Intelligent Web Services Selection based on AHP and Wiki

Chen Wu and Elizabeth Chang

Digital Ecosystems and Business Intelligence Institute
Curtin University of Technology, Australia
{Chen.Wu, Elizabeth.Chang}@cbs.curtin.edu.au

Abstract

Web Service selection is an essential element in Service-Oriented Computing. How to wisely select appropriate Web services for the benefits of service consumers is a key issue in service discovery. In this paper, we approach QoS-based service selection using a decision making model – the Analytic Hierarchy Process (AHP). In our solution, both subjective and objective criteria are supported by the AHP engine in a context-specific manner. We also provide a flexible Wiki platform to collaboratively form the initial QoS model within a service community. The software prototype is evaluated against the system scalability.

1. Introduction

Existing service matchmaking mechanisms (such as UDDI registries, StrikerIron\(^1\), etc.) have provided a clear view of service relevance information. The ranking mechanism does bring the order based on the relevance of service descriptions. However, the static relevance does not necessarily represent run-time appropriateness. For example, a functionally relevant Web service might not be appropriate for some users due to its slow speed or prohibitive cost [1]. Likewise, a Web service with a relatively lower relevance ranking can well suit some service consumers’ requirements for data privacy. Therefore, service communities need to provide service selection enabling techniques to help users determine the ‘degree of appropriateness’ for each service candidate from a user’s perspective. Service selection mechanism represents significant value added by service brokers and syndicators.

In this paper, we approach QoS (Quality of Service) -based service selection using a decision making model – the Analytic Hierarchy Process (AHP). In our solution, both subjective and objective criteria are supported by the AHP engine in a context-specific manner. We also provide a flexible Wiki platform to collaboratively form the initial QoS model within a service community.

2. Conceptual Model

For generality, we will use the QoS meta-model defined in [2] as the basis for the QoS and AHP modelling in following sections. In order to apply the AHP model to the QoS meta-model, we have extended the original QoS meta-model [2] in two ways. First, we introduce the concept of Sub-QA into the QoS meta-model. The motivation is based on the fact that one QA may be dependent on several key factors at the same time. In other words, the performance of these key factors collectively determines the ultimate score of this QA. These key factors are defined as Sub-QA (SQA), and their corresponding QA is termed as Parent QA (PQA). It is noted that an SQA automatically becomes a PQA if its measurement hinges on several other SQAs. This way, a tree-like structure of QoS meta-model is formed with unlimited levels of PQA and SQA. The second extension we have made is to place a constraint on the Assessment Criteria: a QA is allowed to have associated AC only if it is not a PQA of any kind. Simply put, only the ‘leaf’ node in the QoS meta-model tree will have associated AC. This assertion distinguishes the act of measuring QoS and the act of structuring QoS models. Such a decoupling decomposes the complex problem into focused areas. Thus, domain experts can work solely on the QoS modelling, and stakeholders only need to assess AC for the leaf nodes without involving the whole QoS model. The leaf node SQA is termed as LQA in this paper.

The AHP is one of the most widely used multiple criteria decision-making models [3]. The AHP was originally proposed in Saaty [4], allowing decision makers to model a complex problem in a hierarchical structure consisting of Goal, Factors, Sub-factors, and

\(^1\)http://www.strikeiron.com/
Alternatives. Since its foundation, the AHP has been successfully applied in a wide range of applications [5] involving multi-criteria decision making. The AHP model is illustrated in Figure 1, where the AHP hierarchy sits in the middle representing the complex problem, i.e. service selection. From the AHP perspective (right), it has four levels of constructs – Goal, Factors, Sub-Factors, and Alternatives – corresponding to four concepts in the QoS meta-model (left) – Context, Quality Aspects, Quality Criteria, and Web Services respectively.

3. The AHP Approach

The AHP approach has three core processes, each of which includes several steps. During the system design time, two processes (two grey arrows) generate the AHP indices, which keep the calculated priority for each web service under each AC. It also stores the original pair-wise comparison matrices. AHP indices are used during the system run time (the white arrow) to derive the final QoS score based on consumers’ preferences.

The first important task in applying the AHP is to construct the hierarchy, which includes factors and sub-factors that characterize the QoS-based decision making problem. Saaty and Vargas [6] suggest that during the hierarchy construction, one must “include enough relevant detail to represent the problem as thoroughly as possible”. Such a user-centered thoroughness requires a collaborative process by which all stakeholders of the decision problem can easily participate in constructing the complete AHP hierarchy. This collaborative process coincides with the concept of “community” defined in [7], where “a community defines the quality tree for a given class of Web services”. We have thus built a Web Wiki platform in order to foster such a collaborative process within the ‘community’. It enables service brokers and consumers to collaboratively construct the problem hierarchy in a very efficient manner. The second stage of the AHP approach the pair-wise comparison and the eigenvector method to estimate the overall priorities. This results in the AHP indices that keep all relative priorities for Web services under each Assessment Criteria for different reviewers.

The next step is to estimate the ranking of \( WS = \{ws_1, ws_2, ..., ws_n\}, n \geq 2 \) based on these ratio values in matrix. In AHP, this is accomplished by deriving a vector of priorities from the raw pair-wise comparison ratios. Saaty [4] proposed an eigenvector method to derive the priorities, where each component in the principal right eigenvector is normalised to between 0 and 1 to represent the relative priority of alternatives. Choo and Wedley [8] has discussed 18 approaches that attempted to achieve this aim. However, [9] has shown that the eigenvector approach is “a theoretically and practically proven” method for estimating the priorities. Thus, the eigenvector approach formally deals with human errors inherent in any decision making processes. The estimate of the overall relative priority for an alternative Web service \( p_{ws_i} \) is a component in a vector \( P \). \( P \) is derived by solving the eigenvalue problem \( SP = \lambda_{\text{max}} P \), where \( \lambda_{\text{max}} \) is the largest eigenvalue of matrix \( S \), \( S \) is the aggregated comparison matrix, and \( P \) is the principal (dominant) right eigenvector corresponding to the eigenvalue \( \lambda_{\text{max}} \). In practice, this is accomplished by using the Power Iteration algorithm [10]:

\[
\lim_{k \to \infty} S^k e = \lim_{k \to \infty} \left[ \frac{1}{\|S^k\|} S^k \right] e
\]

The next step is to determine the relative preference for each Quality Aspect (QA). The rationale behind is that the QA should not always be considered equally important during the service selection. Instead, each QA has distinct importance contributing to the ultimate QoS value under certain contexts. Therefore, they should be assessed in a way that their respective significances are measured and integrated. Moreover, these relative preferences are unique from user to user. Thus, user preferences play significant roles in determining the most appropriate services. It reflects the dynamics of the service selection during the system run-time. The raw priorities of each service – the AHP indices – obtained in Sect. 4.3 have to be ‘normalised’ with user preferences in order to achieve the user-centred service selection.

Given the hierarchical structure of QoS model, the QA comparison starts from the top layer of the AHP hierarchy. It walks through each level of the QoS
model by performing pairwise comparison in that level. To match all these priorities with their corresponding LQA preferences, we can obtain the final QoS result matrix by left multiplying $P_{all}$ with $FLP$. Thus $FLP \times P_{all}$ yields the final QoS vector:

$$Q = \left[ \sum_{i=1}^{m} p_{w_{i}}^{l} \times f_{l}, \sum_{i=1}^{m} p_{w_{i}}^{l} \times f_{l}, \ldots, \sum_{i=1}^{m} p_{w_{i}}^{l} \times f_{l} \right]$$

where the $j$th component of $Q$ represents the final QoS score of the $j$th web service $w_{j} \in WS$ in a set of functionally-equivalent web services $WS = \{w_{1}, w_{2}, \ldots, w_{n}\}$, $n \geq 2$, and $1 \leq j \leq n$.

4. Implementation and Evaluation

The architectural design of intelligent QoS-based service selection contains three types of architectural elements [11] – data, components, and connectors – as its building blocks. We also provide two levels of abstraction for the server-side architectural design. (See Figure 2).

![Figure 2 Overall architecture](image)

We have implemented the architectural design and produced a proof-of-concept prototype for QoS-based service selection. Firstly, in order to collect more users’ opinions for the QoS mode (i.e. the AHP hierarchy), the Wiki UI provides a convenient collaborative editing environment, where any interested users can have their own thoughts posted. As shown in Figure 3, users can quickly establish potential PQA and LQA of the QoS model by clicking the ‘edit’ link for each item. The AHP hierarchy is automatically generated as a ‘Table of Content’ for normal Wiki headings.

![Figure 3 Wiki platform UI](image)

In order to evaluate the scalability of the AHP solution, we have designed an experiment, where the response time is measured given the increased number of reviewers during the AHP comparison phases. It examines the sensitivity of the AHP eigenvector...
response time when the system load (i.e., the number of the users – reviewers) increases in at least one order of magnitude. Thus, four test runs have been conducted based on four different matrix orders (4 – 7). This yields four curves in Figure 5. Within each test run, a certain number (20 – 200) of reviewers concurrently submit their comparison matrices to the AHP engine. The engine calculates the derived principal eigenvector and its associated Consistence Rate (CR). The reviewer aggregation processes is omitted so that we can focus solely on the eigenvector test result. The eigenvector and the CR are sent back to reviewers via HTTP response afterwards. When all reviewers receive the result, the TestRun manager then collects time duration of each reviewer and calculates the average response time (in milliseconds) shown as the Y-axis in Figure 5.

5. Conclusions

Selecting the most appropriate one amongst these functionally-equivalent Web services becomes an important issue to solve. In most cases, Quality of Services (QoS) plays a crucial role since the major concerns have been shifted from the ‘function’ to the ‘quality’ of a Web service in service selection. In this paper, we have provided an intelligent QoS-based service selection solution. We apply the Analytic Hierarchy Processes (AHP) as the underlying model to characterize the QoS problem and its associated decision making approach. We have defined four steps of the AHP-based service selection approach including detailed mathematical model, algorithms and their applications in service selection amongst the service community. To validate the conceptual mode and approach, the architectural design and the software prototype are provided as a complete solution by which users can easily conduct the service selection through Web user interfaces. The solution is evaluated against the scalability.

6. References