

## A Reputation Evaluation Technique for Web Services

Nianhua Yang<sup>1</sup>, Xin Chen<sup>2</sup> and Huiqun Yu<sup>3</sup>

<sup>1</sup>*School of Business Information Management, Shanghai Institute of Foreign Trade,  
Shanghai 201620, China*

<sup>2</sup>*School of Computer, Beijing Information Science and Technology University,  
Beijing 100101, China*

<sup>3</sup>*Department of Computer Science and Engineering,  
East China University of Science and Technology, Shanghai 200237, China  
cnynh@163.com, chenxin@bistu.edu.cn, yhq@ecust.edu.cn*

### **Abstract**

*To select a most trustworthy one among web services with the same functionality, a trust and reputation management framework for web service selection is proposed. A reputation evaluation algorithm is proposed for the new added web service based on the similarity theory. Similarities and trusts are used as weights for computing reputations from different recommenders. Updating algorithms for trusts and reputations are proposed.*

**Keywords:** *trust, reputation, web service selection, similarity*

### **1. Introduction**

Service oriented enterprise applications have emerged in recently years. They are realized by composing web services used as fundamental building blocks. A lot of services providing similar functionalities have been deployed. Therefore, how to select a right web service is an important issue for service composition as some of services may not provide what they promised. Hence, trust and reputation become important and essential criterions for best web service selection.

Reputation is what is generally said or believed about a person's or thing's character or standing [1]. It is a subjective assessment for a web service based on the user's own experience and recommendations from neighbors. A weight for a recommendation depends on how much the user trust the recommender. Reputation or trust will be reduced over time.

A good trust and reputation framework should consider following issues simultaneously. Firstly, reputations for a new added service should be valued. Secondly, different weights should be assigned to reputations from different recommenders. Last but not least, trusts and reputations should be updated after each interaction.

This paper proposes a trust and reputation management framework for web service selection. A reputation evaluation algorithm is proposed for the new added web service based on the similarity evaluation method [2]. Similarities and trusts are used as weights for computing reputations from different recommenders. Updating algorithms for trusts and reputations are proposed.

Section 2 surveys related work. Section 3 details the proposed trust and reputation management framework for web services. Experiments are described in Section 4. Section 5 concludes this paper.

## 2. Related Work

Three main classes of approaches for trust based web service provision is identified in [3]. They are direct experience based approaches, Trusted Third Party (TTP) based approaches and hybrid approaches that combine techniques of the previous two classes. Direct experience based approaches are based on evaluations given by the user's own direct experience with the target service [4]. Dragoni [3] points out that these approaches are not appropriate for open systems, since the service consumer can not evaluate a web service with no direct experience. TTP approaches are based on the evaluation results provided by trusted party. They argue that web services in a particular domain will aid each other in evaluating the initial reputation of the new added web service [5].

Hybrid approaches combine direct experience based approaches and TTP based approaches. The approach proposed in this paper belongs to this category. Jøsang et al. [6] combine Bayesian reputation systems with a trust model for evaluating the quality of service in a single framework. But it is based on a centralized trusted reputation center [3].

Nepal et al. [7] propose a fuzzy trust evaluation approach for web services. But they do not give an effect method for evaluating trust of a new one. Shao et al. [2] propose a similarity computing algorithm for web services and their consumers based on Euclidean distance [8] theory. Consumers' similarities are used as weights of indirect experiences. Similarities between target web service and other ones are used as weights for QoS prediction of the target web service. The similarity computing algorithm proposed in [2] is used in this paper for evaluating consumers' similarities and services' similarities.

## 3. Trust and Reputation Management Framework

### 3.1 Service Selection Model

The web service selection model, in Fig. 1, consists of two clusters, "Consumers" and "Web services". Web services in the same cluster hold the same functionality. Properties of a web service, claimed by the provider, can be described based on the ontology.

Definition 1. A web service ontology is a 4-tuple,  $WS = (CP, I, O, QoS)$ .  $CP$  is the set of common properties. Service functionality and its classification are included in  $CP$ .  $I$  is the set of parameters for input.  $O$  is the set of parameters for output.  $QoS$  represents the quality of service. QoS properties in a web service mainly include execution Time ( $T$ ), Cost ( $C$ ), Reliability ( $R$ ), Security ( $S$ ) et al.. In this paper, we define QoS as a 4-tuple,  $QoS = (T, C, R, S)$ .

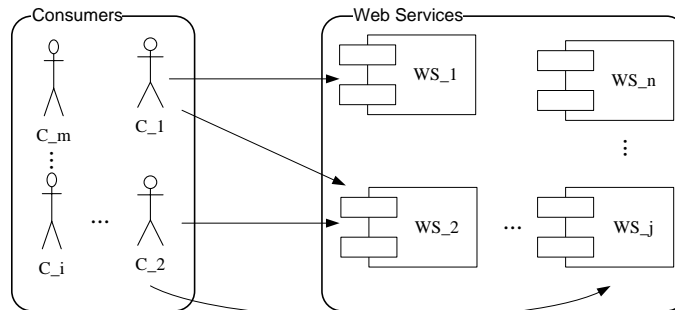


Figure 1. A Service Selection Model

Each consumer holds a compositional reputation for a web service. The compositional reputation is the weighted average of reputations based on the historical direct experiences. Each consumer also holds trust evaluations for the neighbors. The trust value is computed based on the degree of deviation between the recommended reputation value from the neighbor and the value given by the consumer after calling.

### 3.2 Similarity Mining

Euclidean distance [8] based similarity mining approach overcomes some disadvantages of other approaches when it is used to compute similarity of non-linear similar data. This section uses the similarity computing method proposed in [2] which is based on Euclidean distance.

$S = \{s_1, s_2, \dots, s_m\}$  is the set of services with the same function.  $U = \{u_1, u_2, \dots, u_n\}$  is the set of service consumers.  $P = \{p_1, p_2, \dots, p_n\}$  is the set of QoS properties of a service.  $Q_{ij} = \{q_{ij}\}$  represents the property values of  $p_j$  for  $s_i$ , such as shown in Tab. I. Blanks in the table represent no reputation information of the target services.  $R_{i,j} = \{r_{ij}\}$ ,  $r_{ij} \in \mathbb{R}$ , represents the reputation for  $s_j$  in  $u_i$ , where  $0 \leq r_{ij} \leq 1$ , such as shown in Tab. II.  $r_{ij}$  is proposed by the service provider of  $s_j$ .  $\mathbb{R}$  represents the set of real numbers.

**Table 1. Reputations of Services Evaluated by Consumers**

	$s_1$	$s_2$	$s_3$	$s_4$
$u_1$	0.8	0.7	0.9	0.66
$u_2$	0.75	0.45	0.88	0.78
$u_3$	0.6	0.6	0.92	
$u_4$	0.55		0.95	0.81

**Table 2. Properties of QoS for Web Services**

	$p_1$	$p_2$	$p_3$	$p_4$
$s_1$	3	8	0.9	0.96
$s_2$	2	15	0.85	0.9
$s_3$	0.5	30	0.86	0.95
$s_4$	0.2	50	0.9	0.93

Each row (or column) is regarded as a node [2]. Thus, the similarity between two nodes can be represented by the Euclidean distance. The more distance two nodes are, the less similarity they become. Equation (1), proposed in [2], is used to compute Euclidean distance.

$$w = \frac{1}{\left(\sum_{i=1}^n (X_i - Y_i)^2 / n\right)} \quad (1)$$

In Eq. (1),  $X$  and  $Y$  are vectors.  $X_i$  (or  $Y_i$ ) represents the value of the  $i$ -dimension in the vector. Shao et al. [2] fit the missing dimensional place with mean value of the neighbors.

### 3.3 Direct Experience Based Reputation

Let  $r_{i,j}^{(n)}$  be the reputation for service  $j$  given by the consumer  $i$  after the  $n$ -th calling. The comprehensive reputation of  $n$ -times calling is represented as  $R_{i,j}^{(n)}$  which is computed by Equation (2).

$$R_{i,j}^{(n)} = \begin{cases} \alpha R_{i,j}^{(n-1)} + (1-\alpha)r_{i,j}^{(n)}, & n > 1 \\ r_{i,j}^{(n)}, & n = 1 \end{cases} \quad (2)$$

$\alpha$ ,  $0 \leq \alpha \leq 1$ , is the history factor. Encouragement and punishment factors for reputation are not considered in this paper for simplicity. Equation (2) is also used as reputation updating formula for service  $j$  from the consumer  $i$ .

### 3.4 Recommendation Based Reputation

When the consumer has no direct experience based reputation about a service, or in order to get much information about the reputation of a service, recommended reputations from other consumers are important. The reputation of the service  $j$  based on recommendation for the consumer  $i$  can be calculated by Equation (3).

$$C_{i,j} = \frac{\sum_{k=1..n, k \neq i} (w_{i,k}^u \times t_{i,k} \times R_{k,j})}{\sum_{k=1..n, k \neq i} (w_{i,k}^u \times t_{i,k})} \quad (3)$$

$C_{i,j}$  is the comprehensive recommended reputation of the service  $j$  for the consumer  $i$ .  $w_{i,k}^u$  is the similarity between consumers  $i$  and  $k$ .  $t_{i,k}$  is trust evaluation of the consumer  $k$  given by the consumer  $i$ .  $R_{k,j}$  is the direct experience reputation about the service  $j$  given by consumer  $k$ .  $n$  is the number of recommenders.

### 3.5 Reputation of a New Service

If there is no recommender for the reputation of the service  $j$  and no direct experience in the consumer  $i$ . The service  $j$  is regarded as a new added service. The reputation evaluation for the new service is based on the reputations of similar services, given by the same consumer. Services similarity is also computed by Equation (1).

The reputation of a new service for the consumer  $i$  can be calculated by Eq. (4).

$$R_{i,j} = \frac{\sum_{k=1..n, k \neq j} (w_{j,k}^s \times R_{i,k})}{\sum_{k=1..n, k \neq j} w_{j,k}^s} \quad (4)$$

In Eq. (4),  $n$  is the number of services which are called by the consumer  $i$  in history and hold the same function as the service  $j$ .  $w_{j,k}^s$  is the similarity between the services  $j$  and  $k$ .  $R_{i,k}$  is the comprehensive reputation of the service  $k$  given by the consumer  $i$  in history.

For the first added service in a group with the same functionality, the reputation is set to be 0.5 for each consumer.

### 3.6 Integrated Reputation

Reputation for a web service is the weighted average of direct experienced reputation and recommended reputation. It is described in Equation (5).

$$I_{i,j} = \beta R_{i,j} + (1 - \beta) C_{i,j} \quad (5)$$

$\beta$ ,  $0 \leq \beta \leq 1$ , is the weighting factor for reputation based on direct experience.  $R_{i,j}$  is the reputation based on direct experience for the service  $j$  given by the consumer  $i$  in history.  $C_{i,j}$  is the comprehensive reputation of the service  $j$  given by the neighbors of the consumer  $i$ .

### 3.7 Trust and Reputation Updating

After each calling, the consumer will give a new reputation for the called web service. The comprehensive reputation will be updated according to Equation (2). The updated comprehensive reputation includes the influence from the latest reputation given by the consumer.

Trust from a consumer to a recommender should also be update according the degree of deviation between recommended reputation and the given reputation from the consumer after calling. Trust updating method is proposed in Equation (6).

$$T_{i,j}^{(n)} = \begin{cases} 0.5, & n=0 \\ T_{i,j}^{(n-1)} - |R_{j,k} - r_{ik}| \times T_{i,j}^{(n-1)}, & n>0 \ \& \ |R_{j,k} - r_{ik}| \geq \delta \\ T_{i,j}^{(n-1)} + \frac{|R_{j,k} - r_{ik}|}{8} \times (1 - T_{i,j}^{(n-1)}), & n>0 \ \& \ |R_{j,k} - r_{ik}| < \delta \end{cases} \quad (6)$$

$T_{i,j}^{(n)}$  represents the comprehensive trust value of the recommender  $j$  from the consumer  $i$  after  $n$ -times calling.  $R_{j,k}$  is the reputation of the service  $k$  recommended by the consumer  $j$ .  $r_{ik}$  is the reputation of the service  $k$  given by the consumer  $i$  after direct calling.  $T_{i,j}^{(0)}$  is the initial trust value between consumers.  $\delta$ ,  $0 \leq \delta \leq 1$ , is the boundary value for telling whether a recommender is honest or not. Equation (6) shows that a recommender will get trust slowly after providing honest recommended reputations continuously, but will lose trust dramatically after giving a dishonest recommendation.

### 4. Experiment

In the experiment, trusts between users are initialized with double numbers generated randomly between zero and one. Properties and reputations of each service given by consumers are generated randomly. Five consumers and five web services are set initially. After some times calling, a new web service is added with the properties generated randomly. Reputations of the new services given by each consumer are set with zeroes initially.

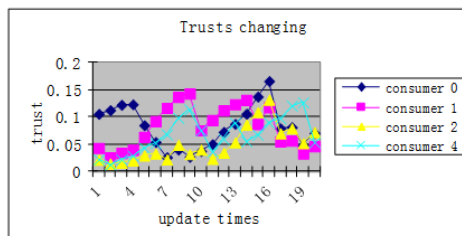


Figure 2. Trusts Changing Processes

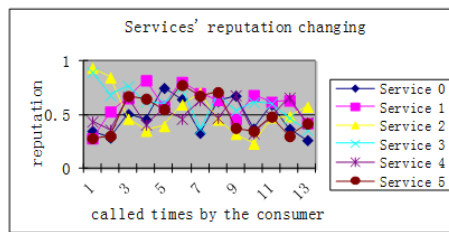


Figure 3. Services' Reputation Changing

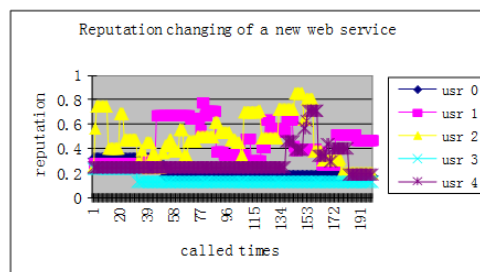


Figure 4. Reputation Changing of a New Web Service

Figure 2 shows trusts changing processes given by the consumer 3. It shows that trusts are lost dramatically with a dishonest recommendation, but got slowly. Figure 3 shows reputations' changes of services updated by a consumer after each calling.

Figure 4 shows the reputation changing processes of a new web service. The reputation is computed according the similar services with the same functionality at the beginning. Then it is updated according to the feedback of the caller.

## 5. Conclusion

Trust and reputation play important roles in trustworthiness evaluation. They provide foundations for estimating a service's trustworthiness based on history information. This paper proposes a trust and reputation management framework for web service selection based on trustworthiness. Similarity mining technique is used for identifying similarity among users or web services. A reputation evaluation algorithm is proposed for the new added web service based on the similarity theory. Similarities and trusts are used as weights for computing comprehensive reputation from different recommenders. Updating algorithms for trusts and reputations are proposed. The framework is applied to select a best one among the web services with the same functionality.

## Acknowledgements

This work was supported by the NSF of China under grants No. 61173048 and 60973107, Shanghai 085 Project for Municipal Universities, Innovation Program of Shanghai Municipal Education Commission under grant No. 12ZS170.

## References

- [1] A. Jøsang, R. Ismail and C. Boyd, "A survey of trust and reputation systems for online service provision", *Decision Support Systems* 43(2), 618-644 (2007).
- [2] L. Shao, Z. Li, J. Zhao, B. Xie and H. Mei, "Web Service QoS Prediction Approach", *Journal of Software* 20(8), 2062-2073. (in Chinese with English abstract) (2009).
- [3] N. Dragoni, "A Survey on Trust-Based Web Service Provision Approaches", *Proceedings of the 2010 Third International Conference on Dependability*, (2010) July 18 - 25; Venice/Mestre, Italy: IEEE Computer Society, pp. 83-91.
- [4] A. S. Ali, S. A. Ludwig and O. F. Rana, "A Cognitive Trust-Based Approach for Web Service Discovery and Selection", *Proceedings of the Third European Conference on Web Services (ECOWS 2005)*, (2005) November 14-16; Växjö, Sweden: IEEE Computer Society, pp. 38-49.
- [5] Z. Malik and A. Bouguettaya, "Reputation Bootstrapping for Trust Establishment among Web Services. *IEEE Internet Computing* 13(1), 40-47 (2009).
- [6] A. Jøsang, T. Bhuiyan, Y. Xu and C. Cox, "Combining Trust and Reputation Management for Web-Based Services", *Proceedings of the 5th international conference on Trust, Privacy and Security in Digital Business*, (2008) September 4-5, 2008 Turin, Italy: Springer-Verlag, pp. 90-99.
- [7] S. Nepal, W. Sherchan, J. Hunklinger and A. Bouguettaya, "A Fuzzy Trust Management Framework for Service Web", *Proceedings of the IEEE International Conference on Web Services (ICWS 2010)*, , (2010) 5-10 July; Miami, FL, USA: IEEE Computer Society, pp. 321-328.
- [8] E. Keogh and S. Kasetty, "On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration", *Data Mining and Knowledge Discovery* 7(4), 349-371 (2003).