Applied Soft Computing 41 (2016) 51–65

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A fuzzy framework for efficient user-centric Web service selection

Zahira Chouiref[a,b,d,*], Abdelkader Belkhir[b], Karim Benouaret[c], Allel Hadjali[d]

[a] University of Bouira, Computer Science Department, LIMPAF Laboratory, Algeria
[b] USTHB University, 16000 Algiers, Algeria
[c] LIRIS, Claude Bernard University, Lyon, France
[d] LIAS, ISAE-ENSMA, Poitiers, France

ARTICLE INFO

Article history:
Received 7 February 2015
Received in revised form 2 December 2015
Accepted 8 December 2015
Available online 31 December 2015

Keywords:
Web services selection
Fuzzy sets
Fuzzy inference rules
Preferences
Contextual profile matching

ABSTRACT

With the development of Web technologies and the increasing usage of Internet, more and more Web Services (WS) are deployed over Internet. Therefore, there will be a large number of candidate services for fulfilling a desired task. In the last decade, several WS selection approaches are proposed to cope with this challenge. In sharp contrast to the existing WS selection approaches that focus only on user-specified preferences, in this paper, we propose a flexible and effective WS selection framework, which gives users an adequate way to express their preferences using linguistic terms, and enhance the WS selection by leveraging their contexts and profiles. The satisfaction of the candidate WS is expressed by an objective score that takes into consideration no only the user-specified preferences, but also additional preferences extracted from both his/her context and profile using fuzzy inference rules, so as to improve the effectiveness of the selection. We then introduce an effective strategy that allows for priority between the two kinds of preferences, for ranking candidate services. Experimental evaluation on a real case study demonstrates the effectiveness of our proposed strategy.

1. Introduction

Nowadays, vast repositories that contain huge amounts of WS[1] on all matters of interest, are available. WS play an increasingly important role in enhancing the user interaction in the Web and enterprise search. Also, the large corporations are now founding their business on an abundant use of WS, the number of publicly available services is then envisioned to be increased in the future in a fast rate [1]. Moreover, the service requestors are often faced with a large mass of competing WS that offer "similar" functionalities but they are associated with "different" constraints, and they are needed to select the best ones with required functionalities and the highest desired quality. Finding optimal Web services – among a collection of services – is known as the Web service query optimization problem in the community of service computing. A great attention has been paid to this problem in the last decade, see for instance [2–4].

On the other hand, the user preferences play a major role in the customization of the selection process. Service requestors usually have varying preferences for the features depending on their context and their profile they find themselves in, and of course different requestors will also have different preferences. In practice, it is difficult to predict how many services' properties are available, and additionally the type of preferences that the user must define it. The more the user is able for expressing his/her needs (preferences), the better the expected outcomes quality is.

A good service selection mechanism should not only allow expressing the preferences that the user defines them for each query attribute, but also deducing additional preferences from his/her situation. Augmenting the user's query by such preferences allows for highly improving the result's quality. On the other hand, it is important to help the users to formulate their preferences in human-like language. A more general and crucial approach to model the preferences in this way is based on fuzzy sets theory [5,6]. Fuzzy sets that are very appropriate for the interpretation of linguistic terms, constitute a convenient way for users to express their preferences. For example, when expressing preferences about the price of a hotel, often the users employ linguistic terms like "rather cheap", "affordable" and "not too expensive".

In this paper, the proposed study falls within WS query optimization and resorts to advanced soft computing techniques to provide user with the best WS that really serve his/her complex and

* Corresponding author at: University of Bouira, Computer Science Department, LIMPAF Laboratory, Algeria. Tel.: +213 561528845/+33 688884899.
E-mail addresses: zahirachouiref@gmail.com (Z. Chouiref), kaderbelkhir@hotmail.com (A. Belkhir), karim.benouaret@univ-lyon1.fr (K. Benouaret), allel.hadjali@ensma.fr (A. Hadjali).

[1] http://www.w3.org/2002/ws/.

http://dx.doi.org/10.1016/j.asoc.2015.12.011
1568-4946/

diverse needs. First, to better personalize the WS selection we make use of two families of user preferences: (i) explicit preferences and (ii) implicit preferences. The former stands for preferences that are formulated in the initial user query. Such preferences are expressed thanks to gradual predicates modeled by means of fuzzy sets. The latter involves preferences that are inferred from information related to context and profile of the user. An appropriate inference mechanism, borrowed from the fuzzy/approximate reasoning field, is used. Second, to obtain an overall degree of the similarity between the service requestor preferences and the service constraints, a priority-based method is introduced to aggregate the elementary similarities. By this way, one can assign more priority to explicit preferences than implicit preferences. To the best of our knowledge, this is the first time that a WS selection process takes into account both explicit and implicit user preferences.

Our aim is to set up an effective WS selection framework which supports the user in finding the most suitable services and return the top-*k* WS according to his/her preferences without much effort from him/her, with taking into account both his/her context and profile.

1.1. Challenges

The main challenges involved in selecting the top-*k* WS are threefold:

- First, how to model the information related to both the required service and the offered services to select the relevant ones that can contribute to answering the user query.
- Second, how to efficiently retrieve the most relevant WS that better satisfy both explicit and implicit user's fuzzy preferences.
- Third, how to rank-order the results services (knowing that the two kind of preferences are not of the same importance w.r.t. end user) and then generate the top-*k* WS.

1.2. Contributions

The made key contributions in this paper are summarized as follows:

We already tackled the first challenge by proposing in [7] a service model as well as a Contextual Profile (CP) model that contain several dimensions able to describe the most information characterizing required/offered service. In the current work, we have also strengthened the CP model, which has been stored in a hierarchical data structure called: *contextual profile ontology* with taking into account the gradual nature of the parameters related to CP and preferences expressed in a human-like language.

In this paper, we focus mainly on the second and the third challenges. Thus, the service selection process is accomplished by the integration of three dimensions: user profile, user context and explicit/implicit user preferences into the end user querying process. Explicit Preferences (EP) are provided in the original user query and modeled thanks to the (fuzzy) membership functions. Due to the commensurability property, the elementary satisfaction degrees are aggregated using one of the fuzzy aggregation operators (e.g., min, product operator, etc.). Then, a list of WS satisfying this family of preferences is returned in a discriminated way. Implicit Preferences (IP) are inferred from user's CP and used to expand the original query in order to retrieve most relevant services. To do so, our framework makes the use of an appropriate fuzzy inference mechanism to faithfully capture the IP. Such preferences operate on the list of WS resulting from the processing of EP. Then, an overall score that takes into account not only the EP but also the IP is defined.

Our proposal allows selecting optimal WS in a qualitative (the *k* WS with the overall scores greater than a given user threshold) or a quantitative (the top-*k* WS) way. An efficient WS selection algorithm which returns the best *k* services according to the one of the above points of view, is provided. Note that the overall score is based on the degree of satisfaction of the requester's EP as first priority, and the one of the IP in a second priority. Experimental results, designed to evaluate our WS query optimization method and ranking mechanism, demonstrate significant performance improvements over the traditional optimization approach.

Outline. The remainder of this paper is structured as follows. In Section 2, we provide first a review of existing proposals for WS selection problem and, second we give some basic notions about ontologies and fuzzy ontologies. Section 3 presents a case-study, which is used as a running example throughout the paper. In Section 4, the modeling aspect related to WS, contextual profiles and preferences is addressed in a detailed way. Section 5 discusses the preference query processing in a real case study related to the field of restaurant business and the ranking mechanism as well. Section 6 describes an experimental study to show the effectiveness of our proposal. Finally, Section 7 concludes the paper and provides some research lines for future work.

2. Related work and background

In this section, we first give an overview of the most closely related work that exist in the literature, then we present some background on ontologies and fuzzy ontologies.

2.1. Related work

In a service oriented domain, the properties that specify a service can be classified as (1) functional *(Input, Output, Precondition, Effect-IOPE)*, (2) behavioral and (3) non-functional *(Quality of Web Service – QoWS, property that identifies the technical standards or protocols for implementing services, categorization, etc.)*, [8,9].

Service selection as the problem of selecting the most appropriate WS from a pool of available ones that best match the functional and non-functional requirements and constraints specified by the requester, has been researched extensively. Existing service selection approaches reviewed below can be summarized in five classes (see Fig. 1) depending on whether that they rely on (i) syntactic and semantic level of service description, (ii) functional and non-functional parameters, (iii) context information, (iv) user preferences and so on.

In the traditional WS matching, service functionalities description is based on syntax that has low recall and precision due to the management of only syntactic service descriptions (search engines index services by the keywords found in their WSDL [10] interfaces), resulting in an inefficient search mechanism [11]. The syntactic discovery approaches are insufficient for creating meaningful descriptions of WS [12]. Semantically enhancing WS descriptions with ontologies[2] helps overcome this drawback [13]. In [14], matching algorithms of semantic approaches are based on ontology concept similarity using expressive semantic Web service languages such as OWL-S,[3] WSMO[4] and so on. Other researchers [15] decided to build extensions on UDDI [10] repositories to allow expression of non-functional properties, thus facilitating better efficiency.

Most of existing approaches [16,17] focus on satisfying the *functional* requirements for functionality (*the functional properties describe what the service can do, e.g. BookFlight, SellingMovies*). A candidate service in search result output although may be fully

[2] See more details about ontology concept in Section 2.2.
[3] http://www.w3.org/Submission/OWL-S/.
[4] http://www.w3.org/Submission/WSMO/.