Electronic Commerce Research and Applications 14 (2015) 532–541

Contents lists available at ScienceDirect

Electronic Commerce Research and Applications

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

A dynamic data driven-based semi-distributed reputation mechanism in unknown networks

Szu-Yin Lin[a,*], Ping-Hsien Chou[b]

[a] Department of Information Management, Chung Yuan Christian University, Taoyuan, Taiwan
[b] Institute of Information Management, National Chiao Tung University, Hsinchu, Taiwan

ARTICLE INFO

Article history:
Received 27 February 2015
Received in revised form 2 July 2015
Accepted 9 August 2015
Available online 28 August 2015

Keywords:
Dynamic data driven application system (DDDAS)
Reputation- and trust-based model (RTM)
Dynamics of trust

ABSTRACT

Trust is a crucial concern related to unknown networks. A mechanism that distinguishes trustworthy and untrustworthy nodes is essential. The effectiveness of the mechanism depends on the accuracy of a node's reputation. The dynamics of trust often occurs in a trusted network and causes intoxication and disguises of the nodes, resulting in abnormal behaviors. This study proposes a semi-distributed reputation mechanism based on a dynamic data-driven application system. This mechanism includes two reputations, local reputation (*LRep*) and global reputation (*GRep*). *LRep* is dynamically and selectively injected into a central controller, and this controller collects the injected data to compute *GRep*, which contains the neural network prediction method, and returns it to provide reference to the distributed nodes. The proposed mechanism focuses on dynamics of trust and the balance between distributed nodes and the central controller. Experimental results showed that *GRep* was computable with only 52.21% (average) *LReps* upload and that *GRep* increased or reduced by 26.5% (average) in a short period, demonstrating that the proposed mechanism effectively handles the problem of dynamics of trust.

1. Introduction

Information security concerns are associated with various types of networks, such as social, peer-to-peer (P2P), e-commerce, and sensor networks. For example, in an e-commerce network, a buyer may encounter fake goods, and in a P2P network, malicious nodes affect normal nodes, resulting in network inefficiency and even meltdown. Therefore, trust relationship in unknown networks is an active research topic. In the past decade, many studies have reported on reputation- and trust-based models (RTMs), and many are currently underway. RTMs distinguish trustworthy and untrustworthy nodes in a network through one or more reputation metrics. In addition, they use different reward and punishment mechanisms to prevent unexpected node behavior. Their main architecture is of two types (Kamvar et al. 2003; Liu and Issarny 2004; Lin 2005), central and distributed architecture. In the central architecture, such as that in some e-commerce websites (e.g., Yahoo!, Auction, and eBay), all transactions and feedback are recorded after each transaction, and these records are referred to during future transactions. The center must have a large storage and high computing ability. Throughput is a concern, particularly in sensor networks. With increasing network traffic, network congestion occurs if all nodes upload all their data to the center. In the distributed architecture, nodes maintain, transfer, exchange, and gather their trust information independently. Finite information is finally used to calculate trust value. A complex algorithm requires high computing ability at each node. For example, mobiles devices are increasingly being used. However, they have limited computing ability and storage. In mobile ad hoc networks, systems using complex RTM exhausts all device resources. Therefore, in the distributed architecture, RTMs cannot be highly complex under limited resource conditions. Conversely, the dynamics of trust (Kanawattanachai and Yoo 2002; Govindan and Mohapatra 2012) causes trust metrics to be nonlinear. For instance, in e-commerce networks, the account of a seller with a high reputation may be stolen by fraudulent buyers because of intoxication, and in P2P networks, a new malicious node pretends to be a good node but becomes malicious after a period, thus displaying an unexpected behavior. Under the dynamics of trust conditions, depending on only historical transaction experiences is insufficient, and time is required to achieve a balance. In this study, this problem is resolved.

This study proposes a semi-distributed reputation mechanism based on a dynamic data-driven application system (DDDAS; Darema 2004) with a semi-distributed architecture. With this mechanism, the system accurately and efficiently distinguishes

[*] Corresponding author. Tel.: +886 3 2655421; fax: +886 3 2655499.
E-mail addresses: szuyinlin@gmail.com (S.-Y. Lin), scott03333@gmail.com (P.-H. Chou).

http://dx.doi.org/10.1016/j.elerap.2015.08.005
1567-4223/© 2015 Elsevier B.V. All rights reserved.

trustworthy and untrustworthy nodes. Furthermore, the system collects dynamic reputation data efficiently, predicts dynamic data, and maintains a resource balance between the central and distributed nodes. The trust mechanism is divided into six components: trust computation, trust propagation, trust aggregation, trust record, trust prediction, and trust application. All components, particularly trust propagation and aggregation, are discussed in the following sections.

By using a DDDAS, reputation data are injected into a central controller (Onolaja 2012). The central controller efficiently gathers only the useful data from all data in the network. It computes new reputation values to be referred to during future transactions. The DDDAS uses a simulation system to predict the next reputation value at the next tick and provides feedback to the physical system. Consequently, predictions relate to the real world. In addition, the semi-distributed architecture balances the utilization of central and distributed resources, rather than relying on only one. In this study, a query-cycle model (QCM; Schlosser 2003) was used to simulate a real-world P2P network to evaluate the availability of this study; in addition, the processing results of the dynamics of trust are presented.

2. Related studies

2.1. Reputation- and trust-based model

Although the concept of trust is encountered every day, trust and reputation have various meanings. Jøsang et al. (2007) used the following examples to illustrate the difference:

(1) “I trust you because of your good reputation.”
(2) “I trust you despite your bad reputation.”

Sentence (1) indicates that trust depends on a trustee’s reputation, which is based on others’ trust. Sentence (2) implies that a trustee may have private knowledge or different standards that override the trustee’s bad reputation originating from others’ trust (Jøsang et al. 2007). Trust is subjective, whereas reputation is relatively objective; that is, reputation is composed of a party’s trust. For example, let trust and reputation metrics range between 0 and 1, and let the reputation value be aggregated on the basis of a party’s trust value (e.g., 0.8) and Node A’s reputation value. The trust thresholds of Nodes B and C are 0.7 and 0.5, respectively. Therefore, Node B trusts Node A because Node A’s trust threshold is higher than the reputation value of Node A, and Node C does not trust Node B because Node B’s trust threshold is lesser than that of Node B.

Transitivity is crucial in RTMs, as detailed in a subsequent section. Briefly, if Node A trusts Node B and Node B trusts Node C, then Node A trusts Node C. In this study, only limited transitivity exists because our trust metric was subjective. Transitivity exists only when the trust threshold is higher than the reputation value. RTMs are implemented in various fields with different metrics. For example, in e-commerce networks, a system aggregate user offers feedback to a metric, and this feedback is applied to the next user assessment before a new transaction starts, and in web service networks, the reputation value depends on stability, transmission speed, and accuracy.

The following RTMs are highly correlated in this study.

2.1.1. EigenTrust

EigenTrust (Kamvar et al. 2003) is an algorithm with a distributed architecture. It aggregates neighbors’ trust to calculate the reputation value that can influence reputation in the next node; that is, Node A requires its neighbors’ help if it does not trade with Node B. In brief, the reputation value is transitive. Let c_{ij} be the reputation value from i to j. The new value is $t_{ik} = \sum_{j=1} c_{ij}c_{jk}$. A high t_{ik} indicates a trustworthy node. Let $\overrightarrow{t_i}$ represent the vector in t_{ik}, and C represent matrix $[c_{ij}]$ such that $\overrightarrow{t_i} = C^T\overrightarrow{c_i}$. If the system demands two-layer neighbors, the power of the equation is two ($\overrightarrow{t_i} = (C^T)^2\overrightarrow{c_i}$), and so on. Computing stops when the reputation value converges.

EigenTrust determines the layer of neighbors which want to ask their trust value. However, computing resources increase with increasing number of layers. Here, the reputation value is equal to the weight of neighbors’ recommendations.

2.1.2. Broker framework

As shown in Fig. 1, a broker framework (Lin 2005) has three components, user, broker, and reputation authority.

Each user represents a network node, such as a sender or receiver in P2P file-sharing networks. Users are connected to brokers responsible for collecting transaction data and maintaining the reputation database. After a transaction, users upload their rating for this transaction. For example, after User A trades with User B, both Users A and B upload their ratings, which may differ. A relevant equation is as follows:

$$R_{new} = e^{-\beta\Delta t}\frac{N}{N+1}R_{old} + (1 - e^{-\beta\Delta t}\frac{N}{N+1})r,$$

where R_{new}, R_{old} , $e^{-\beta\Delta t}$, and r denote the new reputation value, old reputation value, discount factor of R_{old}, and rating uploaded by the user, respectively.

Before a transaction, each user queries the broker on the reputation of the trading user. The broker first searches the reputation of the trading user in its database. If it does not have the related reputation data, it queries another broker or reputation authority. To exclude fake reputation values, legitimate brokers eliminate fake brokers using the following equation:

$X = X + F * (1-X)$, if the recommendation value is consistent with the facts, and

$X = X * (1-F)$, if the recommendation value is inconsistent with the facts,where X is the reputation value of a broker, whose initial value is 0.5, and F is a parameter (0.2).

The higher the X, the more difficulty it is to increase X. However, X can be easily decreased. Therefore, brokers are motivated to maintain their reputation.

Similar to the semi-distributed architecture, users exchange reputation information through brokers. Users only upload their transaction rating and thus the model does not require higher computing resources. The disadvantage is that the model cannot predict future results because it is not a prediction model. Moreover, users have to upload all their ratings, thus resulting in network congestion.

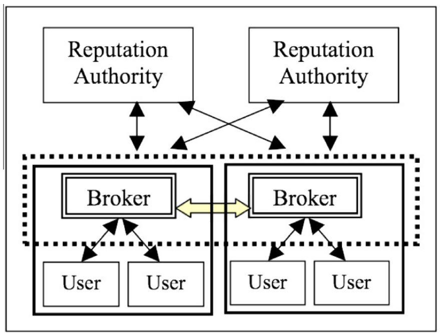

Fig. 1. Broker framework.