Computers and Electrical Engineering 71 (2018) 966–977

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Performance evaluation of semantic based service selection methods☆

S. Maheswari[a,*], G.R. Karpagam[b]

[a] Department of Computer Science and Engineering, National Engineering College, kovilpatti 628503, India
[b] Department of Computer Science and Engineering, PSG College of Technology, Coimbatore 641004, India

ARTICLE INFO

Article history:
Received 8 March 2017
Revised 7 October 2017
Accepted 11 October 2017
Available online 31 October 2017

Keywords:
Semantic web
Quality of service
Analysis of variance
Service selection methods
Paired t-test

ABSTRACT

Web service is a functionality which is accessed through the internet. Different service providers offer same functionality which has increased the efficiency of service to a great amount.The customers have to access by selecting web services.The goal is to find the appropriate web services that fulfill the customer requirements. Service selection is the process of selecting the most suitable service from the list of discovered services. This process is usually based on the metrics like Quality of Services(QoS) and are ranked based on the QoS values using four service selection techniques namely Analytic Hierarchy Process (AHP), Logic Scoring Preference (LSP), Fuzzy Technique for Order of Preference by Similarity to Ideal Solution (Fuzzy Topsis) and Web service relevancy factor (WsRF). Inappropriate service selection leads to many complications which in turn require an evaluation scheme for adopting a distinct method. The performance evaluation of the service selection methods is carried out using analysis of variance (ANOVA). To improve the performance of service selection, the Iterative MapReduce is used. This reduces the execution time of the each selection process by running the process parallel.

1. Introduction

In recent years, web service has been the most successful and popular technology in distributed computing. Web services communicate based on Hypertext Transfer Protocol (HTTP), described using Extensible Markup Language (XML), and transported via Simple Object Access Protocol (SOAP). Due to these characteristics, web services have been widely adopted and applied in the IT industry. The accomplishments in the area of the Web Service (WS) research have led to an explosion of Web Services.

The Web Services are advertised in the registries, and discovered based on the needs [4] of the consumer. These registries rely on the current Web Service standard stack that uses WSDL and UDDI for Web Service description and discovery. Developers and researchers have realized the difficulty of discovering web services to fulfil their tasks in such a large scale publication of web services. Current web technologies are built upon syntactical techniques; hence, search mechanisms cannot differentiate similar web services.

This situation creates a need for augmenting semantics to web services to achieve precise and accurate results. Semantic Web Service descriptions are provided using ontologies which describe the meaning to concepts and relationships between

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. V. Suma

* Corresponding author.
E-mail address: kannanjesus123@yahoo.co.in (S. Maheswari).

https://doi.org/10.1016/j.compeleceng.2017.10.006
0045-7906/

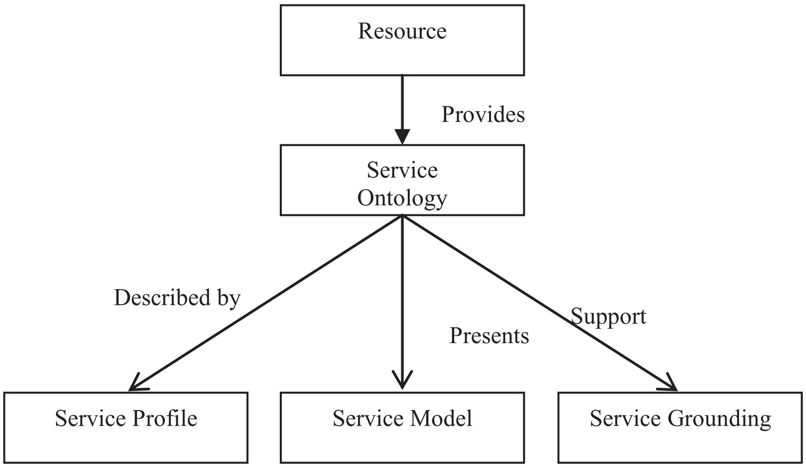

Fig. 1. Organization of OWL-S ontology model.

them. Discovery algorithms [8] are used to select appropriate services among multiple similar services that perform the same function using these ontologies. OWL-S is the joint effort of Semantic Web [9] and Web Service. The semantic of web service is described using OWL-S. The OWL-S ontology is divided into three main categories namely service profile, process model, and service grounding.

Service Profile specifies what the service does. The profile description is used for discovering the service. Profile is a sub class of service profile. For each profile instance, the process it describes and its functional characteristics are associated with their type. It includes the three functionality limitations on applicability and quality of service, publisher and contact information and service name and description. *Process model* specifies how the service works. A process model has a single process which can be atomic, simple or composite. Each process has a set of input, output, pre conditions and effects. The profile contains several links to a process. *Service Grounding* specifies how the service is implemented. The grounding ontology specializes the service groundings as WSDL groundings. It is an interaction between the user needs and services. All the above service ontology description contains service concepts which link the Service Profile, Service Model and Service Grounding concepts together. The motivation for the structuring of the ontology of web services is the need to provide three essential types of knowledge about a service which are shown in Fig. 1.

However, with proliferation of web services, invoking the desired service effectively and efficiently has become a challenging issue. QoS (Quality of Service) which describes the non-functional attributes [21] of the service can be used as a distinguishable factor as it reflects the users' expectations. Further, discovery algorithms should perform QoS-based filtering (Matchmaking) and ranking (selection) on WS advertisements in order to produce ranked results.

Consider an online shopping domain that facilitates services for buying books online from several service providers (suppliers/sellers). The consumers (user/buyer) have the buying preferences like cost, urgency of delivery, trustworthiness, availability of books. Therefore the user has to request for book shopping services along with their preferences. Different users may have different values for non functional properties of a service. Hence, a suitable service selection method is demanded to provide the most suitable book shopping service according to individual's preferences for the non-functional properties such as Response time, Throughput, Reliability, Scalability, Availability, Accessibility, Reputation and Cost.

The service selection method can be classified into two categories: Distance based and Proportion based approach. Based on Distance, it is further classified into linear normalization (WsRF, AHP) and Vector normalization (Fuzzy Topsis). Max-Min (LSP) Normalization technique is a type of Proportion Based approach. AHP and WsRF methods use Hierarchical (H) approach, whereas LSP and Fuzzy Topsis methods use Non Hierarchical (NH) approach for service selection. These Service selection methods can be executed by means of sequential approach and Map Reduce approach. The rest of this paper is organized as follows. Section 2 discusses short overview of background work and role of QoS for web services. Section 3 presents the motivation Scenario for QoS based semantic service selection. Section 4 describes the proposed model for QoS based Semantic web service selection. Section 5 presents the Performance Evaluation of Selection methods using ANOVA. Finally this paper is concluded in Section 6

2. Background

The current semantic web services research focuses on solving the issues of service discovery and service selection. Service discovery is the process of finding web services that meet a user's specified preferences. Service selection is a process of choosing a web service from the services that are discovered. When Web Services offer the same functionality that differs in Quality of Service (QoS) [23], it brings more challenges to select the services attempting to satisfy the users' QoS preferences. [12] Developed a QoS Based Web Service Selection Framework [24]. This framework has been used to rank the web service

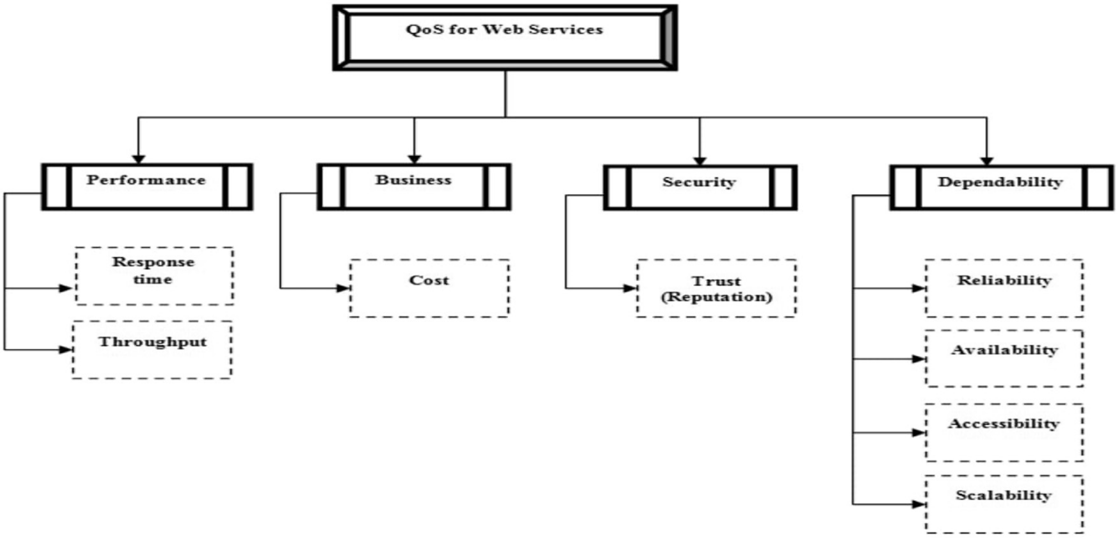

Fig. 2. QoS parameters.

based on QoS Parameters and QoS based ranking of web services using WsRF normalization method. [10] Web Service Discovery is a technique to locate the web services that match the requirements of the consumer request. Discovery algorithms pertaining to semantic matchmaking are analyzed in this paper, further proposes a Standardized Engine that incorporates a layered approach. The first layer, lexicon search, performs semantic matching using word net; second layer, Concepts based refiner, is responsible for comparing the concepts available in the domain ontology; the third layer, QoS (Quality of Service) based refiner, performs a statistical analysis using QoS parameters namely response time, throughput and availability. [20] Discuss the match making algorithm that considers, only input(s) and output(s) of the consumer request and that of the advertised services. [1] Proposed an approach that concentrates Input, Output and QoS values for the service discovery. [13] analyzed interdependencies among QoS parameters and described the QoS based ranking of web services using LSP method. [3,6] proposed evaluation method based on the technique for Order Preference by Similarity to an Ideal Solution (Topsis) to help service consumers and providers to analyze available web services with fuzzy opinions. [13] Proposed a nonfunctional property-based service selection method by modifying the Logic Scoring Preference (LSP) method. It captures the relationship between replaceability and simultaneity. The LSP method is used to evaluate the quantitative aspects of competitive Web services and support selection decision making. Ordered Weighted Averaging operators (OWA) are used to determine the ORNESS degree.

[11] Discussed an enhanced QoS based semantic web service selection and ranking using Fuzzy Topsis [5] method. This method is used to increase the availability of the service using new dynamic replication algorithm. [19] Proposed various service selection methods, selection based on user requirements and QoS parameters**.** [14–16] proposed the Analysis of variance for comparison of service selection methods. [7,17,18] suggested the replication and parallism in service selection.

2.1. Role of QoS for web services

According to International quality standard ISO [7,22]**,** quality is defined as the totality of features and characteristics of a product or service that bears on its ability to satisfy stated or implied needs. In addition to Web Service description, if the QoS description is provided for each Web Services, then more filtered list of Web Services will be displayed. Hence, the QoS [2] attribute values are considered as the significant factor for selecting the services.

In this work the term quality of service has been used for expressing non-functional requirements of a service. The attributes of QoS [6] are defined under two different aspects namely Domain-Independent and Domain-Dependent. The domain independent QoS attributes according to [23] are Performance Related QoS, Dependability Related QoS, Business Related QoS and Security Related QoS are shown in Fig. 2 Response time and Throughput are Performance Related QoS; Availability, Accessibility, Reliability and Scalability are Dependability Related QoS; Execution cost and reputations are classified under Business Related Security Related QoS respectively

2.1.1. Performance related QoS

The performance of Web services [22] measures the speed in completing a service request. It can be measured by:

Response time: The maximum time that elapses from the moment that a web service receives a SOAP request until it produces the corresponding SOAP response is considered as response time. Response time=response completion time-user request time.

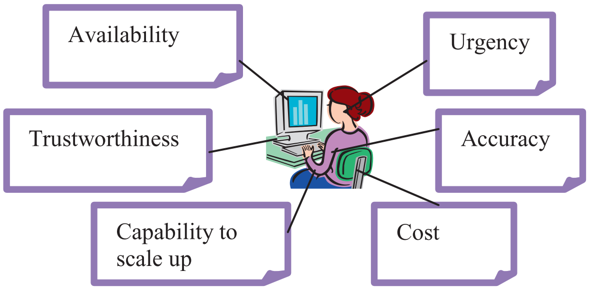

Fig. 3. Different users - Different QoS preferences.

Throughput: The number of Web service requests R for an operation o that can be processed by a service S within a given period of time is expressed to as seconds or milliseconds. The maximum number of services that a platform providing web services can process the request over a unit time. Throughput=max complete requests/unit time

2.1.2. Dependability related QoS

Dependability is the combined aspects of availability, accessibility and reliability support performances and it relates to the ability of an item to be in a state to perform a required function

Availability: It is the probability that a service is operating when it is invoked. It is the probability that a service S is up and running. The ratio of time period in which a web service exists or it is ready for use. Availability=uptime/total time=uptime / (uptime+ downtime).

Accessibility: It is the capability of serving the Web Service request. The Web service might be available but not accessible because of a high volume of requests. The degree that a system is normatively operated to counteract request messages without delay. p_ accessibility=p_ availability at time t.

Reliability: User sets a period of measurements. The number of failures over a period of time is the measure of reliability. It is derived as the unsuccessful invocations for the given period of measurement.

Scalability: The capability of increasing the capacity of service provider's system and ability to process requests in a given time interval.

2.1.3. Business related QoS

Execution cost: The Execution cost is the fee that the service requester has to pay for invoking the web service. Web service providers either directly advertise the execution cost or provide other means for the requesters to inquire about it. It is a measure of the cost involved in requesting the service.

2.1.4. Security related QoS

A trust and reputation mechanism largely relies on user feedbacks to rate and believe a service.

Reputation: It is the measure of trustworthiness of a service, based on the end user's experiences of using the service. Different end users may have different opinions on the same service. The reputation can be defined as the average ranking given to the service by the end users. Reputation = ΣRi /n.

3. Motivation scenario

Consider an online shopping domain shown in Fig. 3 that facilitates services for buying any products online from several service providers (suppliers/sellers). The consumers (user/buyer) have the buying preferences like cost, urgency of delivery, trustworthiness, and availability of products. Therefore, the user has to request for product shopping services along with their preferences. Different users may have different values for non functional properties of a service.

Hence, a suitable service selection method is demanded to provide the most suitable service according to individual's preferences for the non functional properties such as cost, throughput, availability and response time. The objective of the selection process is to select best web service from the available services.

4. Proposed model - QoS based semantic web service selection

The Process of web service selection is shown in Fig. 4. The various components involved in the proposed framework are service provider, service consumer, service registry, service discovery agent and QoS agent.

The **service providers** register the web services in WSDL format. This WSDL is converted into OWL-S (functional description) and OWL-Q (non-functional description) of the services and it is stored in the semantic service registry. **Service consumer** searches desired web service in a semantic **registry**. The basic requirements for service selection can be represented as Triplet {I, P, O}

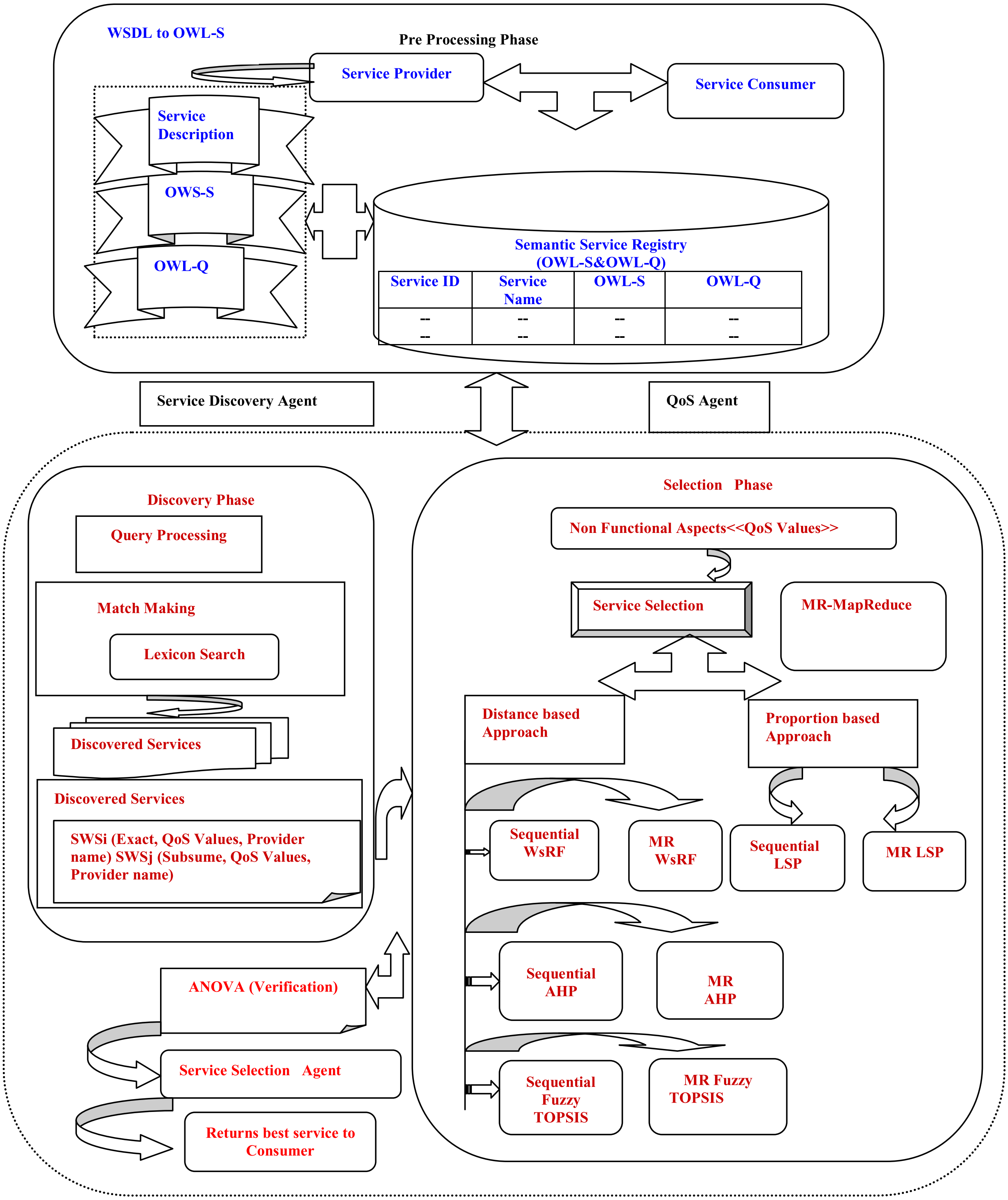

Fig. 4. QoS based semantic web service selection.

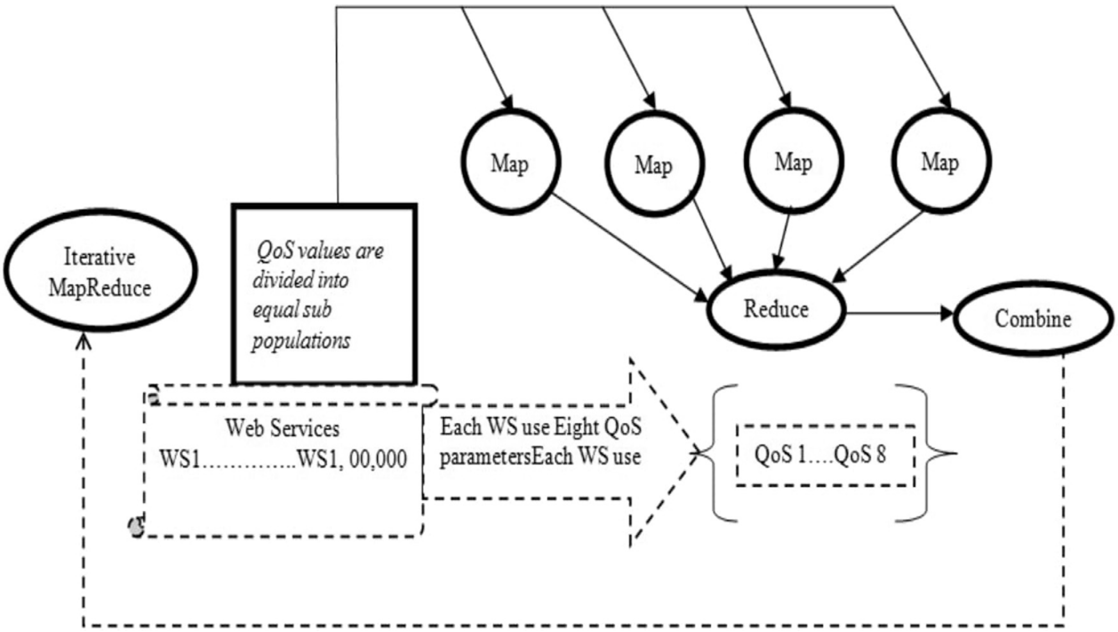

Fig. 5. Iterative MapReduce.

Where, I $\approx$ [C_R, S_P, $S_{R,}$] P $\approx$ [SS_M] O $\approx$ [R_R],

Service selection Process: {I, P, O} => { [C_R, S_P, $S_{R,}$], [SS_M], [R_R]}

Where, I- Input, P- Process, O - Output, C_R - Consumer Request, S_P -Service Provider, S_R - Service Registry, SS_M- Service Selection Methods, R_R- Retrieved Results

Service Discovery agent is responsible for discovery of relevant web services based on Input, Output, Precondition and effect values. Discovery phase performs stop word removal and stemming on the user request and then it is sent to the Matchmaking process. The first step of Matchmaking process is Lexicon Search.

In this the domain of the requested service and advertised services are matched. It is not likely that domain name saved in the registry will be same as the domain that the user selects. It might be available under a different name but with same meaning. If only the given domain names are taken for comparison, then there is a possibility that the suitable domain would not be selected. To avoid this, WordNet is used to find the synonyms of the given domain. The filtered services based on the domain are given as the input for the concept match. Then input(s), output(s), precondition and effect values of requested services are matched with the advertised services to discover the relevant service [10]. The concept matched services are sent to the QoS agent for service selection.

QoS agent is responsible for selection of suitable web services based on non functional parameters. The QoS parameters are Response time, Throughput, Reliability, Scalability, Availability, Accessibility, Reputation and Cost represented in OWL-Q. The services are ranked based on the QoS values using four service selection techniques namely Analytic Hierarchy Process (AHP), Logic Scoring Preference (LSP), Fuzzy Technique for Order of Preference by Similarity to Ideal Solution (Fuzzy Topsis) and Web service relevancy factor (WsRF). Proposed work concentrates on applying Map Reduce technique which reduces the time taken for selecting the services.

To improve the performance of service selection, the Iterative MapReduce is used. This reduces the execution time of the each selection process by running the process parallel. The system has to run using the iterative MapReduce environment and configuration of the MapReduce system which are Dell workstation, Intel Xenon CPU, E3-1240 V3@3.40 GHZ and RAM speed 16GB.

Fig. 5 shows the Iterative MapReduce. The input population for the service QoS values are divided into equal sub populations and are given to the service selection method like AHP ,WsRF,LSP,Fuzzy Topsis. The map task is responsible for performing the service selection method individually. The map task performs the ranking algorithm for that sub population and computes the list of ranked services. The output of the map tasks are given to the reduce task. The reduce task collects all the outputs from all the map tasks. Then ranked results are collected from combine task. The system has to run using the Iterative MapReduce environment. The number of Iterations can be changed according to the weights of service consumer.Finally the non functional parameters like Response time, Throughput, Reliability, Scalability, Availability, Accessibility, Reputation, and Cost are given as the input to the twister .The twister reduces the time taken for the selection of web services and that has been shown in Table 8.

Table 1
Tools and Platform.

S. No	Tools/Platform	Explanation
1	Netbeans (V 6.1)	A generic framework that provides a reliable and flexible architecture for developing an application
2	Protégé (Release 2.5.0)	Protégé is a free, open source collaborative ontology development environment for the web and supports two modeling ontology Protégé-Frames and Protégé-OWL editors. And also is based on the Java.
3	Fusion registry (V.1.0)	The FUSION Semantic Registry is a semantically-enhanced service registry that builds on the UDDI specification and augments its service publication and discovery facilities to overcome these limitations.
4	JADE	Java Agent Development Framework for Agent based service selection
5	Twister (V.0.9)	MapReduce programming model has simplified the implementations of many data parallel applications and Twister is a lightweight MapReduce runtime we have developed by incorporating enhancements MapReduce programming model.
6	ANOVA	Analysis of variance (ANOVA) is used to examine the variance of a dependent variable. And also a statistical method used to test differences between two or more means and it is a method for testing the hypothesis.

Table 2
Computational formulae for performance evaluation.

ANOVA Computational formula	Paired T-Test Computational formula
df (rows) = n−1 , df (columns) =n-1	Mean $\mu = \frac{\sum_{i=1}^{n} x_i}{n}$
df(rows) + df(columns) + df(error) = df(total) , df(total)= N-1	
SS(total) = $\sum (X_{ij} - \bar{X})^2$,	Variance $\sigma^2 = \frac{1}{n} \sum_{i-1}^{n} (x_i - \mu)^2$
SS (column & row) = n $\sum (X_j - \overline{X})^2$ and SS (Error) = $\sum (X_{ij} - \overline{X_j})^2$,	
SS (Error) = SS (total) – SS (column) – SS (row)	Number of Observations (n)
MS (row) = $\frac{ss(row)}{df(row)}$,	Pearson Correlation
MS (column) = $\frac{ss(column)}{df(column)}$,	$r = \frac{(n*\sum_{i=1}^{n} x_i y_i) - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{\sqrt{[n(\sum_{i=1}^{n} x_i^2)][n(\sum_{i=1}^{n} y_i^2) - (\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i)]}}$
MS (error) = $\frac{ss(error)}{df(error)}$	The hypothesized mean $d = \mu_1 - \mu_2$
F (column) = $\frac{ms(column)}{ms(error)}$,	$df = n - 1$
F (row) = $\frac{ms(row)}{ms(error)}$	t-Stat value Standard Error $SE = \sqrt{(\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2})}$

Efforts have been taken to evaluate the methods using Analysis of Variance technique. Two-way ANOVA test was conducted to check whether there is a variation among the results produced by the service selection methods. T-test for paired means is conducted to find the best selection method. **Service selection agent** is responsible for returning the selected services returned by the best service selection method to the service consumer.

5. Performance evaluatıon

This Section explains the tools and the platform involved in the implementation of the prototype which are shown in Fig. 4. Table 1 tabulates the tools and platforms, version used and its functions.

To compare the various selection methods ANOVA [18] and post hoc tests are used. The following are the lemma for ANOVA test and post hoc T-Test.

Lemma. *The probability results have to be interpreted by evaluating the F ratio. If the F ratio is larger than the F critical (F crit) value then there is a statistically significant difference. If it is smaller than the F crit value then there is no difference. If |t-stat| > t-crit, we reject the null hypothesis that means are same, and conclude the difference between the algorithms is significant. These two sets of results are not equally good.*

5.1. Basic notations and computational formulae for performance evaluation Table

Notations: Error - the variation or unexplained random error, Total - the total variation in the data, SS - the sum of squares due to the source, df - the degrees of freedom in the source, MS - the mean sum of squares due to the source, F - The F-statistic, P - The P value, n-total number of observation-total values, X_j = group mean, $\bar{X}$ = grand mean, X_{ij} = particular data values.

Performance evaluation tests the following hypotheses:
H0: The means of all the groups are equal.
H1: Not all the means are equal
Step 1: Formulation of **Null H_0** and **Alternate** Hypothesis $\mathbf{H_1}$

Table 3
ANOVA result table for WsRF and AHP.

ANOVA						
Source of Variation	SS	df	MS	F	P-value	F crit
Rows	11.51245	24	0.479685	1	0.5	1.98376
Columns	94.17159	1	94.17159	196.3196	4.74E−13	4.259677
Error	11.51245	24	0.479685			
Total	117.1965	49				

Table 4
ANOVA result table for LSP and Fuzzy Topsis.

ANOVA						
Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Rows	1.463204	24	0.060967	1.98376	0.483305	1.017418
Columns	0.106801	1	0.106801	4.259677	0.194394	1.782293
Error	1.438154	24	0.059923			
Total	3.008158	49				

Table 5
Post hoc test result for four parameters (statistically different).

t-Test: Paired Two Sample for Means		
	Variable 1	Variable 2
Mean	0.376104	0.468538
Variance	0.078913	0.041977
Observations	25	25
Pearson Correlation	0.009067	
Hypothesized Mean Difference	0	
Df	24	
t Stat	−1.33503	
P(T<=t) one-tail	0.097197	
t Critical one-tail	1.710882	
P(T<=t) two-tail	0.194394	
t Critical two-tail	2.063899	

Step 2: Determining or fixing the level of significance α**=0.05**
Step 3: To calculate test statistics value; to **compare sample** and population **mean**
Step 4: To compare calculated **test statistics** value with **table critical** value
If the calculated **test statistics value are greater than table critical value** reject the null. Hypothesis, otherwise accept the null hypothesis
Step 5: Based on the acceptance or rejection of null hypothesis we draw the **conclusion**
$\boldsymbol{\alpha}$**=0.05 (at 5% level we reject the null hypothesis)**

Both AHP and WsRF are hierarchical selection methods. In AHP the user is allowed to give preferred weights for the QoS parameters but this is not possible in WsRF. So by comparing hierarchy wise, the web service retrieved by both the methods using ANOVA is shown in Table 3.

Based on the lemma, in this case, F ratio = 1 and F crit = 1.983. Here, F ratio < F crit. Therefore, there is no significant difference between two selection methods: WsRF and AHP. Both the methods work similar and retrieve the best web services. Since the recommendations are made based on user preference, AHP selection method is preferred than WsRF.

The web services retrieved by LSP and Fuzzy Topsis selection methods for different user preferences are compared as shown in Table 4. Table 4 shows the user preference for four parameters namely for Availability, Accessibility, Reputation, and Cost. Maximum equal weight is given to these four parameters and least weight is given to all other parameters Response time, Throughput , Reliability , Scalability are equal to the unit value one. From the above weight values ANOVA tool produced the following result return by the ANOVA is seen in Table 4

Based on the lemma, in this case, F ratio = 1.98376and F crit = 1.017418. Here, F ratio > F crit. Therefore, the two selection methods: LSP and Fuzzy Topsis are statistically different. To find which selection method is best post hoc test is used. The result of that tool for the values is as follows, in Table 5.

If t Stat < t Critical. Therefore Fuzzy Topsis method is the best selection method when compared with LSP for Availability, Accessibility, Reputation, and Cost as user preference. Similarly, both the selection methods are analyzed for various user preferences using ANOVA and post hoc tests. The results are discussed in Section 6.

Table 6
Findings hierarchical selection methods results.

User Preference	Selection Method (Hierarchical selection methods: WsRF and AHP)
Performance related parameter (Response Time, Throuhput)	
Response Time, Throughput	No significant difference between WsRF and AHP
Response Time and Throughput	AHP is preferable, whereas WsRF not preferable
Dependability related parameter (Reliability, Scalability, Availability, Accessibility)	
Reliability, Scalability, Availability, Accessibility	AHP is preferable
Scalability and Reliability, Scalability and Availability, Scalability and Accessibility, Reliability and Availability, Reliability and Accessibility, Availability and Accessibility,	AHP is preferable, whereas WsRF not preferable
Scalability ,Reliability, Availability; Scalability,Reliability,Accessibility; Reliability,Availability,Accessibility;	AHP is preferable, whereas WsRF not preferable
Security related parameter (Reputation)	
Reputation	No significant difference between WsRF and AHP
Business related parameter(Cost)	
Cost	No significant difference between WsRF and AHP

5.2. Hierarchical selection methods: AHP Vs WSRF

This Section includes the wrapping up based on the comparison of various selection techniques using ANOVA and Post hoc test as shown in Tables 6 and 7. The following observations are concluded based on the QoS parameters chosen by the user:

5.3. Non-hierarchical selection methods: Fuzzy Topsis Vs LSP

Table 7
Findings non hierarchical selection methods results.

User Preference	Selection Method (Non Hierarchical selection methods: LSP and Fuzzy TOPSIS)
Single parameter preference	
Response Time, Throughput, Reliability, Availability, Accessibility, Reputation, Cost	Fuzzy TOPSIS is preferable
Scalability	No significant difference between LSP and Fuzzy TOPSIS
Two parameter preference	
Response time and Scalability, Scalability and Reliability, Response time and Reliability, Response time and Availability, Response time and Accessibility, Response time and Reputation, Response time and Cost, Throughput and Scalability, Throughput and Reliability, Throughput and Availability, Throughput and Accessibility, Throughput and Reputation, Throughput and Cost, Scalability and Reputation	Fuzzy TOPSIS is preferable
Response time and Throughput, Scalability and Availability, Scalability and Accessibility, Scalability and Cost, Reliability and Availability, Reliability and Accessibility, Reliability and Reputation, Reliability and Cost, Availability and Accessibility, Availability and Reputation, Availability and Cost, Availability and Reputation, Accessibility and Cost, Accessibility and Reputation, Reputation and Cost	No significant difference between LSP and Fuzzy TOPSIS
Three parameter preference	
Scalability, Reliability, Availability; Scalability, Reliability, Accessibility; Scalability, Availability, Accessibility; Reliability, Availability, Accessibility	No significant difference between LSP and Fuzzy TOPSIS
Response time, Throughput, Scalability; Accessibility, Reputation, Cost;	Fuzzy TOPSIS is preferable
Four parameter preference	
Response time, Throughput, Scalability, Reliability; Availability, Accessibility, Reputation, Cost;	Fuzzy TOPSIS is preferable
Five parameter preference	
Response time, Throughput, Scalability, Reliability, Availability; Reliability, Availability, Accessibility, Reputation, Cost	Fuzzy TOPSIS is preferable
Six parameter preference	
Response time, Throughput, Scalability, Reliability, Availability, and Accessibility; Scalability, Reliability, Availability, Accessibility, Reputation, Cost	Fuzzy TOPSIS is preferable
Seven parameter preference	
Response time, Throughput, Scalability, Reliability, Availability, Accessibility, and Reputation; Throughput, Scalability, Reliability, Availability, Accessibility, Reputation, Cost	Fuzzy TOPSIS is preferable
All parameters preference	Fuzzy TOPSIS is preferable

Table 8
Comparison of selection methods.

Service Selection Method	Distance Based			Proportion Based	Proposed
	WsRF	AHP	Fuzzy TOPSIS	LSP	Fuzzy TOPSIS with MapReduce
Hierarchical	√	√			
Non-Hierarchical			√	√	
Number of QoS Preferences	Single	Multiple	Multiple	Maximum 3	Multiple

Number of Services	Without Iterative MapReduce (secs)				With Iterative MapReduce (secs)			
	AHP	WsRF	LSP and OWA	Fuzzy TOPSIS	AHP	WsRF	LSP and OWA	Fuzzy TOPSIS
25,000	39,860	22,260	25,011	19,870	758	489	659	354
50,000	79,590	45,310	50,305	40,112	1,450	1,011	1330	716
75,000	1,59,100	92,150	1,00,300	78,760	2,980	2,168	2317	1428
1,00,000	3,18,050	1,85,240	2,10,050	1,68,500	5,588	4,170	4,675	3,019

5.4. Comparison of selection methods

AHP Vs WSRF: In AHP, the user is allowed to give preferred weights for the QoS parameters but this is not possible in WsRF. Conversely in WsRF, the user cannot request for a web service with their preferred weights as shown in Table 6. In AHP, decomposing the complex decision problems into a hierarchy of smaller sub problems increases the number of pair wise comparisons within sub problems. When the level of hierarchy increases the AHP method it takes a long time to synthesize weights.

Even for a small problem the computational requirement is more, as shown in Table 8. After analyzing the time taken for the selection methods, it can be seen that WsRF takes very less time than AHP. If the process is a sequential one then the time taken to retrieve 25,000 services is 39,860 seconds in AHP and 22,260 seconds in WsRF. After applying iterative Map Reduce, the time taken gets reduced to 758 seconds in AHP and 489 in WsRF. Based on the past experiences it is found, hierarchical approach is not suitable for handling such requests.

Fuzzy Topsis Vs LSP: Non-Hierarchical approaches of service selection allow a consumer to choose multiple preferences. LSP retrieves the best web service when three QoS parameter is considered. In multiple preferences, LSP produced better results for response time, throughput and scalability. Overall, this technique works better when single QoS parameter is considered and gives good results for the user request based on response time, throughput and scalability. Fuzzy Topsis selects best web services for both single and multiple QoS preferences as shown in Table 7.

Fuzzy Topsis takes very less time than LSP. If the process is a sequential one, then the time taken to retrieve 25000 services is 25,011 seconds in LSP and 19,870 in Fuzzy Topsis. After applying iterative Map Reduce, the time taken gets reduced to 25,011 to 659 seconds in LSP 19,870 to 354 seconds in Fuzzy Topsis as shown in Table 8. Therefore, Fuzzy Topsis method gives good results. Among non-hierarchical selection methods, Fuzzy Topsis gives better results for any user preferences.

6. Conclusion

The investigation on the Semantic Web Service Selection led to the identification of the research issues namely service selection and efficiency as discussed in previous Section. The investigate focuses on resolving the above mentioned issues and contributes towards providing solutions. The contributions are summarized in this Section. The first issue deals with QoS based service selection. The services are ranked based on the QoS values using four service selection techniques namely Analytic Hierarchy Process (AHP), Logic Scoring Preference (LSP), Fuzzy Technique for Order of Preference by Similarity to Ideal Solution (Fuzzy Topsis) and Web service relevancy factor (WsRF).Next issue deals on applying MapReduce technique which reduces the time taken for selecting the services. Efforts have been taken to evaluate the methods using Analysis of Variance technique.

In Hierarchical selection method, the consumer is allowed to give the preference based on QoS groups. Consider a situation where the consumers have multiple preferences which belong to multiple groups for example, Group 1 (Response time), Group2 (availability) and Group 3(reputation). This necessitates an approach to handle such multi-group, multi-preference based selection.

In multiple preferences, LSP produced better results, when single QoS parameters are considered and give good results for the user request based on response time, throughput and scalability. Fuzzy Topsis gave good results for any user preference. Best web services were retrieved for single QoS parameter criteria and also for multiple QoS preferences. Certain related study directions which require further investigations in the future as follows: context aware based service discovery, web service composition, Cloud computing

References

[1] Saleh AI, Ali T AFHA. Toward SWSs discovery: mapping from WSDL to OWL-S based on ontology search and standardization engine. IEEE Trans Knowl Data Eng 2013;25(5):1135–47.

[2] Chen X, Zheng Z, Yu Q, Lyu MR. Web service recommendation via exploiting location and QoS information. IEEE Trans Parallel Distributed Syst 2014;25(7).

[3] Lo C-C, Chen D-Y. 'Service selection based on fuzzy topsis method. In: IEEE 24th international conference on advanced information networking and applications workshops (WAINA); 2010. p. 367–72.

[4] Antony D, Mello D, Ananthanarayana VS. 'Semantic web service selection based on service provider's business offerings. IJSSST 2008;10(2):1473–8031.

[5] Fei LI, Yanxiang HE, Wensheng HU, Libing WU, Peng WEN. 'Web service selection based on fuzzy QoS attributes. J Comput Inf Syst 2011;51:198–205.

[6] Hao Gao A. Comprehensive approach to QoS-Based single and composite service selection Thesis. University of Wollongong; 2013.

[7] Ekanayake J. 'Twister: a runtime for iterative mapreduce'. In: Proceeding HPDC'10 of the 19th ACM international symposium on high performance distributed computing; 2010. p. 810–18.

[8] Kritikos D, Plexousakis. OWL-Q for semantic QoS-based web service description and discover. In: Proc. workshop service matchmaking and resource (SMRR); 2007.

[9] Khusro S, Jabeen F, Mashwani SR, Alam I. 'Linked open data: towards the realization of semantic web-a review. Indian J Sci Technol 2014;7(6) 745–64.

[10] Maheswari S, Karpagam GR, Manasaa S. Design of standardization engine for semantic web service selection. J Theor Appl Inf Technol August 2014;66(1).

[11] Maheswari S, Karpagam GR. Enhancing Fuzzy Topsis for web service selection. Int J Comput Appl Technol,Inderscience 2015;51(4).

[12] Maheswari S, Karpagam GR. QoS based efficient web service selection. Eur J Sci Res 2011;66(3):428–40.

[13] Maheswari S, Karpagam GR. Applying logical scoring preference method for semantic web service selection. Int J Comput Appl 2013;65(19) (0975 – 8887.

[14] Surendiran B, Vadivel A. Feature selection using stepwise ANOVA discriminate analysis for mammogram mass classification. ACEEE Int J Signal Image Processing 2011;2(1):17–19.

[15] Chun M, Long Y, Mendes E. Web usability measurement: comparing logic scoring preference to subjective assessment; 2005. ICWE 2005, LNCS 3579 p. 53–62.

[16] Mamat NJZ, Daniel JK. 'Statistical analyses on time complexity and rank consistency between singular value decomposition and the duality approach in AHP: a case study of faculty member selection. Math Comput Model 2007;46(7-8):1099–106.

[17] Mohamed KH, Mohamed H, Mousa A. Framework for adaptive QoS of web services using replication'. Int J Comput Sci Commun Netw 2000;2(2):288–94.

[18] Manoharan R, Archana, Siddhika Nitin A, Cowlagi. Hybrid web services ranking algorithm IJCSI'. Int J Comput Sci Issues 2011;8(3):2.

[19] Paolucci M, Kawamura T, Payne TR, Sycara K. Semantic matching of web services capabilities. In: ISCW'12 Proceedings of the first international semantic web conference on the semantic web; 2002.

[20] Pannerselvam R. Design and analysis of experiments. Delhi: PHI Learning; 2012.

[21] Sathya, M, Swarnamugi, M, Dhavachelvan, P, Sureshkumar, G,"Evaluation of QoS based web-service selection techniques for service composition", Int J Softw Eng (IJSE) , Volume (1):Issue 5

[22] Chen Y. Qos-based semantic web service selection Thesis. Toronto, Ontario , Canada: Ryerson University; 2010.

[23] Wang Y, Vassileva J. Toward trust and reputation based web service selection: a survey multi-agent and grid systems. J (MAGS) 2007 special Issue on New tendencies on Web Services and Multi-agent Systems.

[24] Zibin Z. 'QoS management of Web services' Thesis. Hong Kong: the Chinese University; 2011.

S. Maheswari is working as an Assistant Professor(SG) at National Engineering College, India.She received PhD from, Anna University. Her current research interests include Semantic web service selection, Computer Networks and SoA.

G. R. Karpagam received her PhD degree from, Anna University, Chennai and working as an Professor in PSG College of Technology, India. Her research interests include Service Oriented Architecture, Cloud computing and Semantic Web.