WUJNS
Wuhan University Journal of Natural Sciences

Vol.11 No.5 2006 1302-1305

Article ID:1007-1202(2006)05-01302-04

Trust Degree of Web Services and Its Evaluation with Neural Network

□ HUANG Baohua, HU Heping†, LU Zhengding, YAO Hanbing, LI Ruixuan

College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, Hubei, China

Abstract: The concept and definition of trust degree of Web service is given. Evaluating trust degree of Web services with the method of neural network from the aspect of trust history sequence is proposed. The principle of the method, applicable neural network structure, neural network constructing, input standardization, training sample constructing, and the procedure of evaluating trust degree of Web services with trained neural network are described. Experiments show that it is feasible and effective to evaluate trust degree of Web Service with neural network.

Key words: Web services; trust degree; neural network

CLC number: TP 393

0 Introduction

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It is an abstract notion that must be implemented by a provider agent[1]. With the widespread proliferation of Web services, how to select a trustable Web service has become a crucial problem of Web services requester. So requester agent must correctly evaluate Web services and select the appropriate one to use. We define the result from evaluation of Web service as Web service trust degree. It means the grade that the requester trusts of the Web service. Some works have been developed to solve this problem[2,3]. However, the method used is both inefficient and unwise. In this paper, we propose an intelligent, feasible and effective method based on neural network to evaluate trust degree of Web service. So far as we know, it is an initiation.

Trust degree can be obtained from experience of transaction with the Web service of requester or from recommendation of others. The trust degree obtained from experience of requester is surveyed for the following reasons:

- Trust is context specific and multi-faced[4]. Accordingly, requester would be more interested in his or her own experience;
- In some cases, requester must monitor the trust degree of Web services continually[5]. So the experience of requester own can describe the trust degree of Web service exactly.
- Trust degree directly obtained from experience of requester is the foundation of Web services trust degree recommendation.

The experience of requester exchanging with Web services makes a Web services trust history sequence. From the

Received date: 2006-01-10

Foundation item: Supported by the National Natural Science Foundation of China(60403027), the Natural Science Foundation of Hubei Province (2005ABA258), the Open Foundation of State Key Laboratory of Software Engineering(SKLSE05-07)

Biography: HUANG Baohua (1973-), male, Ph. D. candidate, research direction: security and application of Web services and peer-to-peer computing. E-mail:bhhuang@andin.net

† To whom correspondence should be addressed. E-mail: hphu@andin.net

sequence, trust degree that completely reflects the trust status of Web service can be gained by means of model identification. As artificial neural network has the ability of model identifying[6], we can apply it to evaluate trust degree of Web service from trust history sequence.

1 Trust Degree of Web Service

Trust management was proposed to solve the authorization problem in decentralized environment where identifying strange requester is impossible, but traditional security models assume that system can identify requester effectively[7,8]. In Web services, requester faces the problem how to select strange Web service.

In order to get the function of service, requester agent often has to send sensitive data to Web service. From the viewpoint of requester, trust degree of Web service should include security, privacy and QoS[9]. These concepts have some overlap[10], so we get the main aspects to define trust degree of Web service that is show in the following:

● Availability: Whether the Web service is present or ready for immediate use.

● Integrity: How the Web service maintains the correctness of the interaction in respect to the source.

● Confidentiality: Only authorized people or systems can access protected data.

● Privacy: Data of requester should not be disclosed to other people or systems without the permission of the requester.

● Performance: Measured in terms of throughput and latency.

Definition 1 Trust degree of Web service is the general estimation of availability, integrity, confidentiality, privacy, and performance of Web service. It can be described with the product of these aspects as Eq. (1).

$$t = a \times i \times c \times v \times p \qquad (1)$$

Meanings of theses symbols are listed in the Table 1.

Table 1 Meaning and value range of Symbol

Symbol	Meaning	Range
t	Trust degree	[0, 1]
a	Availability	0 or 1
i	Integrity	0 or 1
c	Confidentiality	0 or 1
v	Privacy	0 or 1
p	Performance	[0, 1]

Definition 1 is used to calculate trust degree of one transaction between requester and Web service. In the experiments presented in Section 6, Eq. (1) will be used to calculate trust degree of web service.

2 Principle of Trust Degree Evaluation with Neural Network

Let there is a trust history sequence A, $A = (a_1, a_2, \cdots, a_n)$, a_i is the trust degree of the ith exchanging between requester and Web service. It can be calculated by Eq. (1). Let there is a complex system S that can produce a trust degree T from A. Let's denote it as $T = S(A)$. In order to evaluate trust degree T' from A with neural network, a neural network (N) must be designed. Let's denote it as $T' = N(A)$. In order to make sure that the output of neural network meets the requirement, formula $\frac{1}{2}(T' - T)^2 < \varepsilon$ should be hold. ε denotes a small positive number as needed.

The principle of trust degree evaluation with neural network is shown in Fig. 1.

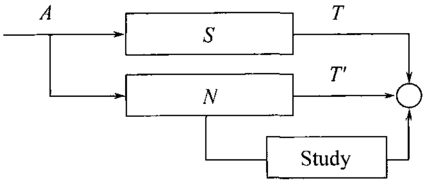

Fig. 1 Principle of trust degree evaluating with neural network

The constructing procedure of N is following:

Step 1 Get a set of input and output of S as training samples;

Step 2 Design a N and train it with the training samples;

Step 3 If the precision of N satisfies the requirement, constructing is end, otherwise do step 2 again.

The neural network that can evaluate trust degree from Web services trust history sequence should be a multi-input and one-output neural network with model identifying ability. BP neural network can be applied for this purpose[11]. The structure of BP neural network is shown as Fig. 2.

In neural network, every node has multiple inputs and one output. The relationship of input and output can be denoted with formula. $y = f(\sum_{i=1}^{n} \omega_i x_i - \theta)$. y is output. f

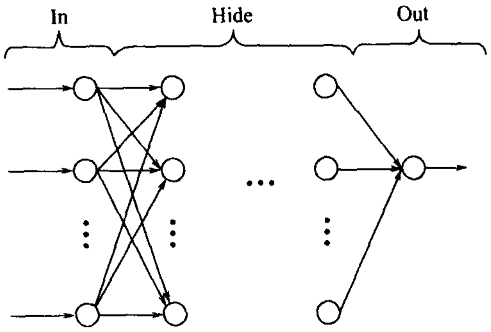

Fig. 2 BP neural network structure

is effect function. θ is threshold. ω_i is the connection weight of the ith input. x_i is the ith input. n is the count of input of the node. The neural network calculates from the in layer to the out layer. The output of out layer is the output of the neural network. f can be selected within threshold function, line function and S style function, e.g. hardlim, purelin, tansig, etc[11]. The number of layers, number of nodes, ω and θ can be selected through experiment.

The number of input k presents the length of trust history sequence which neural network can deal with. In order to balance between precision and efficiency, k can not be too big. Because k is limited and n, length of trust history sequence A, is not limited, A should be standardized.

3 Input Standardization

Definetion 2 Trust history sequence A is a history record of trust degree of Web service which the requester has trade with. It can be denoted with an orderly n tuple, $A = (a_1, a_2, \cdots, a_n)$, a_i is the trust degree of the ith transaction between requester and Web service.

Definetion 3 Active trust history sequence is the recent trust history sequence. It can be denoted with an ordered m tuple. Let $A = (a_1, a_2, \cdots, a_n)$, then $B = (a_{n-m+1}, a_{n-m+2}, \cdots, a_n)$. Let 's note it as $B = \angle(A, m)$

Definetion 4 Constrictive trust history sequence is the subsection average of active trust history sequence. Let $B = (b_1, b_2, \cdots, b_m)$, then constrictive trust history sequence of B is $C = (c_1, c_2, \cdots, c_k)$. Let 's note it as $C = \Diamond(B, k)$. k is the number of input of neural network. Let $s = \frac{m}{k}$,

When $i \times s \leq m$, $c_i = \frac{1}{s}\sum_{j=1}^{s} b_{((i-1)\cdot s+j)}$

When $i \times s > m$, $c_i = \frac{1}{r}\sum_{j=(i-1)\times s}^{m} b_j$, r is arithmetic compliant of m and k.

In order to evaluate trust degree with neural network, requester must record the active trust history sequence B at least.

4 Training Sample Constructing and Neural Network Training

There are two sources to get training sample. One is the real trust history of Web service, and another is the one which is specially designed. The procedure of getting training sample from real Web service is shown as follows:

Step 1 Get trust history sequence A of Web service x;

Step 2 Make active trust history sequences B from A, $B = \angle(A, m)$;

Step 3 Make constrictive trust history sequence C from B, $C = \Diamond(B, k)$;

Step 4 C and T, the given trust degree of Web service x, can make a training sample, (C, T).

The samples obtained from Web service make the result of neural network close to the actual trust degree. The samples specially designed can lead output of the neural network to puniness or encourage some action model.

The procedure of designing special sample is to construct trust history sequence according to k, the number of input of neural network, and to give a trust degree.

5 Trust Degree Evaluation

If there is a trained neural network, the trust degree evaluation is very simple. The procedure is shown as the following:

Step 1 Get the trust history sequence A;

Step 2 Make active trust history sequence B from A, $B = \angle(A, m)$;

Step 3 Make constrictive trust history sequence C from B, $C = \Diamond(B, k)$;

Step 4 Using C as input of neural network, the output of neural network is trust degree.

From the procedure we can see that the complexity of time evaluating trust degree with trained neural net-

HUANG Baohua et al: Trust Degree of Web Services and Its …

work is $O(k)$.

6 Experiments

In order to evaluate the effect of the method proposed in this page, we developed three data storage Web services and a client agent on Microsoft dot Net platform. The first one of the Web services acts on a random trust degree by dropping service request randomly. The second one drops service request periodically with little random. The third one periodically drops service request completely.

Client agent requests these data storage Web services to store data for it separately, and check the data status randomly[5]. Check results are calculated by Eq. (1). The first 300 check results are used to train the neural network. Then the trained neural network was used to evaluate the trust degree of Web service in the next interactions. The evaluating results are presented in Fig. 3.

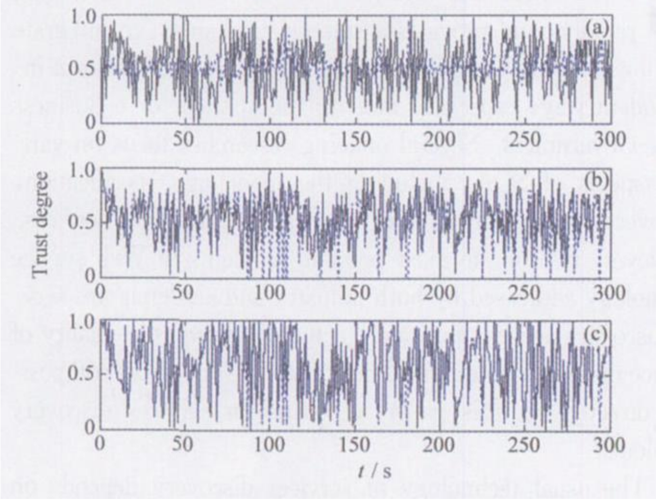

Fig. 3 Result of neural network evaluating

(a) Web service acts on a random trust degree;
(b) Web service acts on a periodically trust degree with little random;
(c) Web service acts on a periodically trust degree;
— The actual trust degree; — the trust degree evaluated with neural network

In Fig. 3 (a), we can see that neural network can hardly evaluate trust degree of Web service acting on a random trust degree. In Fig. 3(b) and Fig. 3(c), we can see that neural network can ideally evaluate the trust degree of Web services acting on a little random trust degree and exactly evaluate trust degree of Web services acting on a periodically trust degree.

7 Conclusion

This paper proposes the concept of trust degree of Web service. The formal definition with availability, integrity, confidentiality, privacy, and performance are given. We apply neural network on Web services trust degree evaluating for the first time. The principle of the approach is described. The suitable neural network structure, neural network constructing, input standardization, training sample constructing, and the procedure of evaluating trust degree of Web services with trained neural network are discussed. Experiments show that neural network can ideally evaluate the trust degree of Web services act on a little random trust degree and exactly evaluate trust degree of Web services act on a periodically trust degree.

References

[1] Booth D, Haas H, McCabe F, *et al*. Web Services Architecture, W3C Working Group Note [EB/OL]. [2005-02-04]. *http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/*.

[2] Emekci F, Sahin O D, Agrawal D, *et al*. A Peer-to-Peer Framework for Web Service Discovery with Ranking [C] // *IEEE International Conference on Web Services*. California: IEEE Computer Society Press, 2004:192-199.

[3] Srivatsa M, Xiong L, Liu L. TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralized Overlay Networks [C] // 14*th World Wide Web Conference*. Chiba: ACM Press, 2005:422-431.

[4] Wang Y, Vassileva J. Trust and Reputation Model in Peer-to-Peer Networks [C] // *Third International Conference on Peer-to-Peer Computing*. Linköping: IEEE Computer Society Press, 2003:150-157.

[5] Huang B H, Hu H P, Lu Z D, *et al*. P2P Data Disaster Tolerance of E-Government [J]. *Computer Science*, 2005, **32** (9A):222-224.

[6] Xu L N. *Neural Network Control* [M]. Beijing: Publish House of Electronics Industry, 2003:60-68 (Ch).

[7] Blaze M, Feigenbaum J, Lacy J. Decentralized Trust Management [C] // 17*th Symposium on Security and Privacy*. Oakland: IEEE Computer Society Press, 1996:164-173.

[8] Blaze M, Feigenbaum J, Ioannidis J, *et al*. The Role of Trust Management in Distributed Systems Security [C] // *Secure Internet Programming, Issues for Mobile and Distributed Objects*. New York: Springer-Verlag, 1999: 183-210.

[9] Chen R, Yeager W. Poblano A Distributed Trust Model for Peer-to-Peer Networks [EB/OL]. [2005-03-01]. *http://www.jxta.org/docs/trust.pdf*.

[10] Pfleeger C P, Pleefger S L. *Security in Computing* [M]. 3rd. New Jersey: Prentice Hall Ptr, 2004.

[11] Hagan M T, Demuth H B, Beale M H. *Neural Network Design* [M]. Boston: PWS Publishing Company, 2002.

□