International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/IJSRCSEIT

To Innovative Approach to Personalized Web Service Level for Selecting the Optimal Web Service Applicant

[1]Aishvery Rathore, [2]Prof. Vinod Mahor

[1]Research Scholar, Department of Computer Science Engineering, Millennium Institute of Technology and Science, Bhopal, India

[2]Assistant Professor, Department of Computer Science Engineering, Millennium Institute of Technology and Science, Bhopal, India

Article Info

Publication Issue :
Volume 8, Issue 6
November-December-2022

Page Number : 476-483

Article History
Accepted: 01 Dec 2022
Published: 15 Dec 2022

ABSTRACT

Web service composition enables developers to build apps. It is widely known that choosing relevant Web services for a composite service that satisfies the developer needs may be difficult and complicated, especially given the Internet's rapidly expanding supply of Web services. The Quality of Service (QoS) is just used as a fundamental criterion to drive the selection process in the bulk of current methods to Web services selection, As a common software development technique, web service composition enables the creation of sophisticated Mashups by skillfully fusing Web services with various functionality. To choose the right Web services to create Web applications that meet functional requirements, however, gets more difficult as the number of Web services grows. A composition pattern aware Web service recommendation approach named EWACP-Deep FM is developed to take user preferences into account when recommending Web services. This method combines the co-occurrence and popularity of Web services with composition patterns between Web services and Mashups. By creating a multidimensional feature matrix, which the depth factorization machine (Deep FM) model then uses to train itself, it is possible to identify potential link relationships between. In this dissertation work based on Web services and Mashup applications and to suggest the Top-N best services for the intended Mashup application. Tests utilizing actual datasets from Programmable Web demonstrate that the suggested strategy works better than others with improved suggestion efficacy and Results from the experiments demonstrate that both strategies outperform those that might be used in a rigorous experimental setting.

Keywords : Web Service, Multi-criteria decision making, quality of service and design, service based software system.

Copyright: © the author(s), publisher and licensee Technoscience Academy. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License, which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited

I. INTRODUCTION

SOA is a way of representing a model which constitutes the logic for automation and is distributed in different and tiny units of logic. When aggregated these small units form a bigger, automated business goals and individually these units can be distributed. Web service is the basic building block of SOA. Though, SOA allows these smaller units to exist independently but not in isolation from others. This smaller unit which pertains logic in them has to adhere some principles which allow them to exist independently and provides some similarity and standardization. Similar to the object-orientation, there are some principles involved that are to be followed in service-orientation [1,2]. Figure 1 shows the design issues that are addressed by service-orientation.

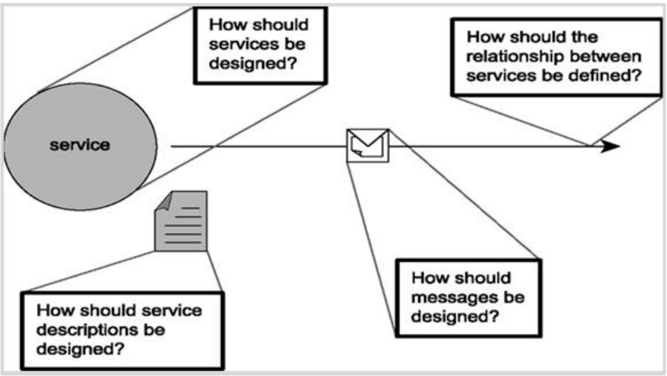

Figure 1 : Design Issues Addressed by Service-Orientation.

The principles that a web service maintains are loose coupling, Discoverability, Service contract, Statelessness, Autonomy, Compensability, Reusability and Abstraction [3-4]. As an architectural model, SOA is defined among three basic components as shown in Figure 2.

(a) Service Provider: It deposits definition of web service candidates in repository.

(b) Service Broker: It stores the definition of services and is a registry of services.

(c) Service Consumer: It explores the repository with service definition to find the needed service candidate.

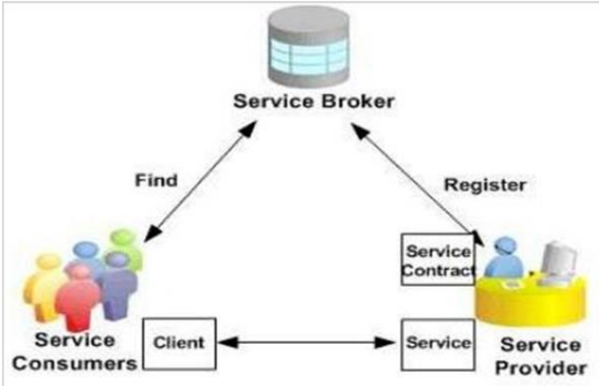

Figure 2 : Service Oriented Architecture

II. LITERATURE REVIEW

A good number of Innovative models have been proposed in past by various authors, for software systems, e.g., Putnam's model, Musa's execution time model, etc. Moreover, many approaches are proposed for Innovative of reliability of SOS and component based systems. But, most of these assume that the component reliability is known and focused on system level reliability Innovative [5-6]. However, in this work Innovative of both component/web service and the whole system is considered. Though, few approaches consider the reliability Innovative at component level, they are mainly built for traditional systems. Both of the approaches need content of the component for reliability Innovative [7-8]. Table 2.2 shows the methodologies proposed by different authors for the Innovative of software system's reliability.

Different from the work done, this work predicts the reliability of web services which are accessed remotely and hosted by different organizations. Moreover, this work also considers the effect of communication links in Innovative of reliability for component/web services. Z. Zheng et. al. proposed a framework for reliability Innovative of SOS but the proposed approach performs better when extensive experiments are conducted. Proposed approach uses the collaborative filtering technique for reliability Innovative of service oriented system [9].

Non-functional characteristics of the software system and components/web services are presented widely by employing the QoS. The performance of non-functional QoS of SOS can be evaluated either from web service provider's perspective, e.g. availability, price. etc. or from client's (designer) perspective, e.g. Throughput, Response time, etc. Various approaches have been proposed in past on the basis of QoS performance for the selection of component, which solves the problem for selection of optimal candidate from a set of equi-functional candidates. For the selection of best performing candidates for SOS, in this work, web service quality ranking framework is proposed using collaborative filtering. Table 1 shows the Methodologies proposed by different authors for the selection of web service/component for software system.

Table 1 : Web service-component selection for software system

AUTHORS	METHODOLOGY
P. A. Bonatti et. al.(2021)	Based on individual criteria, three different kinds of service selection problems are formalized and proved that optimal service selection is hard for one-time costs and when the cost is ignored then the selection problem can be solved in polynomial time.
T. Yu et. al. (2020)	To make the service selection easier, broker based architecture is designed and modeled the service selection problem as graph model and combinatorial model.
L. Zeng et. al.(2018)	Two approaches are proposed and compared: first in on the basis of local selection and second is on the basis of allocation.
L. N. Liu et.al. (2017)	A collaborative filtering approach is proposed to rank items by modeling user preferences.
C. Yang et. al.(2014)	A recommender system is proposed that uses the ranking oriented collaborative filtering approach.
R. C. Cheung (2013)	To calculate the reliability of software system a user- oriented reliability figure of merit is defined based on user environment. How user profiles, which includes the characteristics of the system's users, change the reliability of system.
S. S. Gokhale et. al.(2012)	A hierarchical model is developed for the Innovative of reliability on the basis of the system's architecture. Reliability provided by composite model and reliability predicted by this model is very closer.
K. G.-Popstojanova et. al.(2011)	It describes the scenario to the assess the reliability for components based system and explains how it can be used from design to deployment stage.
Esra Aytaç Adalı and Ayşegül Tuş Işık, 2016	"A Case Study For Forecasting Denizli City Manufacturing Industry Export Data Using Artificial Neural Networks"
Nhien Pham Hoang Bao, Shuo Xiong, and Hiroyuki Iida, 2017	"Swiss system, single elimination, and double elimination. The research shows each tournament structure's advantages and disadvantages".
Bharathan, Chandrasekh	"Ranking accuracy is examined based on the effectiveness of the

aran Rajendran, and R. P. Sundarraj, 2017	algorithm for the ranking job, taking into account the analogies between the sorting algorithm and the ranking process. This essay also examines well-known tournament formats as round robin,"
John Estdale. 2016	"product certification at last. In Proceedings of the Systems Quality Conference"

III. RESEARCH OBJECTIVE

Innovative of the web service's reliability and to propose some compositional style for Innovative of SOS's reliability based on the data observed by other users in past [11]. Selection of the best performing web service candidate from the set of functionally.

- Firstly, a framework for the Innovative of SOS's reliability is proposed, in which the reliability for current user is predicted by using the failure data of other similar users from past.
- Secondly, in this work a framework for web service quality ranking is proposed to get the personalized web service ranking for selecting the optimal web service candidate

The key concern in software engineering states that analyzing the quality of software at implementation phase is too late. Design decisions are to be made at design phase only [12].

Finding the problems at implementation phase requires re-engineering and is very costly. Hence, the quality parameters are to be identified during design phase only. This work is focused on reliability attributes of quality.

The above discussion states that the reliability is to be "built into" the system at design phase. Many approaches focused on reliability Innovative both for component based system or for SOS but all these approaches focused on the system level Innovative of reliability and assumed that the reliability of individual component/web service is known [13].

But, this assumption is not reasonable. It is not clear in these approaches that how the reliability of a component/web services will be obtained. Further sections will explain as to how the reliability of a web service/component can be predicted and how these predicted reliability values can be integrated to predict the reliability of the whole SOS. The proposed approach is also applicable for previous approaches where reliability of component/web service is assumed to be known [14].

(a) Framework Of Service Oriented System

Service oriented system is an integration of different services (here, web services) which are organized in a proper manner to achieve a business goal. In this work, failure probability is the fraction of invocations failed in contrast with total invocations made for a particular component which lies between '0' to '1'.

Figure 3 explains the service flow in SOS. This service flow includes some tasks (T1; T7) which are abstract in nature, pipelined to each other and uses some structures which control their execution like looping, sequential, parallel operations, etc[15]. Each abstract

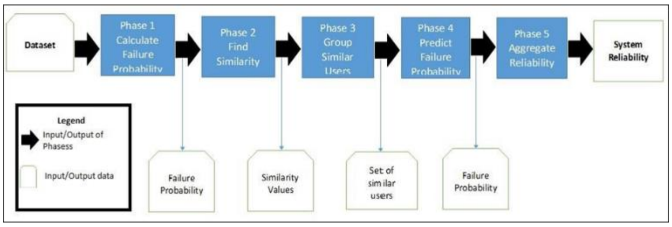

Figure 3: Procedure For Innovative Of Reliability for SoS

IV. PROPOSED METHODOLOGY

Service flow in SOS includes tasks to perform functionalities. For every task a best performing service needs to be chosen from a functionally equivalent candidates set. The easiest and straightforward way is by evaluating every candidate and choosing the optimal one from those candidates. Calculate Failure Probability: In this phase failure probability of every web service candidate 'x' is calculated with respect to every service user's'.

$$f_{s,x} = \frac{I_{fx}}{I_x}$$

Find Similarity: In this phase PCC can be applied for finding the similarity between the current user and other users. PCC has been employed in much recommender system, to calculate the similarity between two users.

$$\mathrm{Sim}_{pcc}(s_1, s_2) = \frac{\sum x \in X_{s1} \cap S_{s2}(f_{s1,x} - \bar{f}_{s1})(f_{s2,x} - \bar{f}_{s2})}{\sqrt{\sum x \in X_{s1} \cap S_{s2}(f_{s1,x} - \bar{f}_{s1})^2}\sqrt{\sum x \in X_{s1} \cap S_{s2}(f_{s2,x} - \bar{f}_{s2})^2}}$$

Parallel: In this type of structure every branch executes at least once and all branches execute parallel. If a single branch fails in execution, the whole structure get fails [16-18].

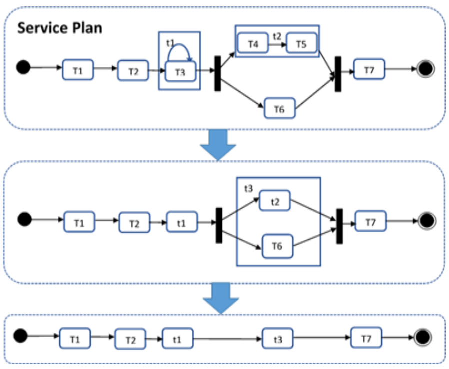

Figure 4 : Failure Probability Aggregation

The Quality ranking framework (named as 'Regressive Rank') for web services is proposed, which is a four-phase process as shown in Figure 4.

In phase 1, average values of QoS attributes are calculated (e.g. Response time, Throughput) which are already known to the user.

In phase 2, two users similarity is calculated on the basis of the rankings given to web services which are commonly invoked by them.

In phase 3, a group of similar users are selected on the basis of similarity values obtained.

In phase 4, final rankings are predicted for both employed and unemployed web service candidates based on the QoS data received from other users using Polynomial Regression (PR) technique [18-23].

V. RESULT AND ANALYSIS

The proposed framework is evaluated using WS-DREAM 1 dataset of web services which include QoS values. This dataset is a real world dataset which include QoS performance of 1.5 million real world web service invocations.

There are 100 web services which are invoked by 150 service users. Each web service is invoked by about 100 times by each user.

The QoS values observed by 150 users for 10 web services is represented by a matrix of 150 _ 100 dimension where each entry is a vector representing the QoS value which can be calculated by using the Equation 2. Response time and throughput are taken as QoS value in experiment to rank web services.

Table 3 : Innovative performance accuracy when sim_{pcc} valuesare between 0.5 to 1

Metrics	Methods	Training Users = 150					
		Sim Values = 0.5 to 1					
		100	110	120	130	140	150
MAE	Linear Regression	0.0042	0.0054	0.0048	0.0059	0.0042	0.0047

	Polynomial Regression	0.0163	0.0181	0.0175	0.0179	0.0186	0.0199
RMSE	Linear Regression	0.0073	0.0068	0.0091	0.0079	0.0089	0.0084
	Polynomial Regression	0.0494	0.0238	0.0236	0.0166	0.0157	0.0137

Table 4.2 : Innovative performance accuracy when Simpcc Values are between 0.1 To 1

Metrics	Methods	Training Users = 150					
		Sim values = 0.1 to 1					
		100	110	120	130	140	150
MAE	Linear Regression	0.0058	0.0101	0.0101	0.0114	0.0105	0.0112
	Polynomial Regression	0.0167	0.0171	0.0139	0.0163	0.0120	0.0132
RMSE	Linear Regression	0.0086	0.0176	0.0173	0.0213	0.0163	0.0193
	Polynomial Regression	0.0171	0.0192	0.0312	0.0247	0.0910	0.0320

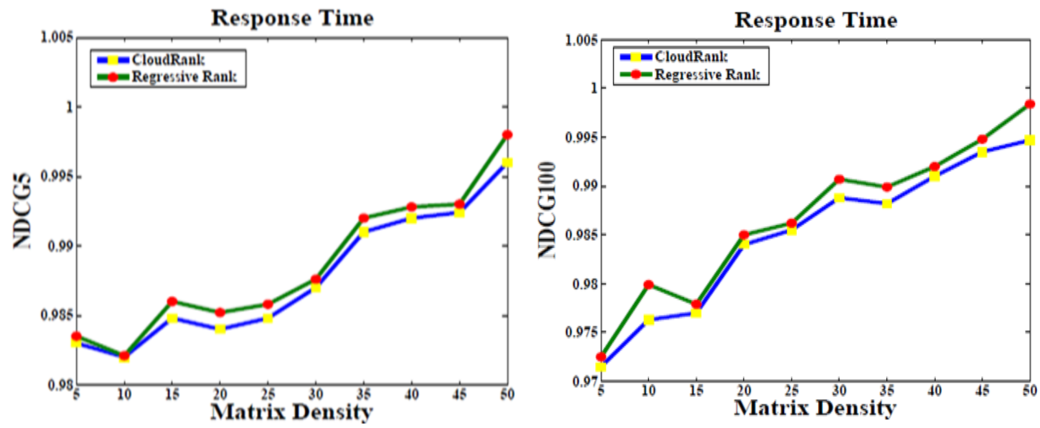

Figure 3: (a) Response time of between NDCG5 and MD. (b) Response time of between NDCG100 and MD

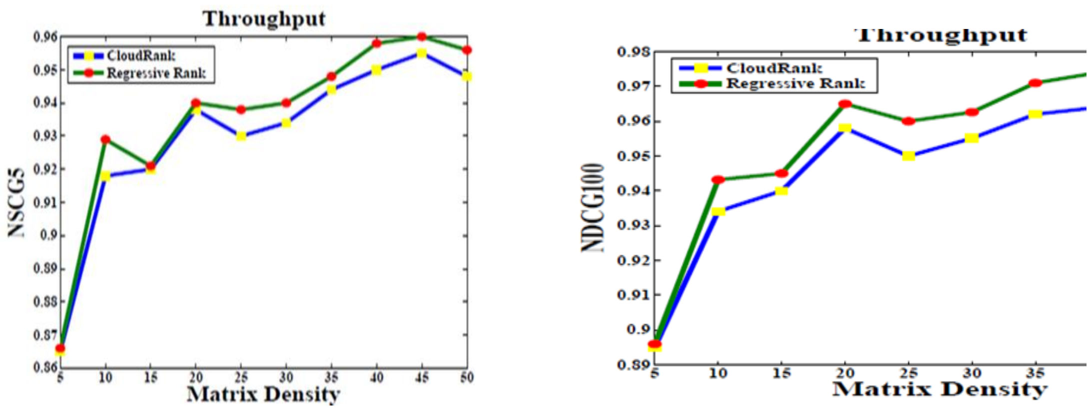

Figure 4: (a) Throughput of between NDCG5 and MD. (b) Figure 6: Throughput time of between NDCG100 and MD

VI. CONCLUSION AND FUTURE WORK

In this Paper, an approach for the Innovative of reliability for SOS is proposed. The idea is to use the data observed by users in past, for predicting the probability of failure of individual web services for the current user. The predicted failure probabilities of the individual web services are then aggregated according to the service flow in a system using some compositional structures for the Innovative of reliability for SOS. The problem of optimal web service selection is also identified during this work. Hence, a framework for the web service quality ranking on the basis of QoS values is proposed, which does not require any extra invocation of the web service. Experimental results show that both the approaches out performs the approaches available under rigorous experimental setup.

To predict the SOS reliability, web service failure probabilities are integrated. While integrating, it is assumed that web service failure probability is independent and does not affect the performance of other web services. In large number of cases, this assumption works because web services are deployed in different servers by different organizations.

In Future; the physical isolation of web service ensures that the failure probability of web service is independent. But, in few exceptional cases, web services' failure is correlated (when two web services are deployed at same server, error propagation, etc.). In future, this issue will be addressed. For web service selection we will analyze some more techniques for the improvement in ranking accuracy (like, matrix factorization, data smoothing, random walk, etc.). We will also plan to analyze the combination of different QoS parameters (the proposed approach ranks different QoS parameters independently) for this purpose.

VII. REFERENCES

[1]. M. R. Lyu et al., Handbook of software reliability engineering. IEEE computer society press CA, 2021, vol. 222.

[2]. L. H. Putnam and W. Myers, Measures for excellence: reliable software on time, within budget. Prentice Hall Professional Technical Reference, 2019.

[3]. J. D. Musa, A. Iannino, and K. Okumoto, Software reliability: measurement, Innovative, application. McGraw-Hill, Inc 2012.

[4]. R. C. Cheung, "A user-oriented software reliability model," Software Engineering, IEEE Transactions on, no. 2, pp. 118–125, 1980.

[5]. S. S. Gokhale and K. S. Trivedi, "Reliability Innovative and sensitivity analysis based on software architecture," in Software Reliability Engineering, 2002. ISSRE 2003. Proceedings. 13th International Symposium on. IEEE, 2002, pp. 64–75.

[6]. K. Goševa-Popstojanova and K. S. Trivedi, "Architecture-based approach to reliability assessment of software systems," Performance Evaluation, vol. 45, no. 2, pp. 179–204, 2001.

[7]. S. M. Yacoub, B. Cukic, and H. H. Ammar, "Scenario-based reliability analysis of component-based software," in Software Reliability Engineering, 1999. Proceedings. 10th International Symposium on. IEEE, 1999, pp. 22–31.

[8]. V. Grassi and S. Patella, "Reliability Innovative for service-oriented computing environments," Internet Computing, IEEE, vol. 10, no. 3, pp. 43–49, 2006.

[9]. L. Cheung, R. Roshandel, N. Medvidovic, and L. Golubchik, "Early Innovative of software component reliability," in Proceedings of the 30th international conference on Software engineering. ACM, 2008, pp. 111–120.

[10]. K. Goseva-Popstojanova, A. Hassan, A. Guedem, W. Abdelmoez, D. E. M. Nassar, H. Ammar, and

A. Mili, “Architectural-level risk analysis using uml,” Software Engineering, IEEE Transactions on, vol. 29, no. 10, pp. 946–960, 2003.

[11]. R. H. Reussner, H. W. Schmidt, and I. H. Poernomo, “Reliability Innovative for component- based software architectures,” Journal of systems and software, vol. 66, no. 3, pp. 241–252, 2003.

[12]. Z. Zheng and M. R. Lyu, “Collaborative reliability Innovative of service-oriented systems,” in Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering- Volume 1. ACM, 2010, pp. 35–44.

[13]. T. Yu, Y. Zhang, and K.-J. Lin, “Efficient algorithms for web services selection with end-to-end qos constraints,” ACM Transactions on the Web (TWEB), vol. 1, no. 1, p. 6, 2007.

[14]. P. A. Bonatti and P. Festa, “On optimal service selection,” in Proceedings of the 14th international conference on World Wide Web. ACM, 2005, pp. 530–538.

[15]. L. Zeng, B. Benatallah, A. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, “Qos-aware middleware web services composition,” Software Engineering, IEEE Transactions on, vol. 30, no. 5, pp. 311–327, 2004.

[16]. R. Burke, “Hybrid recommender systems: Survey and experiments,” User modeling and user-adapted interaction, vol. 12, no. 4, pp. 331–370, 2002.

[17]. N. N. Liu and Q. Yang, “Eigenrank: a ranking-oriented approach to collaborative filtering,” in Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2008, pp. 83–90.

[18]. C. Yang, B. Wei, J. Wu, Y. Zhang, and L. Zhang, “Cares: a ranking-oriented cadal recommender system,” in Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries. ACM, 2009, pp. 203–212.

[19]. Z. Zheng, H. Ma, M. R. Lyu, and I. King, “Wsrec: A collaborative filtering based web service recommender system,” in Web Services, 2009. ICWS 2009. IEEE International Conference on. IEEE, 2009, pp. 437–444.

[20]. K. Järvelin and J. Kekäläinen, “Cumulated gain-based evaluation of ir techniques,” ACM Transactions on Information Systems (TOIS), vol. 20, no. 4, pp. 422–446, 2002.

[21]. P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “Grouplens: an open architecture for collaborative filtering of netnews,” in Proceedings of the 1994 ACM conference on Computer supported cooperative work. ACM, 1994, pp. 175–186.

[22]. Z. Zheng, Y. Zhang, and M. R. Lyu, “Cloudrank: A qos-driven component ranking framework for cloud computing,” in Reliable Distributed Systems, 2010 29th IEEE Symposium on. IEEE, 2010, pp. 184–193.

[23]. Waseem Ahmed, Yongwei Wu,WeiminZheng, ”Response Time Based Optimal Web service Selection”, VOL. 26 NO. 2, FEBRUARY 2015

Cite this article as :

Aishvery Rathore, Prof. Vinod Mahor, "To Innovative Approach to Personalized Web Service Level for Selecting the Optimal Web Service Applicant", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 6, pp. 476-483, November-December 2022.
Journal URL : https://ijsrcseit.com/CSEIT228675