*, 2011, pp. 145–152.-7(4)

[10] S. S. Yau and Y. Yin, "QoS-based service ranking and selection for service-based systems," in *Proc. Int. Conf. Serv. Comput.*, 2011, pp. 56-63

[11] X. Chen *et al.*, "RegionKNN: A scalable hybrid collaborative filtering algorithm for personalized Web service recommendation," in *Proc. IEEE Int. Conf. Web Serv.*, 2010, pp. 9–16

[12] W. Lo *et al.*, "Collaborative web service QoS prediction with location-based regularization," in *Proc. IEEE Int. Conf. Web Serv. (ICWS)*, 2012, pp. 464-471

[13] J. Wu et al, "Predicting quality of service for selection by neighborhood based collaborative filtering," in *Proc. IEEE Trans. Syst. Man Cybern. Syst., vol. 43, no. 2, pp 428-439, Mar. 2013*

[14] Y. Xu *et al.*, "Personalized location-aware QoS prediction for web services using probabilistic matrix factorization," in *Proc. Web Inf. Syst. Eng. (WISE)*, 2013, pp. 229–242

[15] Z. Zheng *et al.*, "WSRec: A collaborative filtering based web service recommender system," in *Proc. IEEE Int. Conf. Web Serv.*, 2009, pp. 437-444

[16] Y. Jiang *et al.*, "An effective Web service recommendation based on personalized, collaborative filtering," in *Proc. IEEE Int. Conf. Web Serv.*, 2011, pp. 211–218.

[17] M. Tang *et al.*, "Location-aware collaborative filtering for QoS-based service recommendation," in *Proc. IEEE Int. Conf. Web Serv.*, 2012, pp. 202-209

[18] L. Shao *et al.*, "Personalized QoS prediction for web services via collaborative filtering," in *Proc. IEEE Int. Conf. Web Serv.*, 2007, pp. 439–446.

[19] Guosheng Kang, Jianxun Liu, Mingdong Tang, Buqing Cao, *Member, IEEE,* and Yu Xu,"An Effective Web Service Ranking Method via Exploring User Behavior", *IEEE Transactions on network and service management, vol. 12, no. 4, December 2015.*

Author Biography

R. Sarala

R. Sarala is currently working as an Assistant Professor in the Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India. She has a teaching experience of 14 years and has published nearly fifteen papers in reputed International Journals and Conferences. Her areas of interest include Computer networks, Soft Computing and Information Security.

P. Manisha

She is pursuing her B. Tech degree in the Department of Computer Science and Engineering in Pondicherry Engineering College from Pondicherry University.

Mukku Vineesha

She is pursuing her B. Tech degree in the Department of Computer Science and Engineering in Pondicherry Engineering College from Pondicherry University.

G. Indumathy

She is pursuing her B. Tech degree in the Department of Computer Science and Engineering in Pondicherry Engineering College from Pondicherry University.