Ranking of Services for Reliability Estimation of SOA System Using FUZZY TOPSIS Method

Neha Singh[1], Dharmendra Singh[2], Kirti Tyagi[3]

[1]Raj Kumar Goel Institute of Technology Ghaziabad, India
[2] Dr. K.N. Modi Institute of Engineering and Technology, India
[3]INHA University Tashkent, Uzbekista, Uzbekistan

Abstract

In software system, there are different quality attributes like reliability, flexibility, reusability and maintainability. Among these quality attributes, reliability is one of the important parameters. Reliability can successfully complete a given system task in a specific environment in specific time period. Reliability estimation is an essential task, which makes users very selective while selecting software for purchase and use. Therefore, estimation of software reliability is necessary and beneficial. In this paper, an approach to estimate the reliability for most often used software system, i.e. Service-Oriented Architecture System (SOAS) has been used. A fuzzy logic technique for order preference by similarity to ideal solution (TOPSIS) has been employed in fuzzy environment to estimate the systems' reliability. Fuzzy logic is used to remove the uncertainty associated with ranking of alternative services and to select the best alternative services in existing algorithms. At first, we have selected four factors which affects the reliability of SOA system.Secondly, we have ranked the services by using Fuzzy technique for order preference by similarity to ideal solution (TOPSIS). In terms of triangular fuzzy numbers, the weight criteria and rating of alternatives has been assessed by linguistic variables that provide the ranking of services to estimate the SOA reliability.

1. Introduction

SOA consists of discriminated characteristics because of its software architecture. [14] The idea of SOA came out in early 1980s and turned out an important architectural style specifically after the creation of web services.

The early era of 1970s was the period of procedure oriented language, a problem was divided into small problems, and writing procedures for those small problems, and then all the procedures were putted together to make a solution of complete problem as show in Figure 1. However, this type of approach was not capable to relate the real-world problems. The concept of object oriented gave birth to object-oriented language.

Now the task was turned into much simpler to address the real-world problems. As the technology is progressing very fast, the expectations and needs ofpeople are also increasing gradually. [11] Has proposed that in the past, clients' requirements were very easy and could be fulfilled by only one or two handful of services. On the other hand, presently, the clients' requirements are increasing rapidly and changes aggressively. Due to this, one or two services are not sufficient to develop a business process. There is a requirement for business process, as elegant services that can be loosely coupled by the beginning of target architecture as given in [16].

This paper is organized in following sections; section I contains a brief introduction of SOA and reliability, section II contains literature survey related to different fields as Architectural based models, Mathematical models and soft-computing models. Section III elaborates the details of fuzzy TOPSIS based approach. Section IV and V contains applications of fuzzy TOPSIS for component ranking and case study respectively. Section VI concludes the result and in last section conclusion, Limitations and future directions are discussed.

To estimate the software cost and quality, one among the important factors is software reliability. IEEE defines software reliability as "the probability of a software system for performing its intended functions under the specified operating conditions over the specific period of time". In other words, it can also be defined as, "The probability of failure free software operation in a particulate environment for a specific period of time".

The most essential aspect of software quality is software reliability. Reliability Engineering concepts are also applied in the field of software. The quantitative study of operational behaviour of software-based system is Software Reliability Engineering (SRE), corresponding to users' requirement concerning reliability.
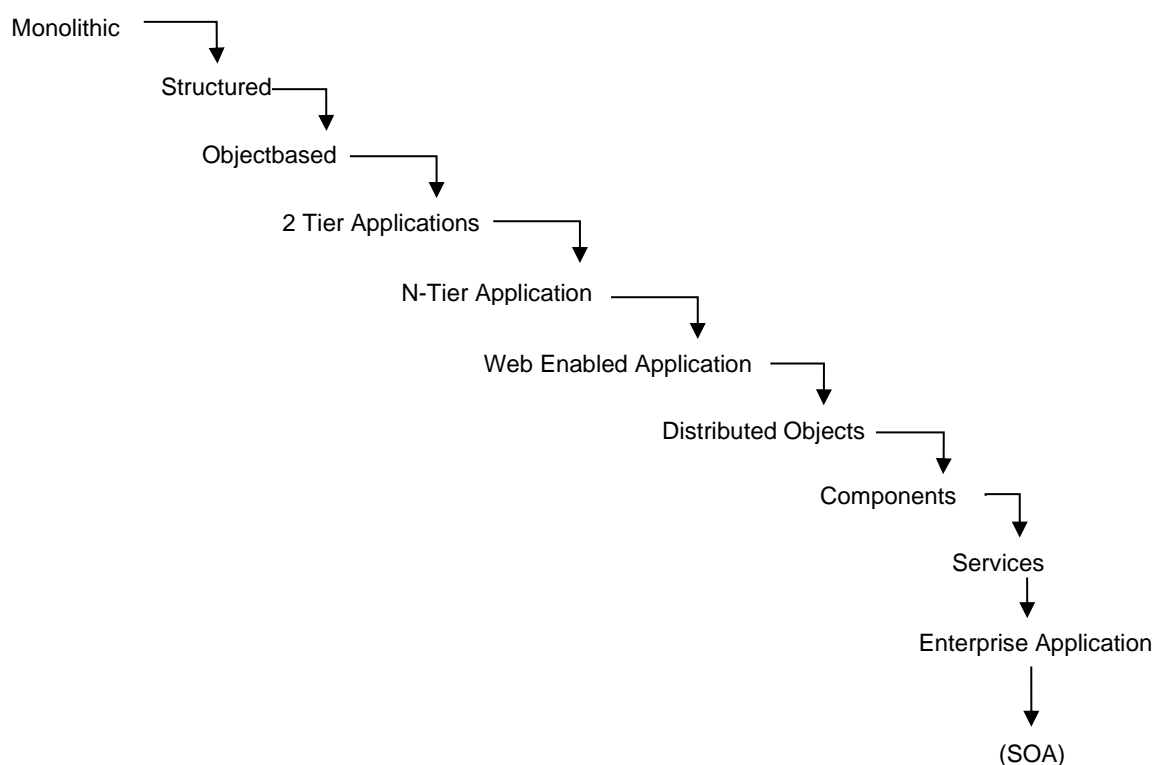
Figure 1. Evaluation of Computing System

In [2], [13], the fuzzy set theory conversion scales are used to convert the linguistic terms into fuzzy numbers. For rating the criteria and alternative, a scale is required from 1 to 9. The intervals are chosen in terms of a unique representation from scale 1 to 9 for the triangular fuzzy number used for five linguistic ratings.

Table 1. Fuzzy Rating for Linguistic Variables

Fuzzy number	Alternative Assessment	QA Weights
(1,1,3)	Very Poor (VP)	Very Low (VL)
(1,3,5)	Poor (P)	Low (L)
(3,5,7)	Fair (F)	Medium (M)
(5,7,9)	Good (G)	High (H)
(7,9,9)	Very Good (VG)	Very High (VH)

2. Literature review

Previously, many mathematical and soft-computing techniques have been used for estimation of SOA system reliability. In this paper, we have divided the related work in few categories namely architectural models, mathematical models and soft computing based models.

2.1. Architectural based models

According to [4], [1] and [7], reliability is one of the important metrics of software system.

Threemodels are used under the category of architectural based models.

State-based model utilizes control graphs generated using Continuous Time Markov Chains, Discrete Time Markov Chains or Semi-Markov Process to characterize the application architecture as proposed by [4]. These models deduce component and interface failure rates that can differ with time. It is proposed that the time spent in a module can be described as a general distribution function with a finite mean.

The path-based model is similar in steps as state-based model, first observing the various execution paths and then their occurrences that particular program can exercise. In path-based model[1],first calculate all possible path reliability and then multiply the components reliability to each path reliability. Path-based model have two example models, one is Shooman model proposed in[4], and other is Scenario Based Reliability Analysis model proposed by [17].

Additive models are used for estimating the complete reliability of the application by combining reliability data of the individual services. Non-Homogeneous Poisson Process (NHPP) is used to model the individual services reliability as suggested by [4]. Two examples of additive model are: Xie-Wohlin model and Everett model.

2.2. Mathematical models

Poernomo et al. [9] have used component-based architecture and proposed the reliability prediction. In this approach, a way is provided to predict the reliability on the basic of Markov model and parametric contracts. Rewrite logic (RABRL) approach is used for reliability estimation proposed by [18]. In this approach, those systems are considered whose specification is considered with an operational profile. Maude's rewrite technique is used for estimating the reliability. Modified adaptive testing (MAT) approach is used to estimate the software reliability proposed by [9]. In this model test case histories are used for testing.

2.3. Soft-Computing models

Genetic algorithm (GA) and support vector machine (SVM) is used to estimate the software reliability proposed in[19]. GA is used by SVM to reliability estimation for the SVM. This model is less dependent on failure size of data as compared to other models and is precise for predicting the reliability.

A reliability prediction and assessment technique that is unified modelling language (UML) proposed in[12] for Bayesian reliability estimation model. In this model, early design phase is used to predict the reliability based on UML diagrams. In, [20] order to reduce the computational cost generated by MCDA algorithms, the researchers has applied fuzzy logic technique to address the uncertainties in the definition of aggregate machine learning to pre-classification of EXEHDA middleware resources and ideal weights for QoS attributes.

In [6], to address the software reliability on various datasets being chosen from industrial software following ML techniques including adaptive neuron fuzzy inference system (ANFIS), feed forward back propagation neural network, general regression neural network, support vector machines, multilayer perceptron, Bagging, M5P cascading forward back propagation neural network, reduced error pruning tree, instance based learning, linear regression, and M5Rules are used.

3. Fuzzy TOPSIS approach

Fuzzy TOPSIS method was first proposed by Hwang et.al.[5].This method is used to get the best alternative among available alternatives.

The abbreviation of Technique for Order Preference by Similarity to Ideal Solution is TOPSIS [2]. It is a method of multiple-criteria decision analysis developed by Hwang and Yoon in 1981. It uses the finite numbers of criteria for ranking the alternatives. TOPSIS has been applied in many fields as proposed by [3] like supply chain management, business, logistics, marketing management, energy management, chemical engineering and etc. and it is

not limited to a specific field. In TOPSIS, the best alternative is chosen on the basis of distant measure, which is the alternative distance very far from negative ideal solution and near to positive ideal solution. In this method, the benefit criteria maximizes while cost criteria minimizes by using positive ideal solution, but on the other side, the benefit criteria minimizes and cost criteria is maximizes by using negative ideal solution.

This method has following steps:

Step 1: Rate all the alternatives by decision makers
In this algorithm, we have three alternatives such as A1, A2 and A3 for comparison with four criteria such as C1, C2, C3 and C4 also we have four decision makers namely D1, D2, D3 and D4. In table-2 show the rating of all alternatives by decision makers.

Table 2. Alternative Rating

Criteria	Alternative	D1	D2	D3	D4
C1	A1	VP	P	F	F
	A2	G	VG	VP	P
	A3	F	G	G	F
C2	A1	P	F	P	P
	A2	P	VP	F	P
	A3	F	G	VG	F
C3	A1	G	VG	G	G
	A2	VP	F	P	F
	A3	P	VP	P	VP
C4	A1	G	P	VP	F
	A2	VG	G	F	F
	A3	G	F	G	F

Step 2: Weightage of each criteria by decision makers.

Table 3. Criteria Weightage

Criteria	D1	D2	D3	D4
C1	H	L	M	H
C2	VH	M	H	M
C3	VH	H	M	L
C4	M	VL	L	M

Step 3: Apply fuzzy number for alternative rating and criteria weightage

Table 4. Fuzzy Number for Alternative Rating and Criteria Weightage

Criteria	Alternative	D1	D2	D3	D4
C1	A1	(1,1,3)	(1,3,5)	(3,5,7)	(3,5,7)
	A2	(5,7,9)	(7,9,9)	(1,1,3)	(1,3,5)
	A3	(3,5,7)	(5,7,9)	(5,7,9)	(3,5,7)
C2	A1	(1,3,5)	(3,5,7)	(1,3,5)	(1,3,5)
	A2	(1,3,5)	(1,1,3)	(3,5,7)	(1,3,5)
	A3	(3,5,7)	(5,7,9)	(7,9,9)	(3,5,7)
C3	A1	(5,7,9)	(7,9,9)	(5,7,9)	(5,7,9)

	A2	(1,1,3)	(3,5,7)	(1,3,5)	(3,5,7)
	A3	(1,3,5)	(1,1,3)	(1,3,5)	(1,1,3)
C4	A1	(5,7,9)	(1,3,5)	(1,1,3)	(3,5,7)
	A2	(7,9,9)	(5,7,9)	(3,5,7)	(3,5,7)
	A3	(5,7,9)	(3,5,7)	(5,7,9)	(3,5,7)

Table 5. Fuzzy Number For Criteria Weightage

Criteria	D1	D2	D3	D4
C1	(5,7,9)	(1,3,5)	(3,5,7)	(5,7,9)
C2	(7,9,9)	(3,5,7)	(5,7,9)	(3,5,7)
C3	(7,9,9)	(5,7,9)	(3,5,7)	(1,3,5)
C4	(3,5,7)	(1,1,3)	(1,3,5)	(3,5,7)

Step 4: The fuzzy values x_{ij}, where $(i=1,….m; j=1,….n)$, for all alternative i with respect to each criteria j. and also the appropriate fuzzy value w_j, $(j=1,…n)$ for each criteria is chosen.
Calculating the aggregate fuzzy rating x_{ij} of alternative i and criteria j as shown in Eq. (1) aggregate fuzzy weight w_j of each criterion is shown in Eq. (2)
And $x_{ij} = (a_{ij}^k, b_{ij}^k, c_{ij}^k)$
Where, $a_{ij} = min_k\{a_{ij}^k\}$; $b_{ij} = \frac{1}{k}\sum_{k=1}^{k} b_{ij}^k$; $c_{ij} = min_k\{c_{ij}^k\}$ (1)
$$w_j^k = w_{j1}, w_{j2}, w_{j3}$$
Where, $w_{j1} = min_k\{w_{jk1}\}$; $w_{j2} = \frac{1}{k}\sum_{k=1}^{k} w_{jk2}$;
$$w_{j3} = max_k\{w_{jk3}\}; \qquad (2)$$
Step 5: Fuzzy multicritera group decision making (GDM) and process of normalizing

$$D = A2 \begin{bmatrix} \bar{x}_{11} & x_{12} & \cdots & \bar{x}_{1n} \\ \bar{x}_{21} & x_{22} & ... & \bar{x}_{2n} \\ ... & ... & ... & ... \\ \bar{x}_{m1} & \bar{x}_{m2} & \cdots & \bar{x}_{mn} \end{bmatrix},$$

$\overline{W} = (\overline{w1}, \overline{w2},….\overline{wn})$
Normalizing the matrix using below Eq.
$\bar{R} = [\bar{r}ij]_{m*n}$,   i=1,…m, j=1,2….n
$\bar{r}ij = (\frac{aij}{cj*}, \frac{bij}{cj*}, \frac{cij}{cj*})$
$cj* = max\ c_{ij}$(benefit criteria)   (3)
$\bar{r}ij = (\frac{\bar{aJ}}{cij}, \frac{\bar{aj}}{bij}, \frac{\bar{aj}}{aij})$
$\bar{a}j = max\ a_{ij}$(Cost criteria)   (4)
Weighted normalized fuzzy decision matrix,
$\bar{P} = [\bar{p}ij]$ where $\bar{p}ij = \bar{r}ij * \bar{w}j$
    (5)
Step 6: Calculate FPIS and FNIS using the given Eq.
$A^+ = (p^+1, p^+2,…,p^+n)$ where
$p^+j = max\ \{pij3\}, i= 1,2,…, m; j = 1,2…,n$ (6)
$A^- = (p-1, p-2,….,p-n)$ where
$p-j = min\ \{pij1\}, i= 1,2,…,m; j = 1, 2,…,n$ (7)
p+ is selected as maximum value from each row and p- selected as minimum value for each row.
Step 7: Calculate FPIS and PNIS for each criteria using the given Eq.
FPIS (A1) = d (pij, p1+) and   (8)
FNIS (A1) = d(pij, p1-)
    (9)

Now, we have to calculate distance of each criteria using Eq. 2 from FPIS and FNIS for all alternatives.

$$d(\bar{a}, \bar{b}) = \sqrt{\frac{1}{3}[(a1-b1)2 + (a2-b2)2 + (a3-b3)2]}$$

    (10)
Step 8: Calculate the distance of each weighted alternative
$d_i^+ = \sum_{j}^{n} = 1 d(\bar{p}ij, pj^+)$ (11)
$d_i^- = \sum_{j}^{n} = 1 d(\bar{p}ij, pj^-)$ (12)
Step 9: Find closeness coefficient of each alternative.
$CCi = d^-i/(d^-i + d^+i)$, i= 1, 2,…,m   (13)
Step 10: Ranking the each and every alternative according to the decreasing order of the closeness coefficient CCi and finally best alternative is selected.

4. Applications of fuzzy TOPSIS for component ranking

Basically, fuzzy TOPSIS approach is used for selection purpose and rank the selected service. Fuzzy TOPSIS has different applications. Some of them are; for selecting the components in CBS [15], selection of Warehouse Location, to scoring the negotiation offers in ill-structured negotiation problems, evaluating sustainable transportation systems, or supplier selection and SWOT analysis.
In the paper, we have used Fuzzy TOPSIS for finding the raking of services. We have selected four important factors: server workload, faulty service replacement in composite service, redundancy at broker level and service reusability as proposed in [13] and as criteria for evaluating rank.

5. Case study

5.1. Graphical representation of safety critical system

In this system, domain is used as a set of components and execution path which start from starting node to the destination node. Experts in the field as the team of decision makers D1, D2, D3, and D4 was formed to evaluate the all alternatives A1, A2 and A3 based on four criterion C1, C2, C3, and C4. Let, the linguistic variables for reliability ranking be: Very Poor (VP), Poor (P), Fair (F), Good (G) and Very Good (VG). Some linguistic variables are used for criterion weighting as follow Very High (VH), High (H), Medium (M), Low (L) and Very Low (VL). Fuzzy numbers for all linguistic variables are shown in Table 1.

6. Results

Step 1: Set the fuzzy values or linguistic rating for alternatives with respect to criteria.

The linguistic variables for each reliability ranking are supposed as: Very Good (VG), Good (G), Fair (F), Poor (P) and Very Poor (VP). Let the linguistic variables for criterion weightage is: Very High (VH), High (H), Medium (M), Low (L), and Very Low (VL). Table 1 and table 2 are used for fuzzy numbers for all linguistic variables and weightage of different criteria's by different decision makers respectively.

Step 2: In SOA system, the contribution in overall system reliability may vary on the basis of services. The decision makers are used for collecting the weightage for each criterion. The weight matrix is used to evaluation of reliability. Experts are used for providing the inputs to design the weightage matrix. Decision makers subjective judgments develop the fuzzy criteria and use the linguistic variables to evaluate the ratings of alternatives with respect to each criterion as presented in Table 3.

Step 3: Table 4 and Table 5 shows the fuzzy number for alternative rating and criteria weightage using fuzzy rating for linguistic variable.

Step 4: using fuzzy values with respect to all criteria, we calculate the aggregated fuzzy value and aggregate fuzzy weight using Eq. (1) and Eq. (2) respectively as show in Table 6 and Table 7.

Table 6. Aggregated Fuzzy Decision Matrix for Alternative

Criteria	Alternative	
C1	A1	(1, 3.5, 7)
	A2	(1, 5, 9)
	A3	(3, 6, 9)
C2	A1	(1, 3.5, 7)
	A2	(1, 3, 7)
	A3	(3, 6.5, 9)
C3	A1	(5, 7.5, 9)
	A2	(1, 3.5, 7)
	A3	(1, 2, 5)
C4	A1	(1, 4, 9)
	A2	(3, 6.5, 9)
	A3	(3, 6,9)

Table 7. For Criteria Weightage, Aggregated Fuzzy Decision Matrix

Criteria	Aggregated Weightage
C1	(1, 5.5, 9)
C2	(3, 6.5, 9)
C3	(1, 6, 9)
C4	(1, 3.5, 7)

Step 5: Applying the normalization process for calculating the normalized aggregated fuzzy decision matrix for alternative and weighted normalized fuzzy decision matrix using Eq. (3), (4) and Eq. (5). As show in Table 8.

Step 6: The Fuzzy Positive Ideal Solution (FPIS) and Fuzzy Negative Ideal Solution (FNIS) are calculated using A-max and A-min in Eq. (6) and (7) as show in Table 9.

Step 7: calculate FPIS and PNIS for each criteria using the given Eq. (8) and Eq. (9) and distance for each criteria for using Eq. (10). The distance is calculated between positive ideal and the weighted alternative. The optimal alternative is calculated using alternative which has farthest distance from the negative ideal solution and shortest distance from the ideal solution. Table 10 shows the distance of criteria of each alternative from FPIS and FNIS.

Table 8. Matrix of Normalized Aggregated Fuzzy Decision for Alternative

$$
D = \begin{array}{c} A1 \\ A2 \\ A3 \end{array}
\begin{bmatrix}
(0.143, 0.5, 1) & (0.143, 0.5, 1) & (0.556, 0.8331, 1) & (0.111, 0.445, 1) \\
(0.111, 0.556, 1) & (0.143, 0.429, 1) & (0.143, 0.5, 1) & (0.333, 0.722, 1) \\
(0.333, 0.667, 1) & (0.334, 0.722, 1) & (0.333, 0.667, 1) & (0.333, 0.667, 1)
\end{bmatrix}
$$

$$
\begin{array}{cccc}
C1 & C2 & C3 & C4
\end{array}
$$

$$
\overline{Pij} = \begin{array}{c} A1 \\ A2 \\ A3 \end{array}
\begin{bmatrix}
(0.143, 0.5, 1) & (0.143, 0.5, 1) & (0.556, 0.8331, 1) & (0.111, 0.445, 1) \\
(0.111, 0.556, 1) & (0.143, 0.429, 1) & (0.143, 0.5, 1) & (0.333, 0.722, 1) \\
(0.333, 0.667, 1) & (0.334, 0.722, 1) & (0.333, 0.667, 1) & (0.333, 0.667, 1)
\end{bmatrix}
$$

Table 9. Fuzzy Positive Ideal Solution (Fpis) And Fuzzy Negative Ideal Solution (Fnis)

$$
\frac{Positive\ FPIS}{Negative\ FNIS} = \begin{vmatrix}
(0.333, 2.75, 9) & (1.002, 4.693, 9) & (0.556, 4.998, 9) & (0.333, 2.527, 7) \\
(0.111, 3.058, 9) & (0.429, 2.789, 9) & (0.143, 3, 9) & (0.111, 1.558, 7)
\end{vmatrix}
$$

7. Discussion

Software reliability is generally an important factor that affects system reliability. With respect to the software quality, reliability plays an important role. Some software quality features are usability, functionality, serviceability, installability, capability documentation, performance, and maintainability. Now-a-days, to achieve Software Reliability is strongly recommended, since the software complication tends to be high. It has become very difficult for a highly complex system as well as Software, to achieve desired level of reliability. The literature of Chen et.al. [3], Tyagi et.al. [8], and Krohling et.al.[10] concludes that TOPSIS method is used for selection of particular components or service on the basis of available criteria and provide the ranking for evaluation of system.

On the basis of brief literature survey, TOPSIS method is one of the best approaches to estimate the reliability of SOA system for ranking the services.

8. Conclusion

The approach suggested in this paper uses technique for order preference by similarity to ideal solution (TOPSIS) to provide the ranking of the

services on the basis of SOA system reliability. It also provides the resultant frequency of each service due to closeness coefficient. It is used to find out the ranking of each service for any kind of SOA system, which is helpful for overall system reliability which is computed on the basis of some criterion.

This TOPSIS method has some limitations; this is time consuming process for complex software application. However, we have selected some important criteria but there may also be some other good criteria.

9. References

[1] S. Arikan, (2012) "Automatic Reliability Management in SOA-based critical systems", Integrated Communication Systems Group University of Kaiserslautern, European Conference on Service-Oriented and Cloud Computing.

[2] Chen and Tung (2000) "Extensions of the TOPSIS for group decision-making under fuzzy environment", Fuzzy Sets and Systems Elsevier. Volume 114, Issue 1, 16 August 2000, doi.org/10.1016/S0165-0114(97)00377-1,pp 1-9.

[3] C.T. Chen., C.T. Lin and S. F. Huang, (2006), "A fuzzy approach for supplier evaluation and selection in supply chain management", International Journal of Production Economics, Vol. 102 No. 2, pp. 289-301.

[4] K. Goseva and K. S. Trivedi, (2001) "Architecture based approach to reliability assessment of software systems", Proceedings of the International Journal on Performance Evaluation, Vol. 45, Issue 2-3, pp 179-204.

[5] C.L. Hwang, KP Yoon (1981) "Multiple attributes decision making methods and applications", Springer, Berlin, pp 12–21.

[6] A. Jaiswal and R. Malhotra (2016) "Software reliability prediction using machine learning techniques", Int J SystAssurEngManag (February 2018) 9(1):230–244, DOI:10.1007/s13198-016-0543-y.

[7] A. Kumar Sharma, M. Sharma, B. Kaushik, (2007) "Software Reliability Engineering and Its Challenges, Proceedings of National Conference on Challenges and Opportunities in Information Technology" (COIT-2007) RIMT-IET, MandiGobindgarh.

[8] K. Tyagi and A. Sharma (2014) "Ranking of components for reliability estimation of CBSS using Fuzzy TOPSIS", Int J SystAssurEngManag DOI 10.1007/s13198-014-0287-5.

[9] IH. Poernomo, HW. Schmidt (2003) "Reliability prediction for component-based software architecture". J SystSoftw 66(3):241–252.

[10] A.R Krohling. et.al. researched on "An approach Based on TOPSIS for Ranking Evolutionary Algorithms", published by Elsevier in 2015 B.V, pp 308-317.

[11] A. Seth, AR. Singh and K. Tyagi (2010) "Aspects of Service-Oriented Computing", The Journal for Computing Teachers, spring.

[12] H. Singh, V. Cortellessa, B. Cukic, E. Gunel, V. Bharadwaj (2001) "A Bayesian approach to reliability prediction and assessment of component-based systems", 12th International Symposium on. IEEE.

[13] N. Singh and K. Tyagi, (2015) "Ranking of Services for Reliability Estiation of SOA System using Multicriteria Analysis with Similarity-Based Approach", International Journal of System Assurance Engineering and Management (IJSAEM), Springer, ISSN 0975-6809, DOI 10.1007/s13198-0150339-5, 20 January 2015.

[14] N.Singh and K Tyagi, (2015) "A Literature Review of the Reliability of Composite Web Service in Service-Oriented Architecture", ACM SIGSOFT Software Engineering Notes, DOI:10.1145/2693208.2693237, Vol. 40, Number 1, pp 1-8, January 2015.

[15] N. Singh and K. Tyagi, (2015) "Important Factors for Estimating Reliability of SOA", 2nd IEEE International Conference on Advances in Computer Engineering and Applications (ICACEA-2015) in Ghaziabad,India, DOI:10.1109/ICACEA.2015.7164734, pp 381-386, 19-20 March 2015.

[16] L. Srinivasan, J. Treadwell (2005) "An Overview of Service-Oriented Architecture, Web Services and Grid Computing", White Paper of HP Software Global Business Unit.

[17] S. M. Yacoub, B. Cukic, and H. H. Ammar, (2004) "A Scenario-Based Reliability Analysis approach for Component-Based Software", IEEE Transactions on Reliability, Vol. 53, Issue 4, pp 465-480.

[18] D. Wang, N. Huang, (2008) "Reliability analysis of component-based software based on rewrite logic", 12th IEEE International Workshop on Future Trends of Distributed Computing Systems, pp 126–132.

[19] JH. Lo (2010) "Early software reliability prediction based on support vector machines with genetic algorithms" Fifth IEEE ConfInd Electron App, pp 2221–2226.

[20] R. Dilli, A. Argou, R. Reiser, A. Yamin (2018) "IoT Resources Ranking: Decision Making Under Uncertainty Combining Machine Learning and Fuzzy Logic", Springer International Publishing AG, part of Springer Nature 2018G. A. Barreto and R. Coelho (Eds.): NAFIPS 2018, CCIS 831, pp. 119–131, 2018. DOI:10.1007/978-3-319-95312-0_11.