ARTICLE IN PRESS

Neurocomputing ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Neurocomputing

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

Proactive service selection based on acquaintance model and LS-SVM

Hu Jingjing [a,*], Chen Xiaolei [a], Zhang Changyou [b]

[a] School of Software, Beijing Institute of Technology, Beijing 100081, China
[b] Institute of Software, Chinese Academy of Science, Beijing 100190, China

ARTICLE INFO

Article history:
Received 27 July 2015
Received in revised form
16 October 2015
Accepted 3 November 2015

Keywords:
Service selection
Acquaintance model
Negotiation
LS-SVM

ABSTRACT

Current service selection is unable to perform proactively. When a service provider overloads, the services list is ever-lengthening, which leads to backlog and failure of service composition. To compensate for this deficiency, this paper improves the proactive service selection. In this strategy, the service provider analyses a time series of services received to forecast the backlog and consign services to the others through a negotiation process. The least squares support vector learning is used to predict a random list of services, and an acquaintance model (AM) makes a consigner allocate backlog services to other providers with high degree of relationship. The backlog of services by forecasting is entrusted to the provider who can implement the same service, and negotiation between the providers with the same role would allow generation of a new service selection solution before a fault occurs. Experiments showed that the least squares support vector machine (LS-SVM) algorithm was more accurate in predicting a services list and a negotiation mechanism using AM decreased communication time effectively, which improved the success rate of service selection and reduced the execution time of service composition.

1. Introduction

During the dynamic execution of service computing, web service composition adopts a reactive service selection [1]. Reactive service selection is a method of choosing provider determined by the service requestor's comprehensive decision making. The advantage of reactive selection is that it integrates the requestor's preference fully and suits for service composition with less requestor and more providers [2]. There is some deficiency for the method because once chosen as a candidate of the service executor, the service provider will complete the service task instead of attempting to reassign the service, which could lead to the backlog of unhanded services. If a provider becomes overloaded for some reasons, the services list will keep growing, and the provider would reply with unable handling to the requestor resulting in a service interruption [3]. In the environment of big-scale service composition with massive service requestors and providers, solving the problem of how to minimize such service interruptions is especially challenging because each service requestor and provider may be distributed in different architectural patterns within organizations, so that the faulty combination cannot be shut down in time. Therefore, the method of service selection is shifting from the choosing of the requestor to the arrangement of the provider.

Proactive service selection is a two-way choice of both the requestor and provider based on reactive selection. It can balance the loads of providers and handle service composition problem with a large number of requestors and providers. Thus leads to the complexity of service selection [4]. In addition, proactive compensation strategy is needed to ensure that when unexpected accidents occur, the quality of service will not be adversely affected. In real-time service composition, proactive service selection is becoming a new trend.

Traditional reactive service composition is developed through communication between the provider and requestor with different roles, while interaction with the same role is neglected [5]. In proactive service selection, the negotiation process between the providers is vitally important. Massive interaction will reduce the efficiency of service selection. The acquaintance model (AM) can be built to improve the efficiency [6]. AM is a kind of collaboration model similar to human society, which is used to represent information about the resources and capabilities of other service providers. AM consists of an acquaintance list and a set of rules. If it needs to cooperate, the acquaintances are evaluated according to the rules firstly, and high evaluation is preferred.

It was proposed that a web service composition system could respond to a backlog fault by using each provider to monitor the queue length of their responsible services [7]. If the length exceeds a predefined threshold, the service provider will send the message to clear the backlog of service items and terminate the execution of service composition in advance. In reality the length of services list is random and sample size is limited, so that the accuracy of prediction is lower and validity of proactive service selection becomes worse [8]. Support Vector Machine (SVM) is the further development of

* Corresponding author.
E-mail address: hujingjing@bit.edu.cn (H. Jingjing).

http://dx.doi.org/10.1016/j.neucom.2015.11.119
0925-2312/© 2016 Published by Elsevier B.V.

machine learning based on statistics [9,10]. A service provider can effectively predict services list using SVM regression.

In this work, we improve the proactive service selection. The original contribution of this paper is that least squares support vector learning is used to predict services list, and AM to allocate backlog service to other providers with the same role. The backlog of services may be entrusted to the provider who can implement the same service, and negotiation between the providers with the same role would allow generation of a new service selection solution before a fault occurs. Experiments verified that the prediction algorithm was more accurate using LS-SVM, and the negotiation strategy was more efficient with AM.

This paper is organized as follows. In the next section the proactive service selection model is presented. The third section describes the service selection based on LS-SVM. The fourth section shows the performance evaluation. The conclusion is given in the last section.

2. Proactive service selection model

Considering the real-time characteristics of dynamic service composition, web service is defined as follows.

A web service is a 7-tuple $ws=(S, s_0, F, M, X, I, R)$, where

- S is a set of finite states.
- $F \subseteq S$ is a set of termination states.
- $s_0 \in S$ is an initial state ($s_0 \in F$)
- $M = M_\varepsilon \cup \{\varepsilon\}$ is the set composed by message set M_ε and empty element ε. For any message $m \in M_\varepsilon$, Polarity (m) is used to identify the attributes of message m. If m is receiving message, the Polarity $(m) = ?$; and if m is sending message, then Polarity $(m) = !$.
- X is a variable set of finite clock.
- $I: S \rightarrow \Phi(x)$ assigns a clock constraint of $\Phi(x)$ for each state of S, when the state does not meet the constraints of the clock, it must be able to perform the migration away from the state.
- $R \subseteq S \times M \times \Phi(x) \times 2^X \times S$, for any migration $r \in R$, suppose $r=(s, m, \phi, Y, s')$, where s is the source state, s' is the target state; m is a message action that can trigger migration; ϕ is the set of conditions enabling migration; and $Y \subseteq X$ is the set of the clock variables which needs to be reset when migration happens.

In services list prediction, the threshold value (h) can be calculated by the service state transition in unit time (τ), and the computation is defined as follows.

$h = \tau / \max(\varnothing(y') - \varnothing(y))$. It shows that in the execution of service, y' is the clock in s' with message 'begin' and y corresponds to s with message 'end' while the Polarity begins with '!' and ends with '?'. For example, suppose τ is one day and $\max(\varnothing(y') - \varnothing(y))=15s$, then h is 5760.

Service providers can be defined as a tuple $sp = <p, \text{Frlist}(p)>$, where p is the provider's name; and Frlist (p) is the friend list of the provider.

2.1. Acquaintance model

The execution of negotiation is implemented by the communication between service providers. Similar to the learning process of human beings, this process requires communication and interaction provided in an acquaintance model (AM) [11].

The item of Frlist is included in the attributes of provider, which represents the list of friends, i.e., the information list of all the acquaintances, donated by $<l_1,\ldots,l_i,\ldots,l_n>$, among which l_i, $i \in \{1,2,\ldots,n\}$refers to the information of the ith acquaintance. li is donated by $li=<p, degree>$, in which acquaintance name and degree of relationship are represented separately where *degree* is the close degree between providers, with range of [0,1].

The following rules are satisfied for acquaintance in service selection.

(i) Rule of collaboration.

When an agreement is achieved in a service consigning negotiation, which means the provider is friendly to the consignor. In this case, the degree for the service provider denoted by R will be increased, $R_i=\min(R_i+\delta,1)$ (the default of degree is 0). δ is a relational increment, which indicates that the degree between the acquaintances grows with a successful service consignment. ($0<\delta<1$).

(ii) Rule of forgetting.

When a service provider sends a consignment request to another that has established friendly relation, no reply or disagreement is answered, which shows their friend degree is descending, $R_i=\max(R_i-\lambda,0)$.λ is a relational reduction, which indicates that the degree between the acquaintances decreases with a service consigning failure. ($0<\lambda<1$).

(iii) Rule of recommendation.

When a service provider gets a consignment request from others, perhaps he cannot give an offer, but he can recommend his friends to the consignor. The recommended provider can then be added to his friend list as one of his friends.

2.2. Negotiation primitives

The negotiation process is distinguished from other activities and modeled based on speech-act theory [12]. The process corresponds to possible conversation. The current service provider is called consigner and other candidate provider is referred as performer. A consigner can communicate with many performers. When a consigner needs negotiation, a negotiation cycle is activated. The primitive of negotiation model for consigning is shown as follows.

Ng_C_inform_start—Consignor announces the beginning.

Ng_C_cfp-ws—Announce the proposal of web service to be performed.

Ng_P_not_understand—Performer can not understand the consignment content.

Ng_P_propose—Performer provides his execution services.

Ng_C_accept_proposal—Consignor accepts the proposal.

Ng_C_reject_proposal—Consignor rejects the proposal.

Ng_C_inform-end—Notify other performers negotiation is over.

Ng_C_request—Make agreement with the selected performer.

The activity of service selection negotiation is shown as Fig. 1.

2.3. Test of acquaintance model

In order to verify the model, the experiments were done. There are 100 service providers with the same role. The backlog rate of each provider is 20%. The initial list of friends is 3. It executes 20 experiments for negotiation based on the acquaintance model and a general model. The comparison of information interaction's number in the process of each negotiation is shown in Fig. 2. The above experiments are performed 20 times. Success rate of service selection is the average value of whether the negotiation is successful or not in each time. The comparison of success rate is shown as Fig. 3, where $\delta=0.05$, $\lambda=0.1$, $R_i=0.6$.

On average, the information interaction number of negotiation process based on the acquaintance model decreases by 71% and the success rate of service selection increases by 19%. It simplifies the communication process effectively.

Negotiation based on AM can decrease the range of interaction between service providers, and reduce the amount of communication. The rules for AM can further promote the efficiency of service consigning due to learning from the successful negotiation experience.

Please cite this article as: H. Jingjing, et al., Proactive service selection based on acquaintance model and LS-SVM, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2015.11.119