Research Article
Fuzzy Hybrid Approach for Ranking and Selecting Services in Cloud-based Marketplaces

Azubuike Ezenwoke
Department of Computer Science, Landmark University, KM 4 Ipetu Omu-Aran, Omu-Aran, Kwara State, Nigeria

Abstract

Background and Objective: The popularity cloud computing has led to the proliferation of services that are commoditized and traded on cloud e-marketplaces. Besides, user’s cloud service requirements-QoS preferences and aspiration are often shrouded in vagueness and subjectivity. Therefore, cloud service selection can be overwhelming and lead to service choice overload. Existing cloud service selection approaches rarely provide mechanisms to elicit both the QoS preferences and aspirations, but rather considers either of them. This study aimed to design fuzzy-based model for service selection in e-market places that articulates both QoS preferences and aspirations. Materials and Methods: This model comprised a fuzzy Analytic Hierarchy Process (AHP) method for deriving relative priority weights of QoS attributes, a fuzzy decision-making method for obtaining user’s QoS aspiration values and a fuzzy multi-objective optimization module for evaluating the services with respect to user requirements. A simulated experiment was conduct using publicly QoS dataset and ranking accuracy produced by the proposed approach compared to existing methods was measured using Normalize Discounted Cumulative Gain (NCDG) metric. Results: The descriptive and inferential analyses of the ranking results from both versions of the proposed approach produce better accuracy results based on the NCDG metric and were in all cases closer to the benchmark metric than the other two existing methods used in this simulation. Conclusion: Results from current simulation experiment showed that the ranking accuracy of this model is not compromised by subjective QoS information from users and this approach is applicable use the subjective QoS requirements of user’s in ranking services in the cloud e-marketplaces.

Key words: Cloud computing, cloud service selection, cloud service e-marketplace, fuzzy theory, AHP


Corresponding Author: Azubuike Ezenwoke, Department of Computer Science, Landmark University, P.M.B. 1001, Omu-Aran, Kwara State, Nigeria
Tel: +2348030626181

Copyright: © 2018 Azubuike Ezenwoke. This is an open access article distributed under the terms of the creative commons attribution License, which permits unrestricted use, distribution and reproduction in any medium, provided the original author and source are credited.

Competing Interest: The author has declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.
INTRODUCTION

The popularity of cloud computing has led to the proliferation of services that are commoditized and traded via cloud e-marketplaces. An organization’s resolution to adopt a new cloud service requires decision support in navigating the vast plethora of services. Without proper articulation of requirements, cloud service selection in the face of so many choices can be overwhelming and leads to service choice overload. Decision support becomes essential because cloud service selection involves the consideration of multiple QoS attributes which are compared among a variety of services; often based on QoS requirements that are vague or subjective in nature.

Cloud service choice overload can be minimized by using low cognitive demanding decision aids for eliciting user QoS requirements, in a way that captures the vagueness that characterizes human expressions. Importantly, eliciting user’s QoS requirements is in two dimensions: QoS preferences and aspiration. Preference describes the user’s priority, while aspiration is the user’s desired values for each QoS dimension, both QoS preferences and aspirations are vital factors in the evaluation and selection of cloud services.

A number of cloud service selection approaches that consider subjectivity in user requirements exist in the literature. However, most of the approach proposed by Esposito et al., Yu and Zhang, Ma and Hu, Tajvidi et al., Sun et al., Kwon and Seo, Zia ur Rehman et al., and Wang et al., rarely provided mechanisms to elicit both the QoS preferences and aspirations, but rather considers either of them. Besides, some of these approaches, for example those provided by Esposito et al. and Zia ur Rehman et al., require that the user’s arbitrary assign importance weights to QoS attributes. In contrast to prior works, the proposed model does not accurately reflect the relative importance of the QoS attributes from user’s point of view. Hence, the need for a cloud service selection approach that effectively elicits user’s QoS requirements, in a manner that captures the inherently subjective nature of human expressions.

A Fuzzy-Oriented Cloud Service Selection method (FOCUSS) as a fuzzy-oriented decision-making model proposed in this study, for selecting services in cloud e-marketplaces. This model comprises three components: A fuzzy AHP method for deriving priority weights of QoS attributes, a fuzzy decision-making method for obtaining QoS aspiration values and a fuzzy multi-objective optimization-based model for evaluating the service alternatives with respect to user requirements.

RELATED STUDIES

In order to derive priority weights, Esposito et al. employed fuzzy set theory to capture the subjectivity in users’ QoS preferences and ranks services based on a TOPSIS-based method. Tajvidi et al. proposed a four-phase framework that handles user’s subjective QoS preferences and adopts a fuzzy AHP-based technique to rank the services, Ma and Hu recommended cloud services to users based on ternary interval numbers (TIN) and Fuzzy-AHP. Yu and Zhang proposed QSSIN_GU as a method for selecting SaaS for group users. The model uses interval numbers to combine vague QoS preferences of group members, meanwhile employing TOPSIS to rank the services. Sun et al. proposed a fuzzy framework that uses fuzzy-ontology, Fuzzy-AHP and fuzzy TOPSIS to aid service selection. Kwon and Seo present IaaS selection model based on Fuzzy-AHP. The approach by Zia ur Rehman et al. ranks services based on the similarity between user’s preferences and the values of the services properties. Wang et al. employ fuzzy synthetic decision to estimate cloud services in accordance with users’ preferences.

From the summary of related works shown in Fig. 1, approaches in Yu and Zhang, Tajvidi et al., Sun et al., and Kwon and Seo considers either the user’s QoS preferences or aspiration, but not both in the evaluation of cloud services. However, Esposito et al., Ma and Hu, Zia ur Rehman et al., and Wang et al. considered both the QoS preferences and aspirations, but did not consider vague information for both QoS requirement dimensions. In contrast, researchers proposed model captures both QoS requirements dimensions (preferences and aspiration) and also, obtains preference weights by evaluating relative importance of QoS attributes using pair wise comparison, similar to Ma and Hu, Sun et al., Kwon and Seo and Kwon and Seo, as against arbitrary assignment of weights like by Esposito et al., Yu and Zhang, Tajvidi et al. and Zia ur Rehman et al.

FUZZY-ORIENTED CLOUD SERVICE SELECTION METHOD

The use of fuzzy theory is a potent tool for representing or concepts in a vague or ambiguous way, similar to human expressions. Therefore, it suffices in capturing the vagueness that embodies QoS requirements of users. The QoS attributes can be characterized as linguistic variables, with which users express QoS preferences and aspiration. In this proposed model, namely Fuzzy-Oriented Cloud Service Selection model (FOCUSS) (Fig. 2), the preference weights were obtained using pair wise comparison method of fuzzy-AHP and the fuzziness in user’s QoS goals (or aspirations).
was elicited and processed as an organization of fuzzy goals and constraints using fuzzy decision making. The services were ranked based on fuzzy multi-objective optimization.

**Fuzzy AHP**: In contrast to crisp numerical values, it is preferable to model the user’s perception of the relative importance of multiple criteria by defining the comparison ratios as fuzzy numbers\(^{19}\). Exact (or crisp) comparison ratio \(a_{ij}\) is characterized as a fuzzy number \(\tilde{a}_{ij}\) in the Fuzzy AHP method and they were defined by the nine linguistic terms from the fuzzified Saaty’s scale\(^{19}\). The fuzzy priority vector can be obtained by applying prioritization methods. Prioritization is the process of deriving the priority values for column vector from the comparison judgment matrix. Details of the obtaining priority weights contained in the study of Buckley\(^{20}\).

**Fuzzy decision making (fuzzy DM)**: Current study modeled user’s QoS aspiration as a combination of fuzzy goals and constraints that allow cloud users to articulate QoS aspirations in a way that captures the vagueness in such judgment\(^{21}\). Some illustrations of fuzzy constraints and goal include “the cost of the service should be low” and “cost should be close to \(K\)”, where the value \(k\) is user defined. The linguistic terms “low” and “close to” are subjective descriptions of user’s judgment. Formally, fuzzy decision making is defined as:

**Definition 1**: Suppose there are \(n\) goals \((G_1\ldots G_n)\) and \(m\) constraints \((C_1\ldots C_m)\), then the resultant decision \(D\) is the interceSSION of goals and constraints, denoted as Eq. 1 and 2:\(^{21}\)

\[
D = G_1 \cap \ldots \cap G_n \cap C_1 \cap \ldots \cap C_m \\
\mu_D(x) = \min \left[ \mu_{G_1}(x), \ldots, \mu_{G_n}(x), \mu_{C_1}(x), \ldots, \mu_{C_m}(x) \right]
\]
According to Eq. 3, the maximizing decision is obtained by the value of \( x \) that has the highest membership grade in the decision fuzzy set \( D \).

\[
\mu_{\pi^*}(x) = \arg \max_{x_i} \mu_{\pi_i}(x)
\]  

(3)

**Fuzzy multi-objective optimization (fuzzy MOP):** Services were ranked using fuzzy MOP based on elicited user’s requirements. The goals of the multi-objective optimization are to find the service in the vicinity of an ideal cloud service that has the best values across all QoS dimensions and also closest to user's QoS requirements. The sources of fuzziness in the objective functions are the words phrases ‘in the vicinity of’ and ‘closest to’. Two utility functions are defined to evaluate each of the services in accordance to the QoS requirements of the user. The utility functions defined include: Simple additive weighting and exponential euclidean distance metric. The simple additive weighting function (4) determines the QoS properties of the service that has the best utility value and exponential Euclidean distance metric identifies those QoS values of services closest to users’ requirements. Both functions are transformed into fuzzy goal and constraint and solved using the symmetric model proposed by Bellman and Zadeh\(^{21}\):

\[
A_i = \sum w_i x_i
\]  

(4)

Where:

\( x_{ij} \) = \( j \)th QoS value of \( i \)th service  
\( w_{ij} \) = \( j \)th QoS weight

\[
e\text{EUD}(x, y) = \sqrt{\frac{\sum_{i=1}^{n} (x_i^2 - y_i^2)}{n}}
\]  

(5)

Where:

\( x_i \) = Value of the \( i \)th QoS properties of the cloud service  
\( y_i \) = Value of the \( i \)th user requirements, respectively.

**SERVICE SELECTION USING FOCUSS**

The practicality of the proposed model is demonstrated based on a list of 38 Customer Relationship Management (CRM) cloud services proposed by Ezenwoke et al\(^{22}\). First, the QoS information of the 38 services were fuzzified by representing three ranges of QoS values with linguistic variable and corresponding membership functions (Table 1). The range of values under each linguistic variable for QoS attributes—availability is shown in Table 2.

Apart from the QoS range, users are also allowed to express fuzzy constraints to qualify whatever linguistic term they select. The linguistic hedges and their associated membership functions are shown in Table 3.

In the next sections, the steps of the FOCUSS model are described in selecting cloud service given user requirements shown in Table 4 and 5.

**Derive QoS priority weights using fuzzy AHP:** The process for deriving weights denoted the relative importance of each QoS attributes was described, thus:

**Step 1:** Perform pairwise comparison

The user performs a pairwise comparison of the QoS attributes using the fuzzified Saaty comparison scale\(^{19}\) to fill fuzzy comparison matrix is filled as in Fig. 3.

**Step 2:** Obtain fuzzy weights and crisp equivalent

Weights representing the relative importance of each QoS attributes was obtained from the fuzzy pairwise comparison matrix following the geometric mean method\(^{20}\). The order of relative importance based on the user comparison matrix is as follows reliability > cost > availability > response time (Table 6).

**Resolve QoS aspiration using fuzzy DM:** The QoS aspiration (Table 5) was resolved as follows:

**Step 1:** Select QoS goal and constraint for each QoS attribute

The linguistic variable is selected by the user to reflect their aspiration concerning the QoS attributes. In the example presented, user’s goal for availability QoS attribute is ‘very high’, while the constraint is set to “in the vicinity of 98%”.

**Step 2:** Apply fuzzy decision making to find best QoS values

---

Table 1: Summary of QoS attributes, fuzzy sets and underlying membership function

<table>
<thead>
<tr>
<th>QoS attribute</th>
<th>Fuzzy sets</th>
<th>Membership function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>Very high, high, medium, low</td>
<td>Trapezoidal membership function</td>
</tr>
<tr>
<td>Response time</td>
<td>Low, acceptable, below average</td>
<td></td>
</tr>
<tr>
<td>Reliability</td>
<td>Very high, high, average, low</td>
<td></td>
</tr>
<tr>
<td>Cost</td>
<td>Premium, standard, moderate, cheap</td>
<td></td>
</tr>
</tbody>
</table>
Fig. 3: Fuzzy comparison matrix

<table>
<thead>
<tr>
<th>QoS Attributes</th>
<th>Availability</th>
<th>Response time</th>
<th>Reliability</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>(1, 1, 1)</td>
<td>(8, 9, 9)</td>
<td>(1/9, 1/9, 1/8)</td>
<td>(1/4, 1/3, 1/2)</td>
</tr>
<tr>
<td>Response time</td>
<td>(1/9, 1/9, 1/8)</td>
<td>(1, 1, 1)</td>
<td>(1/2, 1/1, 1/1)</td>
<td>(1, 1, 2)</td>
</tr>
<tr>
<td>Reliability</td>
<td>(8, 9, 9)</td>
<td>(1, 1, 2)</td>
<td>(1, 1, 1)</td>
<td>(2, 3, 4)</td>
</tr>
<tr>
<td>Cost</td>
<td>(2, 3, 4)</td>
<td>(1/2, 1/1, 1/1)</td>
<td>(1/4, 1/3, 1/2)</td>
<td>(1, 1, 1)</td>
</tr>
</tbody>
</table>

The QoS values (Table 7) were synthesized from users’ fuzzy estimations by finding the element with the highest membership function from the intersection set of the fuzzy sets that reflect user’s QoS aspirations. As an example, it is shown that how QoS aspiration for availability was derived.

The QoS goal (i.e., ‘very high’) for availability attribute, is represented is denoted as Eq. 6:

$$\mu_c(x) = \max \left[ \min \left( \frac{x-1}{m-1}, \frac{u-x}{u-m} \right), 0 \right]$$

According to Table 2, I = 90%, m = 95% and u = 100%, respectively correspond to lower, medium and upper values of a fuzzy set, ‘very high’. The QoS aspiration constraint, is that “the value of availability QoS should be in the vicinity of 98%” and denoted by Eq. 7:

$$\mu_c(x) = \frac{1}{1 + (x - 98)^2}$$

The value of x represents all the possible values of availability QoS contained in the service directory. The intercession of both fuzzy sets is denoted as G=∩C in Eq. 8 is solved as Eq. 3. Table 8 summarizes user’s requirements.

$$\mu_{G\cap C}(x) = \min \left[ \max \left( \min \left( \frac{x-1}{5}, \frac{u-x}{5} \right), 0 \right), \frac{1}{1 + (x - 98)^2} \right]$$

**Service ranking using fuzzy MOP**: Services are ranked following these steps:

**Step 1**: Define the fuzzy goal and constraint
- **Goal**: The utility value of the best service alternative should be “in the vicinity” of the ideal service
- **Constraint**: The QoS values of the best service alternative should be very close to the user’s aspiration
Table 8: Summary of user’s QoS requirements

<table>
<thead>
<tr>
<th>QoS attributes</th>
<th>QoS preference</th>
<th>QoS aspiration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>0.1242</td>
<td>98.49</td>
</tr>
<tr>
<td>Response Time</td>
<td>0.1237</td>
<td>489.46</td>
</tr>
<tr>
<td>Reliability</td>
<td>0.5798</td>
<td>75.43</td>
</tr>
<tr>
<td>Cost</td>
<td>0.1724</td>
<td>390.64</td>
</tr>
</tbody>
</table>

Table 9: Optimized QoS values

<table>
<thead>
<tr>
<th>QoS attributes</th>
<th>Optimized QoS values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>98.49%</td>
</tr>
<tr>
<td>Response Time</td>
<td>489.46 msc</td>
</tr>
<tr>
<td>Reliability</td>
<td>75.43%</td>
</tr>
<tr>
<td>Cost</td>
<td>390.64$/month</td>
</tr>
</tbody>
</table>

Table 10: Top ten services that match optimal requirements

<table>
<thead>
<tr>
<th>Rank</th>
<th>ID</th>
<th>Availability (%)</th>
<th>Response Time (msec)</th>
<th>Reliability (%)</th>
<th>Cost ($/Mon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S3</td>
<td>98.67</td>
<td>546.24</td>
<td>75.43</td>
<td>390.64</td>
</tr>
<tr>
<td>2</td>
<td>S17</td>
<td>99.03</td>
<td>546.24</td>
<td>75.43</td>
<td>386.15</td>
</tr>
<tr>
<td>3</td>
<td>S10</td>
<td>98.49</td>
<td>546.24</td>
<td>74.72</td>
<td>385.64</td>
</tr>
<tr>
<td>4</td>
<td>S35</td>
<td>98.62</td>
<td>489.46</td>
<td>75.72</td>
<td>360.98</td>
</tr>
<tr>
<td>5</td>
<td>S19</td>
<td>99.51</td>
<td>559.35</td>
<td>76.00</td>
<td>390.48</td>
</tr>
<tr>
<td>6</td>
<td>S4</td>
<td>97.16</td>
<td>546.24</td>
<td>72.48</td>
<td>376.66</td>
</tr>
<tr>
<td>7</td>
<td>S18</td>
<td>97.53</td>
<td>546.24</td>
<td>72.48</td>
<td>381.15</td>
</tr>
<tr>
<td>8</td>
<td>S20</td>
<td>98.01</td>
<td>559.35</td>
<td>73.04</td>
<td>380.99</td>
</tr>
<tr>
<td>9</td>
<td>S7</td>
<td>98.29</td>
<td>526.12</td>
<td>74.19</td>
<td>354.14</td>
</tr>
<tr>
<td>10</td>
<td>S32</td>
<td>98.02</td>
<td>551.35</td>
<td>75.62</td>
<td>360.46</td>
</tr>
</tbody>
</table>

Each service in the directory is evaluated using the SAW function Eq. 4. Author define \( \varphi \) as a vector of utility scores. The goal is represented as Eq. 9:

\[
\mu_i = (\varphi_i) = \frac{1}{1 + (\varphi_i - \rho)^4}
\]  

(9)

where, \( \varphi_i \) is the performance score of the alternative and \( \rho \) is the performance score of the ideal alternative. The ideal service has the best QoS values. Likewise, the eEUD (S) metrics computes the similarity between the \( j \)th alternative and the user’s aspiration with respect to QoS values, based on the mapping, eEUD(S,X), \( \theta \rightarrow [0,1] \), where is user’s QoS aspiration vector and correspond to QoS description vector of a service, 0 indicates absolute dissimilarity and 1 correspond to absolute similarity, author define \( \theta \) as a vector variable of all similarity values of user’s requirement to alternatives:

\[
\theta_i = \{ eEUD (X, s_1), eEUD (X, s_2), ..., eEUD (X, s_n) \}
\]

where, \( n \) corresponds to the number of services available in \( S \). The membership function is expressed as Eq. 10:

\[
\mu_c = (\theta_i) = \left( \frac{1}{1 + (\theta_i - 1)^2} \right)^2
\]  

(10)

The elements of the fuzzy set describe by Eq. 10 will have a degree of membership corresponding to the extent to which \( \theta_i \) is close to 1. The fuzzy decision set is denoted by:

\[
\mu_i (\varphi_i, \theta_i) = \mu_c (\varphi_i) \wedge \mu_c (\theta_i)
\]  

(11)

\[
\mu_i (\varphi_i, \theta_i) = \min \left( \mu_c (\varphi_i), \mu_c (\theta_i) \right)
\]  

(12)

The highest degree of the membership in \( \Omega \) is given by:

\[
\arg \max \left\{ \min \left( \mu_c (\varphi_i), \mu_c (\theta_i) \right) \right\}
\]

(13)

**Step 2:** Solve Fuzzy MOP

Based on this, the equivalent is a linear programming model to be solved is:

Maximize \( \min \left( \mu_c (\varphi_i), \mu_c (\theta_i) \right) \)

Subject to:

\[
\mu_i (\varphi_i) = \frac{1}{1 + (\varphi_i - \rho)^4}
\]  

(14)

\[
\mu_c (\theta_i) = \left( \frac{1}{1 + (\theta_i - 1)^2} \right)^2
\]  

(15)

The optimization model was solved with a particle swarm optimization algorithm (PSO). The results obtained were optimal QoS values that best approximates the user’s QoS requirements with respect to the spread of QoS attributes of 38 services. In this case, obtained values similar to initially synthesized values (Table 9).

**Step 3:** Rank services: The final stage is to rank the services in the service directory using optimal QoS values obtained in step 3. This is performed using flat memory technique in case of retrieval, by finding the k-nearest neighbors, for this Eq. 5 was used. Table 10 showed the 10 most suitable CRM services that match user requirements.

**RESULTS**

The simulation tests the following hypothesis \( H_0 \): There is a significant difference between the ranking performances of a method that accepts exact numeric QoS values and those
that use linguistic descriptors to approximate values for QoS requirements. Similar to the procedures outlined by He et al.,\textsuperscript{23} the QWS dataset\textsuperscript{24} was used and it comprises 2,507 services with 9 QoS attributes. For simplicity, only four QoS attributes was used, including reliability, cost, availability and response time, for this evaluation. Since the QWS dataset did not contain values for cost, uniformly distributed values were generated for the cost (interval 10-500: Corresponding to $10-$500). Five groups of datasets were obtained from QWS, resulting in datasets of 50, 100, 350, 750 and 1000 services. QoS aspiration was randomly generated following a uniform distribution from intervals with lower and upper bounds corresponding to the worst and best QoS values respectively, of each of the five groups of the dataset. For example, Table 11 shows the descriptive summary of the dataset (n = 50) and the five QoS requirements randomly generated for it (i.e., dataset, n=50), denoted as Query 1-5.

Similar to the study of Sun et al.,\textsuperscript{23} we chose TOPSIS as a baseline for comparing FOCUSSS with an existing approach, exponential weighted distance (eWD).\textsuperscript{15} The FOCUSSS and eWD are similar in their ranking principle and both methods consider user's aspiration and preferences. The eWD ranks services based on similarity values between QoS vectors of user requirements and service alternatives. We implemented two versions of FOCUSSS (FOCUSSS_Lin and FOCUSSS_Num). FOCUSSS_Lin is the original FOCUSSS method that accepts linguistic descriptions, while FOCUSSS_Num accepts numeric QoS values. Similarly, versions of eWD to process queries expressed using fuzzy linguistic descriptors were also considered. Consequently, four methods were involved in the simulation experiments (Table 11). The ranking accuracy was measured using Normalize Discounted Cumulative Gain (NCDG) metric. The relevance scores (rel) used in computing the NDCG are performance values for obtained by the TOPSIS method in response to a query.

The factors considered for the simulation whereas follows (Table 12): A number of top-k ranked services (k), the number of service alternatives (n) and the QoS requirements input type. There are six factor levels for k corresponding to (3, 5, 7, 10, 15, 20), while there are also five factor levels for n- (50, 100, 350, 750, 1000). The input types are either exact or linguistic, corresponding to two factors. Equal distribution for priority weights are assumed, such that the weight for each QoS attribute is equal to 1/q (where is the number of QoS criteria been evaluated) and q is equal to 4 (availability, response time, reliability and cost).

The protocols followed in the simulation were as follows:

- The first step in each approach was to normalize the five decision matrixes, n, using vector normalization so as to keep the values within [0, 1]
- Five QoS requirements were generated for which each method generated a ranking of cloud services from the decision matrix. The queries were also normalized using vector normalization method
- For each combination, the trials were performed five times using the five QoS requirements described in as queries, after which the average for each combination case was taken. In all, 600 solutions were generated (120 data items per QoS query)
- The average values from all metrics for all methods, resulting in 120 data point, were analyzed with descriptive and inferential statistics
- The median ranking accuracy measured by NCDG is shown in Table 13. Both versions of FOCUSSS produced better ranking results and are closer to TOPSIS than others. Inferential analysis using the Kruskal-Wallis test (Table 13) showed the significant statistical difference in the accuracy performance of four methods compared $\chi^2 (3, N = 120) = 27.251, p<0.05$. Furthermore, higher mean rank suggests better accuracy, FOCUSSS_Lin had the highest mean rank (M = 77.12), closely followed by FOCUSSS_Num (M = 77.02). The method with the lowest mean rank is eWD_Num (M = 43.38). However, there is no significant difference in the input type whether numeric or linguistic (p = 0.925)

| Table 11: Minimum QoS values, maximum QoS values and five test queries for dataset (n = 50) |
|---|---|---|---|---|
| Availability | Response time | Reliability | Cost |
| Min | 18.00 | 49.43 | 53.00 | 111.63 |
| Max | 100.00 | 3321.40 | 83.00 | 496.01 |
| Query 1 | 24.66 | 492.69 | 62.10 | 197.92 |
| Query 2 | 90.79 | 1608.38 | 59.64 | 341.70 |
| Query 3 | 46.99 | 377.46 | 61.34 | 160.98 |
| Query 4 | 96.74 | 1279.35 | 71.90 | 466.13 |
| Query 5 | 60.17 | 346.89 | 74.89 | 152.97 |

| Table 12: Simulation variables, levels, and metrics |
|---|---|---|---|---|
| #Service (n) | 50, 100, 350, 750, 1000 |
| Top-k (k) | 3, 5, 7, 10, 15, 20 |
| Methods to be compared (m) | FOCUSSS_Lin, FOCUSSS_Num, eWD_Lin, eWD_Num |
| QoS Attributes (q) | 4 |
| Queries per run (t) | 5 |

| Table 13: Median Ranking Accuracy by NCDG |
|---|---|
| Methods | Median accuracy |
| eWD_Num | 0.94141 |
| eWD_Lin | 0.93996 |
| FOCUSSS_Num | 0.98218 |
| FOCUSSS_Lin | 0.98173 |
DISCUSSION

The results from both the descriptive and inferential analysis presented in the previous section confirms that the two versions of the FOCUSS methods (FOCUSS_Num and FOCUSS_Lin) produce better accuracy results based on the NCDG metric. Furthermore, the ranking of cloud services by the two version of the FOCUSS methods were in all cases closer to the benchmark metric (TOPSIS) than the other two methods used in the simulation experiment. The significantly higher mean rank of the FOCUSS_Lin methods indicates that FOCUSS_Lin produces more accurate rankings than other methods. In addition, expressing QoS requirements using linguistic terms did not compromise the accuracy of the ranking method, as there is no significant difference in the rankings produced by both QoS input types (Table 14), hence, we reject the null hypothesis, $H_0$.

As demonstrated by Esposito et al., Yu and Zhang, Tajvidi et al., Sun et al. and Kwon and Seo, expressing QoS requirements using linguistic descriptors that are more akin to human expressions still produce comparably accurate rankings. Therefore, the user experience in service selection from the e-marketplace is enhanced over comparison judgment using crisp numerical values. The use of only crisp values lacks the flexibility to effective capture vagueness in human judgment and sometimes leads to unsatisfactory decisions.

CONCLUSION

Cloud service selection from a plethora of options can be overwhelming and places a huge cognitive demand on the user. This paper contributes a fuzzy-based cloud service selection model that elicits vague QoS preferences and QoS aspirations using linguistic descriptors and ranks the services accordingly. Experiment results confirm that the ranking accuracy of the model is not compromised by user’s subjective QoS information. Therefore, the proposed model is can be employed to rank and select services in cloud e-marketplaces.

SIGNIFICANCE STATEMENT

This paper contributes a fuzzy-based cloud service selection model that aids users to articulate their QoS requirements in a manner that caters for the inherent subjectivity in human expressions. This study will be helpful and useful in the improving the quality of user experience in cloud service e-marketplaces by supporting vague user expressions in requirements specification.

REFERENCES


