Flexible user-centric service selection algorithm for Internet of Things services

[1]TRUPTIMAYEE SAHOO,
Gandhi Institute of Excellent Technocrats, Bhubaneswar, India
[2]SUBHAKANTA BEHERA,
Capital Engineering College, Bhubaneswar, Odisha, India

Abstract

In Internet of Things (IoT), the similar functional services are evolving in different quality of services (QoS) due to the widespread deployment of spatially distributed things on dynamic networks through the web. Therefore, the user involvement in service selection systems becomes a vital role to enhance the system performance by taking into account of subjective factors. This paper proposes a flexible QoS-based service selection algorithm (FQSA) which mainly targets to the users to be able to give their subjective preferences in an easy and friendly manner. The FQSA algorithm selects the services based on subjective information provided by the service consumers and objective information supported by the service providers. It adopts artificial neural network backpropagation algorithm to find the objective factors and applies similarity aggregation method to evaluate the creditability of the user subjective factors which have already been evaluated by ontological reasoning with the help of proposed QoS ontology. The comparative study and experimental results show that our FQSA algorithm is superior to other service selection approaches.

Keywords IoT, QoS-based service selection, artificial neural network backpropagation, objective factors, similarity aggregation method, subjective factors; ontological reasoning, QoS ontology

1 Introduction

Nowadays, IoT provides a shift in service provisioning system moving from traditional web era to sophisticated things era, which is connected to the Web. For communication between the web and real things, the object abstraction layer on an IP network accesses the services of different devices with a common language [1]. If devices cannot offer the discoverable services on IP network, the interface sub-layer, communication sub-layer and wrapping function, apply the web service interface to expose the methods available from the devices, to extract the logic behind these methods and to translate different communication languages of external things to standard web service language [1].

In IoT environs, the web connected by IoT things is highly distributed network, comprising of an enormous number of things acting as providers or consumers who share the information. Compared with conventional web service selection, IoT based service selection system is more challenging to handle the highly dynamic services structured on heterogeneity of devices and resources in different qualities. This challenging could push towards the adoption of standard structure for service functional and non-functional information to be able to understand, interpret and select the services and integrate their corresponding functionalities to satisfy highly automatic IoT service demands. This focus is on structuring and specification the exact and precise relevant QoS information which can be offered by each service running on IoT environs. In addition, it is not efficient to solely

[redacted]

depend on the user QoS definition because users are not QoS experts. Therefore, the standard QoS specification based on ontological engineering is demanding to understand the QoS descriptions provided by service providers and service consumers.

Many research works [2–5] have been developed to address QoS aspects for service schemes in different QoS specification with different purposes. While some service selection approaches emphasize the objective QoS information which is offered by the service provider, some approaches consider subjective QoS information which is provided by the service consumers who actually use the service. If the user preferences and their feedbacks are retained as historical users' recommendation on the selected services returned [6], the search quality can be improved because users' subjective feelings and feedbacks can not only help to easily find the most user wanted services, but also upgrade the reliability and performance of searching process, to have group consensus agreement with historical users' recommendation on the previous searched results.

Maximilien et al. [7] and Wang et al. [8] considered only subjective information in their service selection systems. As a result, solely considering the subjective information degrades the system confidentiality. The reason is that the standard of user's feedback is not universal. Therefore, Wang et al. [6] took into consideration of the subjective factors together with objective factors but their QoS assessment is too restrictive for the users under their pre-defined five QoS criteria and fuzzy linguistic terms forQoS values. Besides, they evaluated the trustability of the subjective factors only depending on the user who currently demands the service without consideration of the group consensus agreement among the users who previously demanded that service. Therefore, their trustability evaluation is uncertain. As a result, it can negatively impact on the system performance because of the insufficient and incomplete assessment decided by only one user.

To address these deficiencies, we propose FQSA algorithm which uses both subjective and objective factors for QoS information. For the subjective factors, the users are allowed to demand any QoS requirements in their random ways but within the boundary of QoS criteria offered by the service providers. To understand and extract the QoS meanings and concepts from the user random inputs, the standard QoS specification that is suitable for IoT services is structured as QoS ontology. Then, by our algorithm, the subjective meanings of their inputs are extracted and concluded using QoS ontology, WordNet and ontological reasoning. Finally, depending on the subjective QoS scores from the user random QoS values which are evaluated by similarity aggregation method (SAM) and objective QoS scores which are calculated by artificial neural network backpropagation model (ANN-BP), the most relevant services set is selected and replied to the users.

2 Methodology of QoS based service selection

In underlying the service selection approach, there are three basic steps: matching, ranking and selecting [3]. Matching finds the candidate services which are similar in functional service information and ranking finds the service sets which are similar in non-functional QoS information and rank them according to their quality scores. Selecting chooses the most relevant services set according to the user demands. The service selection step which is applied in this paper can be described as follows.

$$S_{\text{seletedservice}} = f_{\text{select}}(S,V)$$

where

$$S,V = (S_{i(\text{FANDNFNSimilarService})}, V(S_i))$$

$$V(S_i) = \sum_{j=1}^{m} Q_j; \quad i = 1,2,\ldots,n, \; j = 1,2,\ldots,m$$

$$S_{\text{selectedService}} = \{S_i \,|\, V(S_i) \geqslant \phi \quad \text{or} \quad V(S_i) \geqslant f_\delta\}$$

$S_{i\ (\text{FANDNFNSimilarService})}$ is the set of services which is similar in functional and non-functional service descriptions between service advertisements and service request. Then, the services whose average quality values (denoted as $V(S_i)$ obtained from summing the individual quality values Q_j) are higher or equal to the user/system defined variable/function or hypothetic formula are selected and replied to the users.

3 System model

Our model integrates the idea of Wang et al (2007) [6], Lin et al (2008) [2] and our novel idea. In our model, there are one service repository, four main components and two participants called service provider who advertises the services and service consumer who demands the services and involves in the service selection process.

The workflow of our proposed model is initiated when a user request is received. Firstly, service matching process

[redacted]

matches the functional service descriptions of the available services obtained from UDDI repository. Then, the ranking process ranks the candidate services depending on their non-functional quality scores (QoS) calculated by FQSA algorithm from subjective and objective factors. Finally, the selection process selects the services which match with the user expected QoS requirements. The system architecture is shown in Fig. 1 and the main functionalities of each component are described as follows.

UDDI repository: stores OWL-S documents which extend the QoS information according to W3C specification to know which the catalogue of services associates to each IoT object through the Web [1] and keeps the histories of user feedbacks.

QoS acceptor: accepts user QoS requests in any user random format in two fragments: QoS criteria and QoS value.

QoS understander: transforms the user random form to the system understandable form with the aid of WordNet to analyze the language thesaurus, QoS Ontology to understand QoS characteristics and ontological reasoning to evaluate the consistency and reasonableness among QoS criteria chosen by the users.

User-centric subjective QoS evaluator: evaluates the accuracy and certainty of user QoS assessment by applying SAM method to find the consensus creditability among the user group which ever demanded on this service type and calculates the subjective quality score.

ANN-based objective QoS evaluator: evaluates the objective quality factors by adopting enhanced ANN algorithm and calculates the objective quality score.

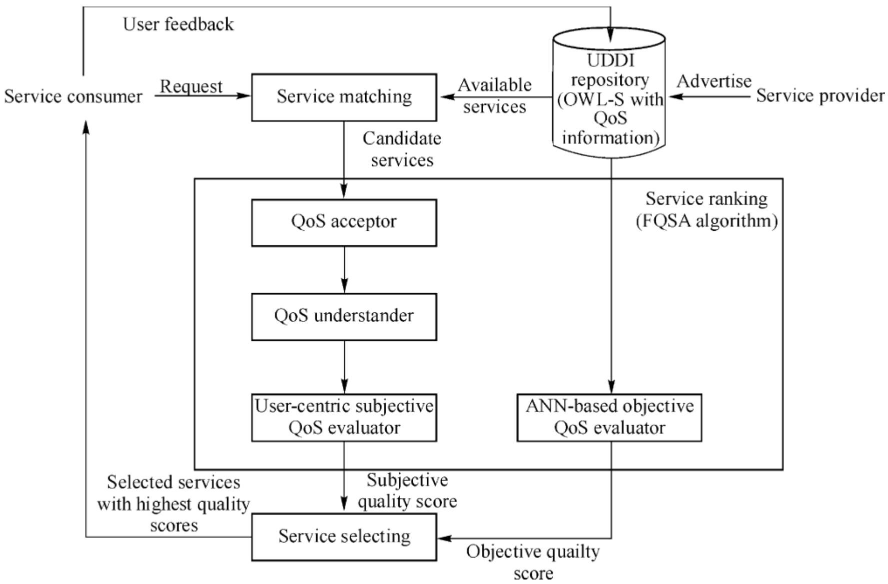

Fig. 1 Service slection model

4 FQSA selection algorithm for IoT services

Finding subjective quality scores from user preferences

Procuring QoS-assessment from the users

In user subjective QoS assessment of service selection approaches [6–8], although they access the subjective factors from the users, they lack of considering on creating the flexible and user-friendly assessment form for any kind of users. This deficiency makes the users be tedious to involve subjective QoS assessment because almost most of the users are not QoS experts and also cannot easily understand the complicated terms defined by the experts who use different QoS languages and models to describe their QoS specification depending on their domains and applications. Therefore, when procuring the user QoS assessment, the users are allowed to request any number of QoS criteria for a service which each QoS criteria has duplet definition as $Q_j < N, E >,\ j = 1, 2, ..., m$ where N is the name of QoS criteria and E is the quality range they expected for a service S_i. Due to flexible QoS assessment proposed by our FQAS algorithm, in the field of *N and E* inputs, the user can enter their understood and familiar words for their expected conditions. e.g., Q_1<Performance,

[redacted]

Good>, Q_2 <responsetime, 0.5 *ms*>, etc.

Understanding and transforming the user input

To understand the user defined inputs and transform them to machine understandable form, WordNet is used for language point of view and ontological reasoning is used for deducing the QoS concepts with the help of proposed QoS ontology for IoT services.

The function of ontological reasoning is to check the class consistencies, implied relationships and asserted inter-ontology [9]. The ontological reasoning supports RDFS and OWL reasoning through the use of QoS ontology structure to infer QoS characteristics such as relations between classes (e.g. transitive), cardinalities (e.g.minimum one), and inferred rules and so on.

For ontological reasoning, we develop well-structured QoS ontology for different domains such as network, thing, activity and other related domains to IoT environs by mainly dividing into two domain levels: generic domain and specific domain. The former one is structured for description of basic QoS service information. The latter one is structured for specific QoS information related to each different domain. Then, the cross-domain QoS information is formed and defined as the basis for domain independent QoS specification to be able to link the generic and specific domain QoS specification. The core QoS parameters which are commonly used by most of service selection approaches are defined and structured according to their related classes, sub-classes, properties and etc. In this paper, we mention only the overall architecture of QoS ontology to flit how IoT related QoS information are structured and how this QoS ontology supports to the ontological reasoning.

The process of understanding and transforming the user QoS random input is described in examples as follows. Let the user input be output (or) result (or) work done, then, WordNet will infer these inputs as throughput which is related to QoS terms. In fact, the meaning behind of those words may be similar, but the descriptive words may be different. After deducing the user random terms to reasonable QoS terms, the actual meaning what the user really expected is extracted with the help of following rules [10] obtained from the ontological concepts of QoS ontology.

1) SubClassOf (?A rdfs:subClassOf? B), (?B rdfs: subClassOf? C) → (?A rdfs:subClassOf? C)

2) TransitiveProperty (?P rdfs:type owl: Transitive Property), (?A ?P ?B), (?B ?P ?C) → (?A ?P ?C)

3) Disjontness (?C owl:disjointWith? D)(?X rdf:type? C), (?Y rdf:type ?D) → (?X owl:differentFrom ?Y)

Then, the ontological reasoning extracts the actual QoS criteria performance from the WordNet reasonable QoS terms throughput as shown in following examples.

1) Output, result, work done, final score → [semantic meaning] throughput [QoS attribute]→ [QoS attribute] Performance [QoS criteria]

2) Processing time, execution time, total time taken, waiting time, reply time, speed → [semantic meaning] response time [QoS attribute] → [QoS attribute] Performance [QoS criteria]

After deducing all QoS criteria from N of user duplet QoS assessment, we need to evaluate their all QoS assessment creditability in the next section.

Finding creditability consensus agreement

SAM is applied for consensus opinion class [2] to resolve the conflicts and uncertainties emerged from different opinions came from a specific problem by comparing with average similarities [10].

To find the creditability consensus agreement value, there are five steps and in each step, the consensus similarity values are collected and then consensus agreement is calculated.

1) For one specific QoS criteria assessment $Q_j(N)$ chosen by the current service consumer denoted as U_k, this step finds the QoS assessment similarity between two users (U_k and U_l)

$$I_{kl} = \frac{\int (\min\{\tilde{\mu}(U_k), \tilde{\mu}(U_l)\})dx}{\int (\max\{\tilde{\mu}(U_k), \tilde{\mu}(U_l)\})dx} \quad (1)$$

2) This step finds the similarity agreement matrix for each similarity value between each pair of total users denoted as p of similar service request group.

$$\boldsymbol{A} = \begin{bmatrix} 1 & I_{12} & \dots & I_{1l} & \dots & I_{1p} \\ I_{21} & 1 & \dots & \dots & \dots & \dots \\ \dots & \dots & 1 & \dots & \dots & \dots \\ I_{k1} & \dots & \dots & I_{kl} & \dots & I_{kp} \\ \dots & \dots & \dots & \dots & 1 & \dots \\ I_{p1} & I_{p2} & \dots & I_{pl} & \dots & 1 \end{bmatrix}_{p\times p}$$

3) This step finds the average consensus similarity value (C) for each single user U_k denoted as $C(U_k)$

[redacted]

according to following Eq. (2).

$$C(U_k) = \frac{1}{p-1}\sum_{\substack{l=1 \\ k\neq l}}^{p} I_{kl} \tag{2}$$

4) This step calculates the relative consensus agreement (*R*) for each user U_k according to Eq. (3).

$$R(U_k) = \frac{C(U_k)}{\sum_{l=1}^{p} C(U_l)}; \quad l \neq k \tag{3}$$

5) After getting *R* value for each specific QoS criteria, this step calculates the creditability consensus agreement (*C*) for all QoS criteria (Q_j) chosen by the user U_k for a service S_i he/she requested.

$$C(S_i) = \frac{\sum_{j=1}^{m} R(Q_j)}{m} \tag{4}$$

As the next step, we standardize all QoS values $(Q_j(E))$ by normalizing all forms of all QoS values and calculate the final subjective quality score.

4.1.4 Standardizing subjective quality score

The subjective QoS score is derived from Q_{value}, which is defined by the providers or users as numerical literal, string literal or etc. In fact, the QoS values are defined as float numbers in the range of 0–1 in the repository. Since *E* may be different ranges under different types, to have standard form of all input values such as numeric, string, float, decimals, we use *Z*-score normalization to derive the standard float number in the range of 0–1 from all possible value types and ranges according to Eq. (5).

$$Q_j(E) = \frac{x(j) - \mu}{\sigma} \tag{5}$$

where $x(j)$ is the attribute value of a QoS criteria Q_j, μ is the average of all attribute values of Q_j, σ is the standard deviation of the attribute values and $Q_j(E)$ is the normalized value of a QoS criteria Q_j. Then, we sum all $Q_j(E)$ alues to calculate the total QoS value denoted as $V(S_i)$ for a service S_i.

$$V(S_i) = \sum_{j=1}^{m} Q_j(E) \tag{6}$$

Finally, subjective quality score (denoted as S_s) for a service S_i is calculated from *C* value and total QoS value *V* as shown in Eq. (7).

$$S_s = C(S_i) * V(S_i) \tag{7}$$

Finding objective quality scores using ANN model

To find objective quality score, we adopt enhanced ANN-BP algorithm [11–12] which solves the drawbacks of traditional ANN-BP such as local minima and slow convergence speed by modifying the derivation of sigmoidal function to escape from local minima and also speeding up the convergence rate. The detailed proof of the theorem can be found in Ref. [12].

For objective quality scores, their ANN-BP is applied with our input parameters such as service and context information of IoT services. The mathematical model can be expressed as follows.

$$\begin{aligned} x_i(t+1) = f[&x_i(t), x_i(t-1), \ldots, x_i(t-n+1), AV_1(S_i)), \\ &AV_2(S_i)), \ldots, AV_n(S_i)), Q_1(S_i), \\ &Q_2(S_i), \ldots, Q_j(S_i), C_1(S_i), C_2(S_i), \ldots, C_n(S_i)] \end{aligned} \tag{8}$$

where $x_i(t)$ is the output of neural network relating to the *i*th element of candidate services at *t* time. $AV_n(S_i)$ is the value of service attributes. $Q_j(S_i)$ is the value of QoS of each service S_i. $C_n(S_i)$ is the context factors related to the current environment of service request and $x_i(t+1)$ is the output of the neural network. The *f*(*) is the function of neural network which results the objective quality score denoted as O_s.

5 Service selection process

Generally, service selection system selects the services whose qualities are higher or equal to the quality standard determined by system experts or specific conditions. In our FQSA, it is denoted as R_s and calculated according to Eq. (9).

$$R_s = w_s S_s + w_o O_s \tag{9}$$

where w_s and w_o are the tunable weight values for subjective and objective scores which can be tuned depending on the number of user (p) who ever used that service, threshold value (*α*) to determine the enough level of user number to bias the subjective score and variable (*β*) to control the bias level, in that case their initialized values are same and summation is 1. If ($p(S_i) \geqslant \alpha$) then, $w_s = w_s + \beta$ and $w_o = w_o - \beta$, else $w_s = w_s - \beta$ and $w_o = w_o + \beta$. If α is large, the system will analyze more the user subjective factors than objective factors. If α is small, the system selection will analyze the system objective factors more than user subjective quality factors.

Finally, the system replies the selected services with their corresponding quality values as $S_i < S,V >$ to the users whose quality score $V(S_i)$ is greater or equal to R_s.

6 Comparative study and experiment evaluation

Comparative study

We compare our FQSA algorithm to other related works: genetic algorithm and fuzzy logic based service selection algorithm (GAFLSS) [6], agent proxy on user preference approach (APUP) [7] and fuzzy linear programming approach (FLP) [8] as shown in Table 1. We denote yes for the evaluation metric they supported and no for not supporting or weaker supporting. The briefly definition of each evaluation metric and their comparison results are described as follows.

QoS aggregation: aggregate the individual score to gain a final score of the service [13]. Our FQSA and FLP can strongly support this metric due to our well structured QoS categories while the support of APUP is unclear. GAFLSS cannot support this evaluation metric due to lack of QoS categorization and summarization.

QoS reasoning: evaluate the reasonableness of QoS criteria requested for a specific service type. FQSA use ontological and OWL reasoning to reason the consistency and reasonableness among QoS criteria requested by the current user and previous users while other three approaches lack supports for this reasoning process.

QoS scalability: allow QoS properties to extend without affecting the pre-defined structures and also system processes and performances. Our FQSA, we can strongly support QoS scalability by allowing to derive instances of QoS classes from QoS ontology and inserting their new defined metrics and values to these instances. But, the other three approaches did not explore the scalable purpose.

Personalized confidentiality: evaluate the confidentiality of user involvement and their assessment. Our FQSA supports strong confidentiality by analyzing the individual or group consensus on the user QoS criteria choices while GAFLSS evaluates only individual trustability. Meanwhile, APUP and FLP barely support user confidentiality. However, we conclude that every approach can support personalized confidentiality in their achievements.

User friendliness: analysis the flexibility of user involvement and satisfaction level on system defined framework. Apparently, only FQSA can support the most friendliness user involvement form among others.

Table 1 Comparison of service selection approach on evaluation metrics

Evaluation metrics	FQSA	GAFLSS	APUP	FLP
QoS aggregation	Yes	No	No	Yes
QoS reasoning	Yes	No	No	No
QoS scalability	Yes	No	No	No
Personalized confidentiality	Yes	Yes	Yes	Yes
User friendliness	Yes	No	No	No

Experiment evaluation

For implementation, 25 service classes which each has 100 service candidates are created with random QoS datasets, arbitrary values and experts defined reasoning rules.

Fig. 2 compares the reliability of trust mechanism based service selection algorithm (TMSS) [14], GAFLSS and FQSA by calculating the harmonic mean of the recall and precision on each system usage frequency. This result is evaluated on 30 experimenters and 10 times selection from 30 candidate services in each experiment. Results show that our FQSA is better than other service selection methods.

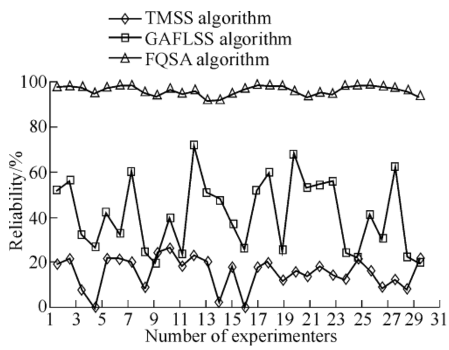

Fig. 2 FQSA reliability on different experimenters

Fig. 3 describes the satisfaction rates of the users on FQSA by using five-point rating scales on their satisfaction levels of QoS-assessment form and selected services. According to Fig. 3, approximately 32.4% of users highly satisfy FQSA and nearly 57.6% agree that it is flexible and easy to use. Therefore, most of the people satisfy our FQSA flexible user involvement and selective performance.

[redacted]

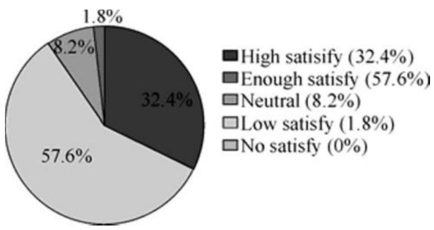

Fig. 3 User satisfaction level on system service selection

7 Conclusions

In this paper, we develop FQSA algorithm under the following contributions.

1) QoS Ontology under different context structures of IoT services.

2) Flexible user-friendly assessment form for the users.

3) Consensus creditability evaluation on subjective user preferences. Furthermore, we adopt the enhanced ANN-BP algorithm to increase selection rate to be suitable for real-time service selection abilities. As a conclusion, according to the comparison and experimental results, our FQSA algorithm is apparently much better worth in selective performance, user satisfaction level and friendliness rates than other proficient service selectionapproaches.

References

1. Atzori L, Iera A, Morabito G. The Internet of Things: a survey. Computer Networks, 2010
2. Lin W L, Lo C C, Chao K M, et al. Consumer-centric QoS-aware selection of web services. Journal of Computer and System Sciences, 2008, 74: 211–231
3. Tran V X, Tsuji H, Masuda R. A new QoS ontology and its QoS-based ranking algorithm for Web services. Simulation Modeling Practice and Theory, 2009, 17: 1378–1398
4. Cai H, Hu X, Lu Q, et al. A novel ANN-based service selection model for ubiquitous computing environments. Expert Systems with Applications, 2009: 2200–2212
5. Zhao L, Ren Y, Li M, et al. Flexible service selection with user-specific QoS support in service-oriented architecture. Journal of Network and Computer Applications, 2012, 35: 962–973
6. Wang H C, Lee C S, Ho T H. Combining subjective and objective QoS factors for personalized web service selection. Expert Systems with Applications, 2007, 32: 571–584
7. Maximilien E M, Singh M P. Conceptual model of web service reputation. ACM SIGMOD Record, 2002, 31(4): 36–41
8. Wang P, Chao K M, Lo C C, et al. A fuzzy model for selection of QoS-aware Web services. Proceedings of IEEE International Conference on e-Business Engineering (ICEBE'06), 2006: 585–593
9. Gu T, Pung H K, Zhang D Q. A service-oriented middleware for building context-aware services. Journal of Network and Computer Applications, 2005, 28: 1–18
10. Chiclana F, Herrera F, Herrera-Viedma E. Integrating multiplicative preference relations in a multipurpose decision-making model based on fuzzy preference relations. Fuzzy Sets Systems, 2001, 122: 277–291
11. Ng S C, Leung S H, Luk A. Fast and global convergent weight evolution algorithm based on modified back-propagation. Proceedings of IEEE International Conference on Neural Networks Proceedings, 1995, 6
12. Hecht-Nielse R. Theory of back propagation neural network. Proceedings of IJCNN, 1989, 1: 593–603
13. Sathya M, Swarnamugi M, Dhavachelvan P, et al. Evaluation of QoS based Web-service selection techniques for service composition. International Journal of Software Engineering (IJSE)
14. Yuliang S H, Guang-an H, Liang Z H, et al. A method to select the optimum web services. Proceedings of 8th Asia-Pacific Web Conference on Frontiers of WWW Research and Development. Harbin, China, 2006: 868–873