SERVICE RECOMMENDATION BASE ON SERVICES BACKGROUND AND USERS FEEDBACKS

Mohammad Motamedi*

Afshin Salajegheh**

Abstract

In order to select services in effective manner, Considering Service`s Quality Attributes among Services having same functionality is key parameter. To achieve this goal We need fair and dynamic methods for calculate and evaluate service`s qualities. Important Rule is that this method must be fast and with least possible overhead. In Service oriented environment, most service requestors are unknown to service provider so recommending services base on content base filtering and collaborative filtering method are not useful. In this paper we will demonstrate a method for recommending services to unknown requester base on services quality attributes and their background and requesters preferences.

Keywords-component; Service, QOS, Ranking, Service Recommendation, Web Service

* MS Student, Azad University South-Branch, Student of Aptech computer education.

** Department of software engineering, Islamic Azad University Tehran south branch, Tehran – Iran.

Introduction

Problem of finding required service was about process of submitting request to UDDI and finding appropriate service. In this field there are many solutions as Syntactical Matching [14] or Requirement Extracting [17] and so on. Using these solutions can help us to find appropriate services in UDDI but with increasing number of available services result may be consist of many Services with similar functionality. In this step we need to make decision to chose the service , but if there could be a step which can rank these services base on our goal`s characteristics, would be very pleasant or a step which can filter result base on our preferences would be very helpful . In very easy scenario we may need a service for our software as developer which can tell us the weather temperature, searching UDDI may result of finding many services with getWeather() function , in this case we may have limitation (base on our inputs from other components of our software) that required service should accept location Longitude and latitude as input parameters and we need the

Result to be in Fahrenheit. In this case we need feature base recommendation or filtering [13].

Even with this step we may face too many options, in this step we need result to be ranked base on our Quality parameters. So we need a recommendation system which its input is Services with same functionality and requester quality Parameter (Non functional requests) and its output is Services which are ranked descending base on their net value for the requester.

Our proposed Algorithm is aimed this step in which can rank services to ease the selecting phase. It means request will be transmitted to UDDI and the appropriate services will be sending to our algorithm to rank services.

In order to rank services, The Initial parameters we have, are requester`s preferences and Services` providers Quality parameters. But service provider's parameters have to be validated. This task is not easy because these parameters have dynamic nature so they need to be validated constantly. One solution is to have some institute to validate services but this task is costly and has many other problems. In Our approach Users will validate services. Service consumers can vote for their services that they used. Another parameter we use is Contracts histories. We save each service Consumer Quality parameters and selected service. We then conduct a data-mining process on this data to use in recommendation process. The idea is base on those Users with similar preferences that want to select similar services. We believe that service requestors definitely have chosen services base on a very careful examining and searching among available

options. We log Requesters` asked quality parameters and the service which they finally chosen. With our method we can recommend services to service requestor when a requester with similar needs to them has chosen the service.

Related Works :

Before Service oriented, a recommendation system for software function discovery have been proposed in [9] which is based on user similarity methods. In [7] Proposed method recommends web service to software developer. Every clients have to install a remote client, which is responsible for tracking users behavior and submitting user`s request. In this system User can optionally provide feedback about the service. In [8] an eye catching algorithm have been proposed which consist of finding services score (As another way of calculating score for services base on their quality parameters) and then binary search tree will be constructed . And the right most in node in the tree have the biggest score which will be suggested to user. To add extra option to [8] Like Our proposed Algorithm in this paper it would be better to find score for requested service and then search BST to find nearest service to user request. It is not advisable to conduct every request to best service because of traffic problems and also best service are not always best choice because it may have much more fees than what we want. In [4] Using similarity calculation is use among keyword to find and recommend service to requestors have been proposed. It also recommends other related operation about requested Service to requester. In [19] Algorithm consist of 4 step , Finding relevant services, filter poor ones, rank Services, Updating user preferences based after contract this algorithm is very interesting, but like most other algorithms needs registered users.

Proposed Algorithm:

We use similar Quality parameters of services with similar functionality. User can submit his/her request base on these quality criteria. For example user may request service with response time 1ms and cost 10$/m. Also requester can express which parameter is more important by ranking them.

Now we have services with quality parameters and requester required parameters. We consider requester parameters as a service, now we have:

Service 0 (Q1,Q2,….Qm)

Service1(Q1, Q2,…,Qm)

.

.

.

Qn(Q1,Q2,…Qm)

Which S0 is a virtual service with Quality parameters like User`s preferences.

Now we compute score of each service. Score of a service is a number indicating service net quality grade which will be calculated with service provider's quality parameters. There are some methods for calculating this score, we utilized method which used in [11].

{Score0,Score1,Score2,…,Scoren}

It is obvious that scores near Score0 are more like that they are appropriate for requester.

Now We Utilize Users Votes for Services. Voting for a service has two limitations and one property.

Limitations:

1. Only Users which has contract with service can vote.
2. Each Vote will have unique contract and user.

Property:

Votes are a number between 1 and 7. (Inspired by Ringo music recommendation system)

If a service has not any vote, 4 will be applied as average.

If service has votes, we use modified version of base case reasoning method.

Instead of dividing sum of votes on count of users, because we don't have number of users we divide on maximum number of users which have been voted a service.

If result is less than 4 then the service score will be distanced from Score0,

If result is greater than or equal to 4 the service`s score will be push toward Score0, new score will be :

Scorei = scorei X result/7 * (score0-scorei)

X is operator which could be + or – base on score0.

The reason that services with no rating will give 4 and therefore push them toward score0 is inspired by no news is good news.

Now we conduct data mining process, the data mining algorithm is modified algorithm proposed in [12].

As mentioned before we log history of contracts consist of requested quality parameters and final selected service. We map Customers attributes to requested quality parameters and books are mapped to selected service. It means we use requested Quality parameters as customer's characteristics and attributes. As Result we may have some rules. In order to use these rules we have two approaches .At first approach if any of services match a consequence of a rule, we simply set this service score equal to score0 which means giving first rank to this service.

The benefit of applied data mining algorithm is that we can use Users priority on the quality parameters , as mentioned, User can specify in his/her request. Another thing to note is we have just 1-Large itemset in data mining process which will result in less required time.

Another approach is to utilize Method used in for recommendation base on data mining process in [18] .In this paper Score for article I is calculated as

$$score_{artide_i} = \sum_{\text{all rules that recommend article}_i} (support_{rule} * confidence_{rule})$$

We calculate this Score and apply it like rates to the scores in this way that services which have any rule we calculate above formula and push these services score toward service0.

Finally we sort Service Scores from least distance to Score0 to most one. The result will be the sorted list and will show to Requester.

We developed software to simulate the Algorithm. Database consist three tables. One table contains Quality parameters for Services. Another table log contracts log and the other keeps User`s votes.

And the last one contains Services quality attributes. Users can express their preferences and their order of preferences. After submitting the request process of recommendation will be applied and available services will be shown to users in order of their distance to user request. Users can see their score and score of available services. Finally this may end with a contract between requester and one of the services in the list. In this case we will log the requestor preferences and service which selected for future recommendations. Users can vote for service which they have selected in order to improve the recommendation process.

Experiment show that with increase of contracts, the 3 first recommendations are most of the time the best choices. The speed of the data mining phase because the algorithm needs just one scan of database and also in this scenario we have just 1-large itemsets is reasonable. But we also add sort of rule maintenance methods to improve the performance.

Summery

In this paper we introduce a recommendation base on requests similarity. It consists of conducting data mining algorithm on contracts. We believe that one of the best ways to evaluate services performance and their commitment to their advertised quality attributes are consumers .The total process in nutshell is, we will find services` Score base on their quality parameters and each parameter importance, then these scores will be scaled base on previous user Rates and the service previous contracts. Result is list of available services ranked by their scores.

REFERENCES:

[1]Bean , James (Jim) ,SOA and WebServices InterFace Design (Principles,Techniques,Standards) . 2010 by Elsevier Inc

[2] Reference Model for Service Oriented Architecture 1.0 Committee Specification 1, 2 August 2006

[3]Qing Gu van Vliet, H. , SOA decision making - what do we need to know ,Sharing and Reusing Architectural Knowledge, 2009. SHARK '09. ICSE Workshop on ,16-16 May 2009 ,25 – 32

[4] Laiping Zhao a , Yizhi Ren c , Mingchu Lib , Kouichi Sakurai Flexible service selection with user-specific QoS support in service-oriented architecture

[5] Mike P. Papazoglou · Willem-Jan van den Heuvel,Service oriented architectures: approaches, technologies and research issues,The VLDB Journal (2007)

Volume: 16, Issue: 3, Publisher: Springer-Verlag New York, Inc., Pages: 389-415

[6] Asit Dan, Robert D. Johnson, Tony Carrato SOA Service Reuse by Design,SDSOA '08 Proceedings of the 2nd international workshop on Systems development in SOA environments ACM New York, NY, USA ©2008 ,P 25

[7] Natallia Kokash, Aliaksandr Birukou and Vincenzo D'Andrea, Web Service Discovery Based on Past User Experience, BIS'07 Proceedings of the 10th international conference on Business information systems Springer-Verlag Berlin, Heidelberg ©2007 ,p. 95-107

[8] Minhyuk Oh ,Jongmoon Baik Sungwon Kang,Ho-jin Choi , Efficient Approach for QoS-Aware Service Selection Based on A Tree-Based Algorithm,Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008),p6

[9]Naoki Ohsugi ,Akito Monden ,Ken-ichi Matsumoto, A Recommendaion System for Software Function Discovery,APSEC '02 Proceedings of the Ninth Asia-Pacific Software Engineering Conference IEEE Computer Society Washington, DC, USA ©2002

[10] Philip Bianco ,Grace A. Lewis,Paulo Merson "Service Level Agreements in Service-Oriented Architecture Environments" http://www.sei.cmu.edu/ 2008

[11] Yutu Liu, "QoS Computation and Policing in Dynamic Web Service Selection", WWW Alt. '04 Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters ACM New York, NY, USA ©2004

[12]Sheikh, L.M.; Tanveer, B.; Hamdani, M.A.; FAST-NUCES, Mining interesting association rules from customer databases and transaction databases Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International

[13] Eui-Hong, George Karypis Feature-Based Recommendation System Proceedings of the 14th Conference of Information and Knowledge Management (CIKM), 2005, pp. 446 - 452,

[14] Blake, M.B. Nowlan, M.F. A Web Service Recommender System Using Enhanced Syntactical Matching Georgetown Univ., Washington Proceedings of the 14th ACM international conference on Information and knowledge management CIKM 05 (2005) Pages: 446

[15] Umardand Shripad Manikrao, Dynamic Selection ofWeb Services with Recommendation System Next Generation Web Services Practices, 2005. NWeSP 2005. International Conference on 22-26 Aug. 2005 ,5 pp

[16] Le-hung Vu , Manfred Hauswirth , Karl Aberer QoS-based Service Selection and Ranking with Trust and Reputation Management In Proceedings: OTM Conferences,2005,p18

[17] Da Ning and Rong Peng A Requirements Recommendation Method Based on Service Description Springer - CloudCom 2009, LNCS 5931, pp. 656–661, 2009

[18] WeiYang LIN,Sergio A.Alvarez,Carolina Ruiz Efficient adaptive-support Association rule mining for recommender systems Kluwer academic Publishers , Netherland 2001

[19] QoS-based Service Selection and Ranking with Trust and Reputation Management?