Hybrid Web Services Ranking Algorithm

R.Manoharan, A. Archana and Siddhika Nitin Cowlagi

Pondicherry Engineering College
Puducherry, India

Pondicherry Engineering College
Puducherry, India

Pondicherry Engineering College
Puducherry, India

Abstract

Web services are dynamic applications that are published over the network to help in data exchange between systems and other applications. The selection of these web services is an important part of Web service composition. As the number of Web services on the internet increase, the need for finding the exact web service that matches the user's request also increases. So ranking of web services is required in order to find the right web service. Earlier methods of ranking involved using a matrix to rank the web services by using their QoS property. But in many cases it might be tedious to define the QoS values accurately. So a fuzzy logic was proposed to deal with the improper QoS constraints. Many approaches have been proposed both in Quality based and fuzzy based ranking. In this paper we take the advantages of both the methods and propose a new ranking method which is a hybrid of matrix ranking method and QoS based fuzzy ranking method. We also review the existing works done and summarize the benefits and conclusions of the presented ideas.

Keywords: *web services, matrix, fuzzy, entropy, Quality of Service.*

1. Introduction

Web services are being widely used throughout the world because of the advantages they provide. This is due to the fact that they help in developing distributed applications that can be accessed across the internet. They use the three basic platform elements: SOAP, WSDL and UDDI. It is necessary to discover and select suitable web services based on client's requests. As in [1], WSDL standard is used to describe the web services along with its functionalities and the services it provides. Along with description it provides an end point to invoke the web service. In order to make web services available to the clients, every web service must be published in some UDDI [2] registries. When request for a web services is issued the service descriptions are matched with the request description to find the web service with exact functionality. In order to issue the request a SOAP protocol is used.

A major application of these web services is web service composition. In many cases when individual web services are not able to satisfy the complex requirements of an application they are combined construct applications which meet the needs of the user. In order to compose such web services, it is very essential to select those which truly meet the criteria. This brings the need of having some sort of ranking procedures while searching for services taking into consideration the actual requirement of the user. Also when a client requests for a service, there may be many web services that provide the same functionality that the client looks for. Ranking the web services will help a client find out the better web service. In order to differentiate between the web services that provide the same functionality we need to examine the QoS attributes like throughput, availability, response time, performance, latency etc[3].These non functional parameters help to determine the quality aspects of a web service. Service providers should publish their web services by incorporating the QoS parameters and periodically update them during revisions so that they can make their service top the list while ranking. Traditional ranking approaches employ different methods to rank the web services by evaluating a final score using the QoS parameters. Different QoS parameters may have different importance level depending on the client. So QoS parameters are difficult to be defined precisely. In order to overcome this, a new approach based on fuzzy logic was proposed. It deals with selecting services by associating a fuzzy number to each service and selects services based on the fuzzy range as opposed to the earlier binary approach.

This paper deals with a hybrid algorithm that combines the working of matrix based and fuzzy based ranking methods.

2. Literature Survey

Web services are used in the development of numerous web applications in the current scenario. The existing searching techniques for web services mostly concentrate on keyword matching. But this method is now being replaced by newer methods as keyword matching does not yield effective results according to the user's requirements.

A more relevant approach is to provide the users with a list of web services which would satisfy their needs. There are various web services with offer similar functionalities but might have different non-functional or quality of service parameters.

Ranking of web services has been a wide area of research. Various methods are being followed for ranking. The most common is the matrix method approach [4]. This method takes the QoS parameters of various web services as input to a matrix, normalizes it and derives a Web Service Relevancy Function (WsRF) based on it to provide a rank for the services.

In the context based method [5], the WSDLs are analyzed semantically to extract a more accurate result to the user's query. The proximity of the context of similar web services are determined to give the final rank list. Selecting specific sites for context extraction is a shortcoming of this method. The quality driven approach [6] makes use of a quality constraint tree (QCT) with inputs as functionally similar web services. The traversal algorithm performs three actions namely filtering, scaling and ranking to obtain the result.

If we consider the ontology semantics [7], the XML documents of the web services are scanned and the input constraints and output specifications of similar web services are compared. A function to map these is proposed based on which ranking is done. Another method in this approach [8] proposes an ontology OWL-Q, for QoS based web service description. This ontology includes descriptions for the various metrics that a web service provides. It further provides a QoS metric matching algorithm for comparing similar web services.
A semantic search agent approach [9] is also followed in case of searching semantic web services. Here a system is described which uses Ontology Web Language (OWL) to search web services according to the user's need. The disadvantage of this method is that only web services whose semantic descriptions are available in the UDDI registries can be searched. Ontology based retrieval [10] is possible in some cases where a crawler is designed to search the ontologies of the web services to retrieve the require result.

The single value decomposition technique [11] is another method that uses a matrix to represent the QoS attributes. This single matrix is decomposed into three matrices based on various criteria. The values of these matrices are represented in 2D space in the form of a graph and based on the proximity of values the ranks are allocated. The disadvantage here is that for complex attributes the size of the matrix becomes too large.

In another approach, dominance relationships [12] are used for ranking. A balance between the number of dominating services and the number of dominated services is obtained and the results are plotted to obtain a rank for the various services. The dominance scores for both cases are determined using separate algorithms which first calculate the paradigm degree of match (PDM) and then calculate the service degree of match (SDM).

The exact process and algorithms for this approach are presented in [13]. This paper traces the steps involved in arriving at the final dominance score which is used as the criteria to rank services according to their parameters.

In the fuzzy approach for ranking [14], the attributes are converted into a fuzzy constraint specification problem which is then defuzzified to produce a quantifiable result. The fuzzy limits are used to determine if a service can be added to the rank list. Since the approach is fuzzy and not binary, the number of web services considered for filtering is more which an advantage of this method is. The problem in this approach is certain QoS criteria cannot be expressed in fuzzy logic. QoS criteria play a major role in the evaluation of web services for any method. The issues involved in the selection of proper QoS parameters for a web service are elaborated in [15].

In the fuzzy approach, various weights are used for the normalization of the fuzzy values. One such method is the entropy weights [16]. This paper describes the various stages in the calculation of entropy weights and also their relevance in the fuzzy algorithm is analyzed.

2.1 Extract of Literature Survey

There are various existing approaches for the ranking of an available set of web services. Of these methods the most preferred ones are the matrix methods for their simplicity and the fuzzy methods for their accuracy. The above two methods pose various issues in the selection of the right web services for an application. The major advantage of the matrix method is the effective capturing of the needs of the user and the preference to data given. This algorithm is used in most of the crawler engines for effective ranking. But it fails to define certain QoS parameters precisely and thus cannot support complex QoS constraints. The fuzzy method on the other hand stresses on the importance of weights and normalization of the inputs given by the user. The use of fuzzy logic further allows wide number of web services to be taken into account. Another advantage of this method is the fact that imprecise QoS criteria can be defined in fuzzy. However, in fuzzy method, we cannot easily specify which QoS parameter will have an effective fuzzy representation. Our proposed work aims at clubbing the advantages of both these techniques and providing an effective hybrid ranking algorithm. The representation and computation of the QoS values in the hybrid algorithm considers both the crisp and fuzzy form.

3. Proposed Work

Even though there are various ranking methods for web services as described above, each of these methods has disadvantages of its own. In this paper we propose a new algorithm which is a hybrid of matrix and fuzzy based ranking algorithms. Our basic architecture begins with searching the WSDL documents for the related web services and terminates with a ranked list of those services.

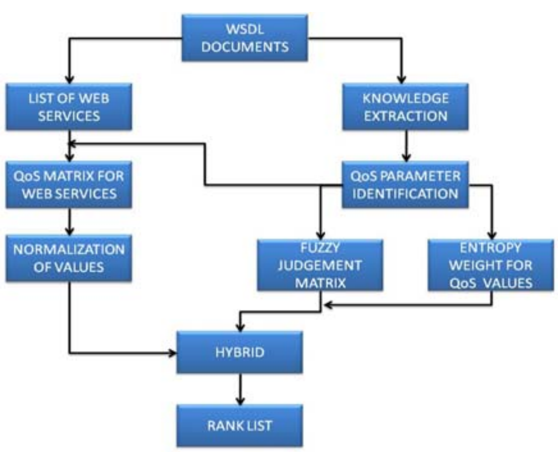

Fig 1: Basic Architecture

The process begins with searching the WSDL documents of the related web services as per the user's query. Each web service registers its WSDL document in the UDDI registries from where we obtain them. These WSDL documents contain the details about the various Quality of Service parameters offered by them. From these the knowledge about the QoS values is extracted. Most of the existing algorithms use a web service crawler engine to obtain this knowledge. Once these values are determined a QoS matrix is constructed with the QoS values of the services as the matrix elements. For each web service the QoS parameters would be in varying units. Also for a valid ranking method, few criteria need to be maximized and few minimized.

Popular ranking methods do not take into account such discrepancies while ranking the web services. To overcome this we have proposed a normalization method which takes into consideration the weight of the QoS parameters and normalizes the QoS matrix accordingly.
Simultaneously we also construct a fuzzy judgement matrix based on the fuzzy limits described for each parameter. An entropy weight is used as a normalization criterion for this fuzzy matrix. The output of this entropy normalization is then introduced in the matrix method.
In this step, the variation of each normalized QoS value in comparison to the entropy weight is determined. This gives a hybrid matrix with the final standardized QoS attributes. From this matrix we compute the web service relevancy score for each service.

The algorithm for computing the relevancy score using hybrid ranking is given below:

Algorithm:

Ranking of web services by a hybrid of matrix and fuzzy based ranking algorithms.

Input:

m candidate web services and n QoS criteria.

Method:

For a set of web services WS = { $ws_1, ws_2, \ldots ws_i$ } that share the same functionality and set of QoS parameters P = { $p_1, p_2, \ldots p_j$ }, the hybrid algorithm computes a web service relevancy score based on the client's request. By examining the QoS values for each web service we get a matrix E where each row represents a web service and each column represents a QoS criteria.

$$E = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,j} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,j} \\ \vdots & \vdots & \vdots & \vdots \\ q_{i,1} & q_{i,2} & \cdots & q_{i,j} \end{bmatrix}$$

Each user might have various QoS requirements as to which are more relevant to the application. The preferences of the user are obtained in the form of binary decisions in an array (say preference[j]). This array will have a '0' for parameters which need to be low and '1' for parameters which should be high.

Since these QoS values are difficult to be defined precisely we introduce fuzzy information which is represented by triangular fuzzy numbers. Triangular fuzzy numbers are nothing but fuzzy limits for each QoS parameter represented as [L, M, N] where L, M, N are the lower, medium and upper values for that QoS parameter.

With these fuzzy limits we calculate a membership function for each matrix value q_{ij} defined as

$$q'_{ij} = \begin{cases} (q_{ij} - L)/(M - L) & if\ L \le q_{ij} \ge M \\ (U - q_{ij})/(U - M) & if\ M \le q_{ij} \ge U \\ 0 & otherwise \end{cases} \quad (1)$$

The actual objective of each criterion is needed to distinguish between various web services which have the same value for a specific parameter. In order to this we introduce entropy weight calculation in the step. When the entropy value increases the information express quality decreases and thus the entropy method serves as an efficient way to achieve an overall assessment of the information quality.

For every j^{th} criterion the entropy weight is defined as

$$E_j = -e * \sum_{j=1}^{m} f_{ji} \ln f_{ji} \quad (2)$$

where e = $1/\ln m$ and

$$f_{ji} = a_{ij} / \sum_{i=1}^{m} a_{ij} \quad (3)$$

The total entropy is given by H = $\sum_{j=1}^{n} E_j$ (4)

The final entropy weight for the j^{th} criterion is defined by 1 - E_j.

The balanced entropy weight for every j^{th} criterion is

$$E_{w_j} = (1 - E_j)/\sum_{j=1}^{m}(1 - E_j) = (1 - E_j)/n - H \quad (5)$$

Now each of the matrix elements need to be compared with the entropy weight to analyze how the QoS values vary from the entropy weights of each QoS parameter. This is done in order to get a uniform overall score for all the candidate web services. The comparison is done based on the following equation:

$$t_{ij} = q'_{ij} / E_{w_j} \quad (6)$$

By doing the comparison for every matrix element we get a matrix as follows:

$$E' = \begin{bmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,j} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,j} \\ \vdots & \vdots & \vdots & \vdots \\ t_{i,1} & t_{i,2} & \cdots & t_{i,j} \end{bmatrix}$$

Every QoS parameter varies from the other in terms of its units and magnitude, so each parameter must be normalized to get the exact relevancy score. It is an essential step as certain values should be high and certain should be low in order to get best results. Normalization helps to achieve uniform distribution over the QoS parameters. This is done based on which QoS criteria needs to be maximized and which needs to be minimized. Normalization can be done by using the formula:

For every matrix element

if(q'_{ij} == ' positive')

$$Y = \frac{X - X_{min}}{X_{max} - X_{min}}$$

else if (q'_{ij} == ' negative')

$$Y = \frac{X_{max} - X}{X_{max} - X_{min}} \quad (7)$$

Where X is the value of each metric considered and X_{min} and X_{max} correspond to the minimum and maximum QoS values offered by the web services. The positive or negative criteria can be obtained from the preference array.

This gives the final normalized QoS matrix from which the relevancy score for each of the candidate web services is calculated. The final matrix obtained is

$$W' = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,j} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,j} \\ \vdots & \vdots & \vdots & \vdots \\ w_{i,1} & w_{i,2} & \cdots & w_{i,j} \end{bmatrix}$$

Once this matrix is computed a relevancy function is derived for each of the web service. This function takes into account all the normalization done on the QoS values offered by each of the candidate services. Finally each web service is assigned a specific rank or score based on this relevancy function value. The relevancy score is calculated as follows:

$$WsRF\,(ws_i) = \sum_{j=1}^{m} t_{ij} \qquad (8)$$

The ranks for the web services are based on the above function with the web service having the highest *WsRF* score being ranked 1st.

4. Experiments and Results

The inputs given for this hybrid algorithm are actual data values which are available over the web and hosted on sites like Xmethods.net etc. Since QoS attributes are the basic criteria for the ranking approach used, we define four such parameters below:

Cost(C): It is the cost of each service request determined by the service provider (unit: cents).

Availability (AV): It is the measure of the uptime of a particular service within a given unit of time (unit: %).

Interoperability Analysis (IA): It is a measure of how well a particular web service conforms to a particular set of standards (unit: %).

Response Time (RT): It is the measure of the amount of time taken by a particular web service to respond to a specific request (unit: milliseconds).
Taking into consideration the above given QoS parameters, we arrive at a set of input values to be given to the hybrid algorithm, which are tabulated below. The Table 1 shows four web services sharing the same functionality of validating an email address. These values are obtained from commercial sites and verified in [4].

Table 1: Qos Metrics for Various Email Service Providers

	C (cents)	***AV (%)***	***IA(%)***	***RT (ms)***
Web Service1	1.2	.85	.80	720
Web Service2	5	.81	.92	1100
Web Service3	7	.98	.95	900
Web Service4	7.5	.76	.90	710

Along with these values, the hybrid algorithm also requires the fuzzy limits for each parameter. It is using these limits that the membership criteria for the candidate web services can be determined. The fuzzy limits for the above set are given in Table 2.

Table 2: Fuzzy Limits

QoS Criteria	***Fuzzy Limits (L,M,U)***
Cost	(1,5,10)
Availability	(0.5,0.8,1)
Interoperability Analysis	(0.70,0.85,1)
Response Time	(500,700,1200)

Once the input values are obtained as in Table 1, the matrix in E of the hybrid algorithm is established. The values of Table 2 are then utilized to define the membership criteria for all the candidate web services. The *L, M, U* values here represent the *Least Preferred, Most Preferred and Upper Limits* of each QoS parameter.
The various steps involved in the hybrid algorithm are followed and the resultant rankings of the web services are presented in Table 3.

Table 3: Web Service Relevancy Score

.

	Rank Values
Web Service 1	7.716
Web Service 2	4.244
Web Service 3	2.592
Web Service 4	6.311

The values listed above are based on the ideal case wherein the scenario is taken as cost and response time to be the least and availability and interoperability analysis to be the highest. Various other scenarios are also possible which will be dealt in the later part of this section. The relevancy score in any scenario is fundamentally based on the user preference and the fuzzy limits listed in Table 2.

The hybrid algorithm includes the ranking methods described in both [4] and [14]. But in neither of the above mentioned methods user preference is taken into account. The input values mentioned in Table I were given separately to the algorithms in [4] and [14] and the results of all the three methods are compared in Figure 2. The analysis of the graph shows a clear deviation in the ranks obtained by the hybrid algorithm. This is due to the fact that ranking is done based on the most preferred value in the fuzzy limits and also according to the weight given by the user to a particular QoS parameter. Again Figure 2 describes only one scenario which was mentioned in the preceding paragraph.

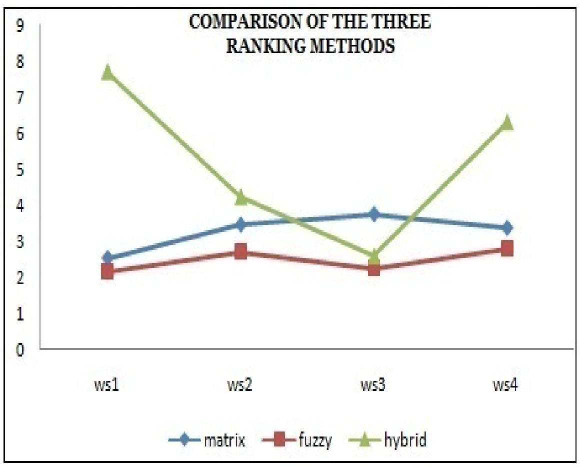

Fig 2: Comparison of the Three Ranking Methods

The actual scores obtained for each of the candidate services by the three methods are mentioned in Table 4.

Table 4: Comparison of Scores of the Three Ranking Methods

	Matrix	Fuzzy	Hybrid
Web Service 1	2.523	2.165	7.716
Web Service 2	3.461	2.706	4.244
Web Service 3	3.751	2.234	2.592
Web Service 4	3.368	2.789	6.311

Scenario 1: Ideal Case

The ideal case would be a scenario in which the ranking is based on the criteria that cost and response time should be the least and availability and interoperability should be high. Thus the preference array would be C=0, AV=1, IA=1 and RT=0.

Fig 3: Qos Ranking Based on Ideal Case

In this scenario, web service 1 has the least cost (1.2 cents), less response time (720 ms) and AV and IA very close to the most preferred fuzzy limits (.85 and .80 respectively) as can been seen from Table I. Thus it conforms to all the set conditions and hence is ranked the highest.

Scenario 2: Average case

An average case here would be wherein all parameters are given equal preference ie no specific user priority is set. The preference array in this case would be C=1, AV=1, IA=1 and RT=1.

Fig 4: Qos Ranking Based on Average Case

On analysis of Figure 4 we can determine that web service 4 has a low response time (710 ms) and a relatively high cost (7.5 cents). But these are balanced as the overall variation or differences from the most preferred fuzzy limits for these parameters are quite acceptable. Also the interoperability (.90) and availability are quite close to the fuzzy limits. Based on

IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011
ISSN (Online): 1694-0814
www.IJCSI.org

the overall satisfactory values for all the QoS attributes, the web service 4 is given the highest score for this case.

Scenario 3: Worst Case

The worst case scenario taken here is where cost and response time are given higher precedence ie these two should have a higher value and availability and interoperability need to be low. The preference array in this case would be C=1, AV=0, IA=0 and RT=1.

Fig 5: Qos Ranking Based On Worst Case

In this case, web service 3 has a high cost (7 cents), high response time (900 ms) and greater deviation of availability and interoperability (.98 and .95 giving a variation of .18 and . 10 respectively) from the most preferred fuzzy limits as is seen from Table 1 and 2. Hence, web service 3 is given the highest rank value.

The various test cases described above give a clear picture of how the hybrid algorithm functions in various situations. It can be inferred from the various graphs that the selection of the best candidate service is more precise. Various other test cases, similar to the ones described above can be simulated. Each time the user's preference plays an important role in the selection and ranking of the services.

5. Performance Evaluation

The performance of a particular algorithm can be determined using various evaluation metrics. These metrics justify the efficiency of a particular algorithm in comparison to the other techniques. The evaluation criteria should be such that it takes into account all the aspects of a particular algorithm and then predict the result.

The performance evaluation metrics used to estimate the performance of the hybrid algorithm are Normalized Discounted Cumulative Gain (nDCG) and f- measure (using precision and recall).

Normalized Discounted Cumulative Gain:

nDCG is a measure of the effectiveness of a ranking algorithm and its usefulness or gain. It gives a weight to the algorithm based on the position of a particular web service in the list and also the score of the rank.
This metric takes the actual ranking order of the web services based on the ranking algorithm as well as the scores generated by each algorithm and brings out a normalized gain value for the algorithm as a whole. The exact formula for determining the nDCG is as follows:

$$nDCG = \text{Highest rank score} + \sum_{i=2}^{n} rank\ score_i / \log_2 i$$

Here i is the order of the rank list and rank score (i) is the actual score of the i th web service.

Figure 6 shows the comparison of the nDCG values for the three ranking methods for all the test scenarios discussed above. From the figure, it is evident that the hybrid method provides a better rank when compared to the other two existing techniques.

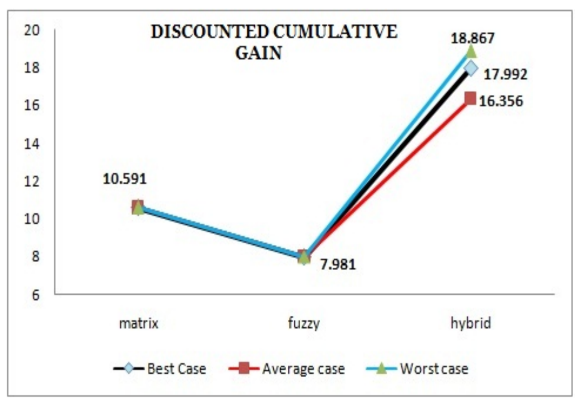

Fig 6: Evaluation Based On nDCG

f- measure:

It is a measure of the tradeoff between the precision and recall of the particular ranking algorithm. Precision is the accuracy of the ranks i.e. how well the algorithm has ranked the services according to the user preferences. Recall is the deviation between the top ranked service and the next relevant service in the list. Both these metrics are used together to arrive at the f-measure which then tests the algorithm efficiency.

The formula for f-measure is given as follows:

$$f\text{-}measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

where

$$Precision = \frac{Highest\ rank\ score}{Total\ rank\ score\ of\ all\ services}$$

And

$$Recall = \frac{Highest\ rank\ score}{Score\ of\ 2nd\ highest\ service}$$

The f-measure too is determined for all the scenarios for each of the ranking algorithms. The analysis and comparison of the algorithms based on this evaluation metric is shown in Figure 7. This metric also proves that the hybrid method offers a better solution to the ranking of the web services when compared with the earlier methodologies.

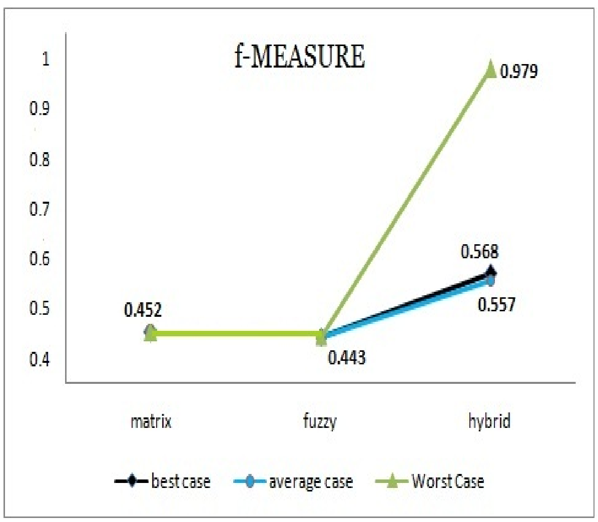

Fig 7: Evaluation Based On F-Measure

The comparison of the three algorithms using these two metrics is described in Table 5. It clearly shows the difference in the metrics for each of the ranking techniques and thus establishes the efficiency of the proposed hybrid algorithm.

Table 5: Analysis of Evaluation Metrics

	nDCG			***f-measure***		
	Case 1	Case 2	Case 3	Case 1	Case 2	Case 3
Matrix	10.59	10.59	10.59	0.45	0.45	0.45
Fuzzy	7.98	7.98	7.98	0.44	0.44	0.44
Hybrid	17.992	16.356	18.867	0.568	0.557	0.97

6. Conclusion

The concept of a hybrid ranking method which combines the benefits of two predominant approaches has been presented in this paper. This facilitates the inclusion of imprecise QoS constraints and also combines the advantage of having user preference. The usage of fuzzy sets and entropy normalization provides a ranking that is more effective than the existing ones. Also since the user preferences are taken into account a more client oriented approach is being followed. The overall effectiveness of the proposed hybrid algorithm has been justified by various performance evaluation metrics.

Acknowledgement

The authors would like to thank Amala Sherine. M, Student, Pondicherry Engineering College and Vimala. S, Student, Pondicherry Engineering College, for their valuable contribution to this paper.

References

[1] David Booth and Canyang Kevin Liu, “Web Services Description Language (WSDL) 2.0”, W3C Recommendation, 26 June 2007.

[2] L.Clement, A.Hately, C. Von Riegen and T. Rogers, “UDDI Specification and Related Documents Version 3.0.2”.

[3] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell and Carl Landwehr, “Basic Concepts and Taxonomy of Dependable and Secure Computing”, IEEE Transactions on Dependable and Secure Computing, 2004,Vol.1,No.1,pp.11-33.

[4] Eyhab Al-Masri, Qusay H. Mahmoud, “QoS-based Discovery and Ranking of Web Services”, IEEE, 2007 pp 529-534.

[5] Aviv Segev and Eran Toch, “Context Based Matching and Ranking of Web Services for Composition”, IEEE Transactions on Services Computing, Vol.2, No.3, pp 210-222, September 2009.

[6] Demian Antony D’Mello and V.S. Ananthanarayana, “Quality Driven Web Service Selection and Ranking”, Fifth International Conference on Information Technology: New Generations, April 2008,pp 1175-1176.

[7] Hong- En Lu, "Ranking Web Services Based on Ontology Semantics", Proceedings of Fourth International Conference on Machine Learning and Cybernetics, August 2005,pp 2161-2165.

[8] Kyriakos Kritikos and Dimitris Plexousakis, "Semantic QoS Metric Matching" , Proceedings on Fourth IEEE European Conference on Web Services (ECOWS'06), December 2006.

[9] Duygu Celik and Atilla Elci, "Discovery and Scoring of Semantic web Services based on Client Requirement(s) through a Semantic Search Agent", Proc. Of 30th Annual International Computer Software and Applications Conference, September 2006, pp 273-278.

[10] Raman Kumar Goyal, Vikas Gupta, Vipul Sharma and Pardeep Mittal, "Ontology Based web Retrieval", pp 1-4.

[11] Hoi Chan, Trieu Chieu and Thomas Kwok, "Autonomic Ranking and Selection of Web Services by Using Single Value Decomposition Technique", 2008 IEEE International Conference on Web Services , pp 661- 666.

[12] Dimitrios Skoutas, Dimitris Sacharidis, Alkis Simitsis, and Timos Sellis , "Ranking and Clustering Web Services using Multi-Criteria Dominance Relationships", IEEE Transactions on Service Computing,July-September 2010, pp 163-177.

[13] Dimitrios Skoutas, Dimitris Sacharidis, Alkis Simitsis, Verena Kantere and Timos Sellis, "Top-k Dominant Web Services UnderMulti-Criteria Matching",

[14] Vuong Xuan TRAN and Hidekazu TSUJI, "QoS based Ranking for Web Services: Fuzzy Approaches", IEEE ,2008 , 4th International Conference on Next Generation Web Services Practices , pp 77-82.

[15] Daniel.A. Menasce, "QoS Issues in Web Services", IEEE paper on Internet Computing, November 2002, pp 72-75.

[16] Pengcheng Xiong and Yushun Fan, "QoS-aware Web Service Selection by a Synthetic Weight", FSKD Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery - Volume 03, 2007,pp 632-637.

R.Manoharan is currently working as an Associate Professor in the Department of Computer Science and Engineering of Pondicherry Engineering College. His areas of specialization are Web Technologies and Networks.

Archana.A is currently a B.Tech final year student in the Department of Computer Science and Engineering of Pondicherry Engineering College.

Siddhika Nitin Cowlagi is currently a B.Tech final year student in the Department of Computer Science and Engineering of Pondicherry Engineering College.