

PAPER • OPEN ACCESS

Semantic SLA for Selecting SAAS ERP

To cite this article: A Hijriani and R Sarno 2019 *J. Phys.: Conf. Ser.* **1338** 012056

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Semantic SLA for Selecting SAAS ERP

A Hijriani¹ and R Sarno²

¹Computer Science Department, Lampung University, Bandar Lampung, Lampung, Indonesia

²Department of Informatics, Sepuluh November Institute of Technology, Surabaya, East Java, Indonesia

¹astria.hijriani@fmipa.unila.ac.id; ²riyanarto@if.its.ac.id

Abstract. Service Level Agreement (SLA) is a contract between a service provider and a service consumer. It consists of quality of service (QoS) property and the business properties. In a business environment using Software as a Service (SaaS), SLA and QoS play an essential role in their promising service performance. In this case, the attributes of QoS specified in SLA should show the non-functional attributes of a web service; yet, there, so far, have not been any QoS statement in web service definition language (WSDL) file. This research proposes the selection of SaaS ERP based on the semantic SLA developed using ontology. Analytic Hierarchy Process (AHP) method is employed to establish the rank of a tenant's SLA. Subsequently, to create the rank of SLA which has the same AHP scores, the weighted tree algorithm is used. The results show that the combined method (AHP-Ontology) can provide precision reaching at 0.792, better recall at 0.795 and better f-measure at 0.793. Also, the best accuracy result through this method can reach 86.23%, better than AHP-Euclidean method - only reaching the best accuracy level at 59.03%.

1. Introduction

SaaS is on-demand software that offers several advantages and flexibility with ease of cross platform. In it, web service providers and tenants are two significant roles [1]. Web service providers are parties to build and publish a web service WSDL file. The survey conducted by Gartner, Inc shows that SaaS revenue in 2018 reaching USD. \$ 72.2 billion. What is more, this number is predicted to grow until entering the USD. \$ 113.1 billion by 2021.

The potential of this growing business encourages SaaS developers to attempt to give the best efforts to create technology that can expand and offer reliable and cost-effective services with Multi-Tenancy Architecture (MTA). SaaS model with MTA has a customization mechanism that puts SaaS tenants possibly to have packages that suit their needs. SaaS customization is not only on the functional side but also from non-functional or Quality of Service (QoS). For example, a tenant may require applications with high availability and are willing to pay a cost, but another tenant may not be so prioritized high availability and price put forward. In response, SaaS providers must balance the functional and non-functional requirements of SaaS [2].

In the web-based system service, an agreement on non-functional parameters between the provider and the consumer is poured into SLA - a contract between service provider and service consumer to ensure service consumer to have a paid service and the fulfillment of the requirements of the service. SLA also includes performance information from the web service it represents [1].



The problem faced is the lack of information of non-functional or quality of the Web service in WSDL document [3]. Then, a series of web service will build a SaaS package. WSDL only contains information about the functional requirements of a Web Service, therefore many researchers develop a variety of studies to add non-functional information such as [1], [4-9]. Nevertheless, there is no standard agreed so far on how to specify the number of non-functional requirements in a web service-based system. Another challenge in this research is how to determine the quality of the items necessarily to be included in the SLA. Research [1], [10-15] provide different information about what items should be included in SLA [4-9].

Research [9] provides one of the methods to add annotation on the web service by grouping web service into service profile, service model and service grounding. The way for an SLA research for web service is categorized into three: WSDL extensions, Universal Description Discovery and Integration (UDDI) extensions and broker architecture [1]. WSDL extension requires the development of web services in the manner and annotations for the whole web service provider. Adding SLA annotations in UDDI extension also needs an expansion after the UDDI repository was not further developed. In the broker architecture, the most commonly used method; each web service from the provider will be registered on a specific registry, added by the SLA before stored in the ontology repository.

In this research, domain ontology is used to create SLA documents and to identify web service needed on a SaaS Enterprise Resource Planning (ERP) package [16,17]. SaaS ERP here can be selected as specified packages, each of which contains a series of web services that make variations suitable to the ERP business processes. On the other hand, each web service has been given some additional documents by SLA offered by the SaaS ERP provider. The matching process of a package in suiting the tenant desires will involve various QoS properties included in every web service to be a complicated process. The AHP method is used to deal with multi-criteria decision-making process [18]. Broker architecture using AHP is also used in [3], [19-26]. The study is only applied on quality items of the numeric data type. AHP approach for different data types proposed by [23] with the simulation calculations on 5 SLA multi-criteria is listed on a certain web service. However, AHP multi-criteria calculations performed in [23] can produce the same score of AHP for multiple web service. Hence, we propose the addition of a weighted tree similarity algorithm [26-27] to represent the electoral weight of tenant demand. AHP method and weighted tree similarity will be implemented on broker architecture using the concept of Ontology Web Language for Service (OWL-S) in the broker repository.

This paper will be organized into four sections. Section 1 describes the motivation and several related works. Section 2, furthermore, is to describe the data, framework and methods used in this study and Section 3 will explain the results and analysis, followed by Section 4, the last part presenting the conclusions and directions for further research on the Web Service SLA of SaaS ERP.

2. Data and Methods

2.1 SLA Ontology Architecture

Broker architecture used on SaaS ERP package selection has positioned the brokers to be the parties acting on matching the tenant demand and the supply from SaaS provider. This architecture consists of SaaS Provider, Tenant, Service, SLA repository, and SLA Matchmaker. The service provider, on the other hand, is a party advertising a web service based on the required ERP functionality and accessed as a WSDL file by other application.

In this research framework, a provider also uploads web service SLA to add a non-functional attribute that is not included in the WSDL file. Each service provider registers a web service and SLA based on business agreement in which each of them will then be stored in the repository in OWL-S form. Figure 1 shows the broker architecture.

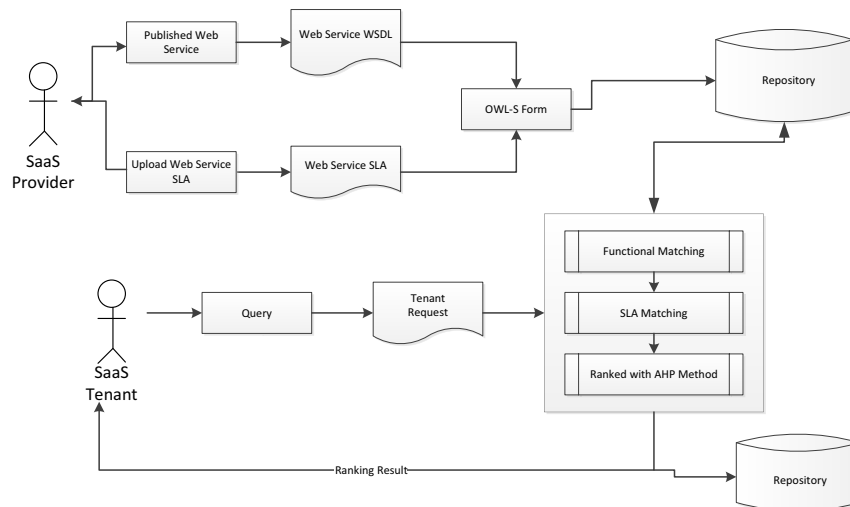


Figure 1 The Broker Architecture for SLA Ontology

2.2 Functional Matching Module

Web Service and SLA are stored in the repository in OWL-S form. In the first stage, functional matching, web service is selected based on the domain ontology of SaaS ERP package. The selected web service is the one in which its service name is indexed on domain ontology, built by Protégé software that, in turn, will describe which web service will be included in one package chosen by a tenant. Figure 2 shows the illustration between package and business process class.

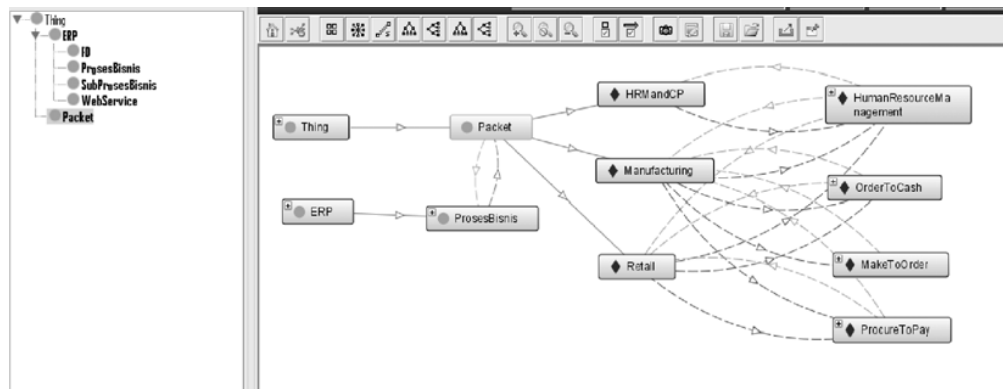


Figure 2 Relation between packet class and business process class

Each business process has some sub-business processes [28], each of which uses one or more web service. The term of web service was used as the selection criteria. In OWL-S form, each web service has a service profile, service grounding, service model and SLA branch. In this case, the SLA branch will be used for the next process.

2.3 SLA Matching Module

SLA information will be stored in the format of the ontology domain as illustrated in Figure 3.

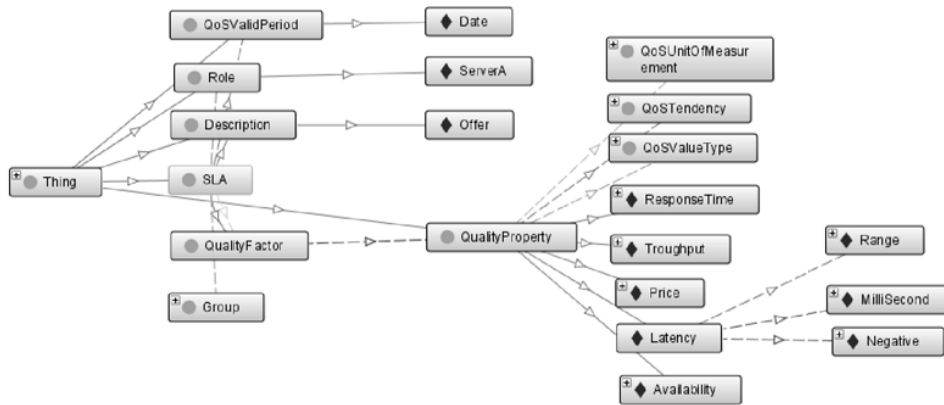


Figure 3. SLA information in Ontology form

The SLA information for each of web service will determine the alternatives in order according to the tenant request. Based on the tenant form, we could obtain some criteria as described in Table 1 for the request.

Table 1. Tenant Request

QoS Group	Mandatory			Optional	
Group Weight	0.6			0.4	
QoS Property	Price	Availability	Response Time	Latency	Server Location
QoS Weight	0.5	0.167	0.33	0.5	0.5
Unit	USD	%	ms	ms	America/Europe
Value Type	Numeric	Numeric	Range	Range	Boolean
QoS Tendency	negative	Positive	-	-	-
Requested QoS Value	-	-	0.022-0.032	144-152	America/Europe

In multi-criteria decision making, tenants tend to compare the brands and products of interest in the list of considerations or evoked set. The list of consideration refers to the number of alternatives that appeal to consumers during the selection process criteria. The amount present in this list may be less than the total choices or options available. Thus, the option that has the closest criteria number with the list of the tenant desire will be evaluated as a better option [23-26]. At this stage, the weighted tree similarity algorithm is used. This algorithm will select the Web Service which has the closest SLA desired by tenants. Figure 4 and Figure 5; show the weighted tree chart from tenant request and provider offer, respectively.

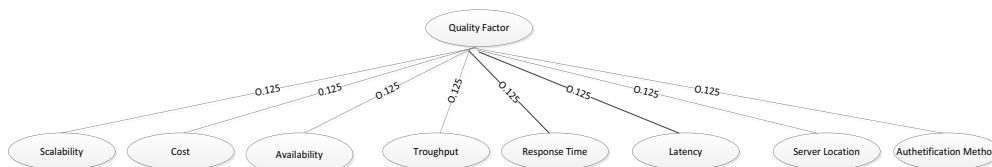


Figure 4 Tree for SLA Offer from SaaS Provider

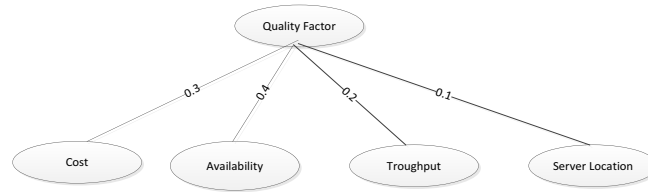


Figure 5. Tree for SaaS tenant request

The equation for weighted tree similarity is formulated as follows [27].

$$\sum(A(S_i)(w_i + w'_i)/2) \tag{1}$$

According to equation 1, the similarity value for the supply and tenant demand as presented in Figure 4 and Figure 5 can be formulated as follows.

$$\begin{aligned} &= \sum (0((0.125+ 0)/ 2) + 1((0.125+ 0.3)/ 2) + 1((0.125+ 0.4)/ 2) + 1((0.125+ 0.2)/ 2) + 0((0.125+ 0)/ 2) + \\ &0((0.125+ 0)/ 2) + 1((0.1+ 0.125)/ 2 + 0((0.125+ 0)/ 2)) \\ &= 0+ 0.425+ 0.525+ 0.325+ 0+ 0+ 0.225+ 0 \\ &= 0.75 \end{aligned}$$

2.4 Analytic Hierarchy Process Ranking Method

The next process is to perform a ranking process using AHP. Let v, v_R^Q be the value of QoS property, Q of the service S_i and the request R , respectively; w_R^Q be the normalized weight of the property Q in its group specified in the request R and E^Q be the enumeration of Q if the value type of Q is an enumeration. Furthermore, let $srrm_{ij} = S_i^Q/S_j^Q$ denotes the service relative rank of the service S_i over the service S_j according to the property Q . For all QoS property Q , we have $S_i^Q/S_j^Q = 1/S_j^Q/S_i^Q$ [23].

There are four cases: the first three special cases are applied to all QoS properties in spite of their value types and tendency while the fourth case is concretely considered.

- i. $v_{S_i}^Q$ is unspecified and $v_{S_j}^Q$ is specified: $S_i^Q/S_j^Q = w_R^Q$. When $v_{S_i}^Q$ is not specified, we cannot directly compare S_i to S_j . Therefore, we use the normalized weight w_R^Q to indicate; in this case S_j is better than S_i according to QoS property Q .
- ii. $v_{S_i}^Q$ is specified and $v_{S_j}^Q$ is unspecified: $S_i^Q/S_j^Q = 1/w_R^Q$. This case is similar to (i), and S_i is better than S_j according to Q .
- iii. $v_{S_i}^Q$ and $v_{S_j}^Q$ are unspecified: or $v_{S_i}^Q \equiv v_{S_j}^Q : S_i^Q/S_j^Q = 1$
- iv. $v_{S_i}^Q$ and $v_{S_j}^Q$ are specified. In this case, we need several different ways to compare S_i to S_j according to the value type and the tendency of the QoS property Q .

Table 2 below presents an example of the calculation that is about a list of SLA for one of the web services. Five different SLAs for the same function of web service is involved. The advertised SLA will further be calculated against Table 1 with the AHP method.

Table 2. Advertised SLA

QoS Property	Price	Availability	Response Time	Latency	Server Location
QoS Weight	0.2	0.2	0.2	0.2	0.2
Unit	USD	%	ms	ms	America/Europe
Value Type	Numeric	Numeric	Range	Range	Boolean
SLA 1	4	84	0.021-0.030	141-155	America/Europe
SLA 2	3	83	0.026-0.030	142-150	Non-America /Europe
SLA 3	4.5	85	0.023-0.028	143-150	America/Europe
SLA 4	3.4	84	0.020-0.035	145-150	Non-America /Europe
SLA 5	3.7	88	0.021-0.028	140-148	America/Europe

2.4.1. Calculating Service Relative Ranking Matrix

The first step is to calculate Service Relative Ranking Matrix for each of QoS Property.

- Price Property
Here, Service Relative Ranking Matrix Calculation is a rule used for negative numerical calculations as the tenant does not provide query value but provides a negative tendency.

- Negative Tendency [23]

$$S_i^Q/S_j^Q = v_{S_j}^Q/v_{S_i}^Q \tag{2}$$

- Availability Property
This QoS property still uses the similar calculation rules to price because the tenant provides positive tendency as the basis for the ranking.

- Positive tendency [23]

$$S_i^Q/S_j^Q = v_i^Q/v_j^Q \tag{3}$$

- Response Time Property
Following rules are used to calculate range data type.

$$S_j^Q = \begin{cases} w_R^Q & \text{if } v_{S_i}^Q \cap v_R^Q \equiv \emptyset \wedge v_{S_j}^Q \cap v_R^Q \neq \emptyset \\ \frac{1}{w_R^Q} & \text{if } v_{S_i}^Q \cap v_R^Q \neq \emptyset \wedge v_{S_j}^Q \cap v_R^Q \equiv \emptyset \\ 1 & \text{if } v_{S_i}^Q \cap v_R^Q \equiv \emptyset \wedge v_{S_j}^Q \cap v_R^Q \equiv \emptyset \\ \frac{\text{len}(v_{S_i}^Q \cap v_R^Q)}{\text{len}(v_{S_j}^Q \cap v_R^Q)} & \text{if } v_{S_i}^Q \cap v_R^Q \neq \emptyset \wedge v_{S_j}^Q \cap v_R^Q \neq \emptyset \end{cases} \tag{4}$$

For range data type, it is better to evaluate when the slice is more prominent than SaaS tenant demand.

- Latency Property
In this study, the latency is defined as range data type. Here, calculation rules for Service Relative Ranking matrix are similar with that of the response time.
- Server Location Property

Server Location is defined as a Boolean data type. It involves two choices: “America and Europe” and “non-America/Europe”. The rules for Boolean value comparison are formulated as follows [23].

$$S_i^Q/S_j^Q = \begin{cases} w_R^Q & \text{if } v_{S_j}^Q \equiv v_R^Q \\ \frac{1}{w_R^Q} & \text{if } v_{S_i}^Q \equiv v_R^Q \end{cases} \quad (5)$$

At this stage, Service Relative Ranking Matrix will be evaluated using an assumption that if the value is the same as tenant request, then it will be estimated as better. The weight for each category of Server Location is 0.5.

2.4.2. Calculating Group Relative Ranking Matrix

At the second step, it is done by calculating Group Relative Ranking Matrix.

- mandatory group
 $grrm_{ij} = (srrm_{ij} \text{ cost} * \text{weight}) + (srrm_{ij} \text{ availability} * \text{weight}) + (srrm_{ij} \text{ response time} * \text{weight})$
- optional group
 $grrm_{ij} = (srrm_{ij} \text{ latency} * \text{weight}) + (srrm_{ij} \text{ Server Location} * \text{weight})$

2.4.3. Calculating Final Service Ranking

After obtaining Group Service Relative Ranking Vector, the next step is to combine it to get Final Service Ranking. Weight for mandatory and optional is set by default 0.6 and 0.4.

$$frrm_{ij} = (grrm_{ij} \text{ mandatory} * \text{weight}) + (grrm_{ij} \text{ optional} * \text{weight})$$

In this study, calculations on the quality properties that have a string data type are also conducted and formulated as follows:

$$S_i^Q/S_j^Q = \begin{cases} w_R^Q & \text{if } sim(v_{S_i}^Q, v_R^Q) \equiv 0 \wedge sim(v_{S_j}^Q, v_R^Q) \neq 0 \\ \frac{1}{w_R^Q} & \text{if } sim(v_{S_i}^Q, v_R^Q) \neq 0 \wedge sim(v_{S_j}^Q, v_R^Q) \equiv 0 \\ 1 & \text{if } sim(v_{S_i}^Q, v_R^Q) \equiv 0 \wedge sim(v_{S_j}^Q, v_R^Q) \equiv 0 \\ \frac{sim(v_{S_i}^Q, v_R^Q)}{(v_{S_j}^Q, v_R^Q)} & \text{if } im(v_{S_i}^Q, v_R^Q) \neq 0 \wedge sim(v_{S_j}^Q, v_R^Q) \neq 0 \end{cases} \quad (6)$$

Similarity calculation for String data type is calculated using Symmetric for Similarity Measure library.

2.5. AHP-Euclidean Distance Method

The rating also will be ranked with AHP-Euclidean Distance to make a comparison with the models of calculation, [3]. Such ranking step using AHP-Euclidean Distance is to form a matrix of performance for the normalized value of each property. P is a matrix with the performance, and then every property is normalized by equation 7.

$$q_{ij} = \frac{p_{ij}}{\sqrt{\sum_{k=1}^n p^2_{ik}}} \quad (7)$$

Equation 8 is the equation for Euclidean distance based on value request is used to compare with tenant demand. Euclidean distance shows that the closer the distance, the higher the SLA rank.

$$E_j = \sqrt{\sum_{i=1}^m \left(v_{ij} - w_i r_i / \sqrt{\sum_{i=1}^m P_{ij}^2} \right)^2} \tag{8}$$

3. Results and Discussion

Thirty-six simulated servers have been prepared for this research (appendices). All servers represent twelve different providers as SaaS provider. Each provider offers three different SLA categories: high, medium and low. These categories, besides serving web service capabilities and price, also show the number of the involved QoS properties. Table 3 shows three samples for providers that offer an SLA on High Categories. Table 4 shows three categories of examples provided by one provider.

Table 3. SLA from Three Different Providers in High Category

SLA Name	Cost	Availa-bility	Scalabi-lity	Response Time	Throughput	Latency	Authenti-cation Method	Server Loc.
SLA A	40	98	0.95	0.001-0.010	575-1425	100-130	One-time password	America / Europe
SLA E	37	98	0.9	0.004-0.010	550-1125			Non-America /Europe
SLA I	29	95	0.89	0.006-0.011	575-986			

Table 4. SLA Offering from Provider A

SLA Name	Cost	Availa-bility	Scala-bility	Response Time	Throughput	Latency	Authenti-cation Method	Server Loc.
SLA A	40	98	0.95	0.001-0.010	575-1425	100-130	One-time password	America/ Europe
SLA M	18	90	0.89	0.011-0.020	535-986	125-140	One-time password	America/ Europe
SLA Y	4	84	0.85	0.021-0.030	450-874	141-155	One-time password	America/ Europe

Testing is conducted to prove that the ranking with AHP-Ontology and weighted tree similarity can have a better result either in precision, recall or accuracy. Here, counting precision and recall is formulated as follows.

$$\text{Precision} = \frac{\text{Relevant retrieved}}{\text{Retrieved}} \tag{9}$$

$$\text{Recall} = \frac{\text{Relevant retrieved}}{\text{Relevant}} \tag{10}$$

$$\text{F-Measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{11}$$

Twenty-four types of queries are performed on the system. These queries performed a variety of weights, QoS properties and different data types. Also, a relevant list for each of tenant requests is made in which each list contains 12 of the most relevant SLA according to the tenant requests. This list, in turn, will be compared to AHP-Ontology ranking method result and AHP-Euclidean ranking result. The ranking result also determines false positive (FP), false negative (FN), true positive (TP), and true negative (TN) area. The tenant requests can be categorized as follows.

- A tenant with a request that assigns some numerical values
For tenant P2, P9 and P11, both AHP-Ontology and AHP-Euclidean Distance give the same ranking.
- A tenant with requests that only provide tendency values
For tenant P1, P4, P6, P7, P12, P19, P20, P24, AHP-Ontology method results in a better precision result. While, AHP-Euclidean distance method could not give ranking method close to the relevant list of SLA.
- A tenant with requests that provide value to non- numerical values
For tenant P23, only AHP-Ontology method could give the result.
- A tenant with requests that provide numerical and non-numerical values and tendency
For tenant P3, P5, P8, P10, P13, P14, P15, P16, P17, P18, P21, and P22, AHP-Ontology also gives a better precision result.

Table 5 shows that precision, recall and F-measured for AHP-Ontology method are better than that of AHP-Euclidean method. For some QoS properties requests, the SaaS tenant demand needs a concern in that AHP-Ontology method is more sensitive to the demand. For example, if the tenant requires a lower price, then it can choose a negative tendency to the price property.

Table 5 Comparison of Precision, Recall and F-Measure

	<i>AHP-Euclidean</i>	<i>AHP-Ontology</i>
Precision	0.403	0.792
Recall	0.406	0.795
f-measure	0.404	0.793

The ability of AHP-Ontology method to handle various data types also becomes the reason for the high value of f-measure when using the method.

Based on positive and negatives areas, we made true positive rate and false positive rate comparison for both methods. These areas are also calculated for an accuracy level of AHP-Ontology method with the following formula.

$$\text{Accuracy} = \frac{TP+TN}{(TP+TN+FP+FN)} \tag{12}$$

To determine the better accuracy of the method, true positive and false positive rate curve could be used. Further away from the diagonal line, the accuracy is better. Figure 6 shows that AHP-Ontology method can provide a better true positive rate, far from the diagonal line of comparison rate.

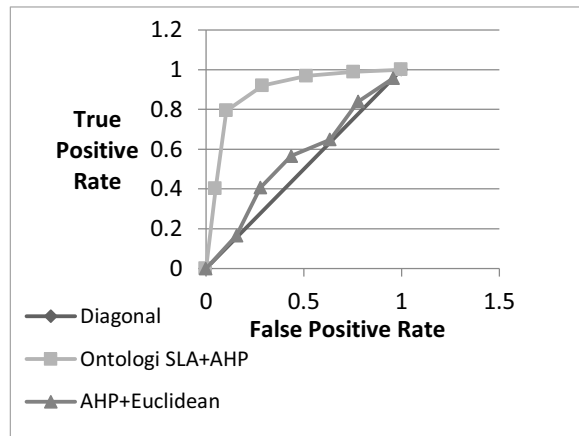


Figure 6. True Positive Rate

Figure 7 shows true positive area based on FP-TN frequency and TP-FN frequency in a normal curve. In this case, we can provide a threshold value that will determine a particular area. If moved into the positive region (right), the threshold value will affect the number of true positives that are drawn. Meanwhile, if the threshold value is moved into the negative region (left), the number of true positives will be more then, but the number of true negatives will also be more and drawn. The threshold value can be taken from the intersection of the positive and negative curve to achieve a maximum accuracy value.

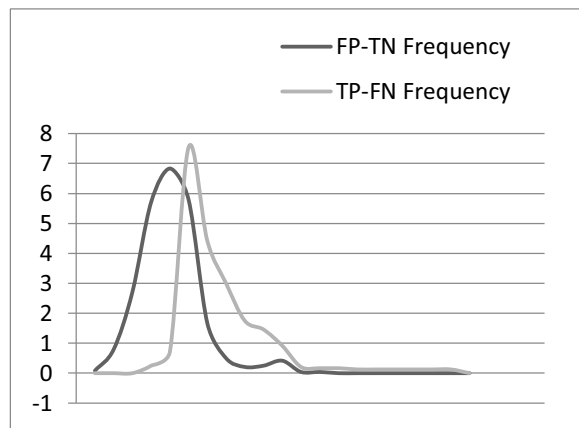


Figure 7. Normal Curve for FP-TN and TP-FN Frequency

In [29], as illustrated in Figure 8, the level of accuracy value diagnostic is grouped. Here, AHP-Euclidean method results in the best accuracy level at 59.03 % when the appropriate list contains 6 of the most relevant SLA. However, AHP-Ontology method gives a better result in 6 and 12 categories of the most relevant SLA with 86.23% of the accuracy level.

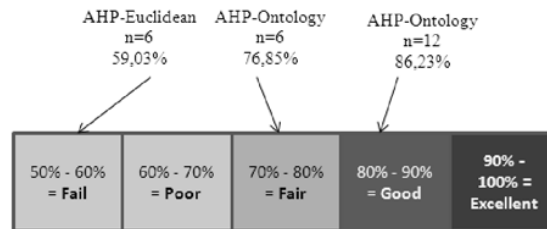


Figure 8 Accuracy Level Classification

Since this research work was conducted on laboratory data source, we still found lack in it. At the request of tenants P23 and P24 in which tenants made a demand for the most appropriate Authentication Method and Availability, the system would restore to rank 3 SLA with the same AHP score, 0.3697 for A, M and Y. The situation with the same AHP score is the situation where the weighted tree similarity will be used. The system in broker architecture will recommend SLA with a higher weighted tree similarity value as a priority in the order. The higher similarity value with the weighted tree represents consumer behavior in evaluating the alternatives, meaning closer to the tenant request.

Furthermore, some lacks occur when we must define the priority level for the same AHP score, and the corresponding weighted tree similarity score. In this case, we need further research using the real data in SLA advertisement and provider weighting for a better result as well as the implementation of the algorithm for real deployed service of SaaS ERP.

4. Conclusions

In this paper, we initially have studied some related researches and proposed multi-criteria decision making using AHP method and weighted tree algorithm. Both methods were used as a QoS-based ranking algorithm to support web service selection using ontology annotation. The use of AHP method type was by considering that it can offer various evaluation strategies for different data types, values and tendencies, depending on QoS properties. The algorithm also supported grouping and prioritizes specific QoS properties. Some simulated servers, subsequently, were created to show a differentiation between the user requirements and the system.

The result shows that our AHP-ontology approach can result in better precision (0.792), better recall (0.795) and better f-measure (0.793). Meanwhile, AHP-Euclidean method working on the same data shows 0.403 for precision, 0.406 for recall and 0.404 for f-measure. True positive and false positive rate curve illustrated that AHP-ontology method could result in a better accuracy level since its line is further away from the diagonal line. Based on the classification of accuracy level, our AHP-ontology method shows a good result reaching 86, 23% much higher than that of AHP-Euclidean method - only reaching 59, 03%, meaning that the classification is failed.

For future works, we will implement the ontology and ranking model to the deployed SaaS Services - not only simulated server. From an architecture view, the real time monitoring of quality property will increase reliability. On the other hand, in case of implementation to the real world, some flexibility for QoS property addition and the real advertised SLA will be needed. Further, we must improve the method by using more varieties in each of QoS property weight from the provider.

Appendices

SaaS Tenant	Price	Availability	Scalability	Weight and Value					Authentication Method	Server Loc.
				Response Time	Throughput	Latency	Throughput	Latency		
P1	w=1 negative									
P2	w=1, 10									
P3	w=0.3, 10	w=0.1, 92	w=0.1, 0.855	w=0.1 0.015-0.021	w=0.1 480-900	w=0.1 118-135	w=0.1 one-time password	w=0.1 America-Europe		
P4	w=1 positive									
P5	w=0.2 negative									w=0.8 America-Europe
P6	w=0.5 negative									
P7	w=0.7 negative									
P8	w=0.3 10	w=0.2 92	w=0.1 0.855	w=0.1 0.015-0.021	w=0.1 480-900	w=0.1 118-135	w=0.1 one-time password			
P9	w=0.7 10	w=0.3 92								
P10	w=0.4 negative	w=0.3 positive	w=0.3 positive							
P11	w=0.4 10	w=0.3 92	w=0.3 0.855							
P12	w=0.2 negative	w=0.6 positive	w=0.2 positive							
P13	w=0.4 10	w=0.3 92		w=0.3 0.015-0.021						
P14	w=0.3 positive	w=0.5 positive							w=0.2 one-time password	
P15	w=0.3 10	w=0.3 92			w=0.4 480-900					

SaaS Tenant	Weight and Value							
	Price	Availability	Scalability	Response Time	Throughput	Latency	Authentication Method	Server Loc.
P16	w=0.2 10	w=0.2 92		w=0.2 0.015-0.021			w=0.2 one-time password	w=0.2 America-Europe
P17	w=0.2 10	w=0.2 92	w=0.2 0.855	w=0.2 0.015-0.021	w=0.2 480-900			
P18	w=0.1 10	w=0.3 92	w=0.1 0.855	w=0.1 0.015-0.021	w=0.1 480-900	w=0.1 118-135	w=0.1 one-time password	w=0.1 America-Europe
P19	w=0.3 negative	w=0.7 positive						
P20	w=0.2 positive	w=0.6 positive	w=0.2 positive					
P21	w=0.2 positive	w=0.2 positive					w=0.6 one-time password	
P22		w=0.2 positive					w=0.8 one-time password	
P23							w=1 one-time password	
P24	w=0.5 positive	w=0.5 positive						

References

- [1] Kyriakos K Dimitris P Pierluigi P 2016 Semantic SLAs for Services with Q-SLA *Proc. Computer Science* (Madrid: Elsevier) pp 24-33
- [2] Virginia M A Jose A V 2013 Business and Technical Requirements of Software as a Service : Implications in Portuguese Enterprise Business Context *Int. J. in Found. of Comp. Sci. and Tech. (IJFCST)* **3** 1-14
- [3] Eleyan A Liping Z 2011 Service Selection Using Quality Matchmaking *Proc. of 2011 Conf. on Service Communication and Information Technology (ICCIT 2011)* (U.S.A : Institute of Electrical and Electronics Engineers) pp 107-115
- [4] Christopher R Ivan B Ivona B Schahram D. 2012 Automatic SLA Matching and Provider Selection in Grid and Cloud Computing Markets *2012 ACM/IEEE 13th Int. Conf. on Grid Computing* (Beijing : IEEE) pp 85-94
- [5] Haiteng Z Zhiqing S Hong Z Jie Z 2014 Web Service Reputation Evaluation Based on QoS Measurement *The Sci. World J.* **2014** 1-7
- [6] Chiraz E Raoudha B Ikram A 2014 A User-Aware Approach to Provide Adaptive Web Services *J. of Univ. Comp. Sci.* **20** 944-963
- [7] Radwa E Mohamed A 2015 A Framework for Negotiating Service Level Agreement of Cloud-Based Services *Proc. Computer Science* **65** 940-949
- [8] Ramsha B Waqas A Irfan U Irfan M 2017 Agent-Based SLA Negotiation Protocol for Cloud Computing *2017 Int. Conf. on Cloud Computing Research and Innovation (ICCRRI)* (Singapore: IEEE) pp 33-37
- [9] David M Mark B Drew M Sheilla M Massimo P Katia S Deborah L Evren S Naveen S 2007 Bringing Semantics to Web Services with OWL-S *J. World Wide Web* **10** 243-277
- [10] Sharavanan K Silas S Rajaram M 2016 An Automated Semantic Negotiation for Cloud Service Level Agreements *Circuits and Systems* **7** 2443-2451
- [11] Hanane B Daniel A 2014 Consumer-Centric Non-functional Properties of SOA-based Services *Proceedings of the 6th International Workshop on Principles of Engineering Service-Oriented and Cloud Systems* (Hyderabad: ACM) pp 18-27
- [12] Krunal D Narendra J 2013 Cloud Based Conceptual Framework of Service Level Agreement for University *Int. J. of Computer Applications* **65** 16-21
- [13] Haider A Shouman B 2018 Service Level Agreement (SLA) for Cloud Computing Compilation with Common and New Formats *Int. J. Scientific Research and Management* **6** 2018-01-09
- [14] Anjana G Sangeeta S Rolly G 2013 Non-Functional Requirement Classification for Service Oriented Data Warehousing *Int. J. of Scientific and Research Publications* **3** 1-6
- [15] Mengmeng L Peng L 2017 Automatic Classification of Non-functional Requirements from Augmented App User Review *Proceedings of the 21st Int. Conf. on Evaluation and Assessment in Software Engineering* (Karlskrona: ACM) pp 344-353
- [16] Demian A Ivnet K Namratha R Ananthanarayana V 2009 Semantic web service selection based on provider's business offering *IJSST* **10** 25-37
- [17] Sophea C Nejib M Yacine 2016 QoS Ontology for Service Selection and Reuse *J. of Intel. Manufacturing* **27** 187-199
- [18] Thomas L 2008 Decision making with the analytic hierarchy process *Int. J. Serv. Sciences* **1**(1) 83-98

- [19] Duygu C Atilla E 2008 Semantic QoS model for extended IOPE matching and composition *In Proceedings of 32nd Annual IEEE Int. Comp. Conf. on Software and Application 2008 COMPSAC'08* pp 993-998
- [20] Usha K Thamarai S 2018 Efficient Agent-based Negotiation by Predicting Opponent Preferences using AHP *J.Appl. Res.Technol.* **1**
- [21] Zar L 2016 SLA Guaranteed Migration Decision Model Using MCDM Approach *Genetic and Evolutionary Computing* (Cham: Springer International Publishing) p 369-379
- [22] Raju R Dhivya D Saranya R Abbinaya S 2013 SLA Negotiation for Web Service Consumption Based on Analytic Hierarchy Process *Int. J. of Sci. And Mod. Eng. (IJISME)* **1(3)** 72-75
- [23] Vuong X Hidekazu T Ryosuke M 2009 A new QoS Ontology and its QoS-based Ranking Algorithm for Web Services *Simul. Mod. Pract. Theory* **17**:1378-1398
- [24] Chunqi S Donghui L Toru I 2012 User-Centered QoS Computation for Web Service Selection *2012 IEEE 19th Int. Conf. on Web Services* p 456-463
- [25] Arezoo J Khanli L Seyed N 2014 W_SR: A QoS Based Ranking Approach for Cloud Computing Service *Comp. Eng. And Apps. J.* p 55-62
- [26] Ruby A Aisha B Subash P 2016 Comparison of Multi Criteria Decision Making Algorithms for Ranking Cloud Renderfarm Services *Indian Journal of Science and Technology* **9(31)** 1-5
- [27] Riyanarto S Faisal R 2008 Weighted Tree Similarity Algorithm Implementation for Semantic Search *Scientific Journal of Information Technology* **7(1)** 39-46
- [28] Anang K Riyanarto S Dwi S Imam M 2013 Semantic Web Service Composition for ERP Business Process *Kursor J.* **7(1)** 29-38
- [29] Florin G 2011 Data Mining Concept Model and Techniques Vol 12 (Berlin:Springer) p 328-330