An Indirect Security Trust Method Based on Credibility Evaluation

Wei Xianghe[1, 2], Zhang Hong[1], Liu Zhen[1], Li Qianmu[1], Xia Bin[1]
[1]School of Computer Science and Technology
Nanjing University of Science and Engineering
Nanjing 210094, China
[2]School of Computer Science and Technology
Huaiyin Normal University
Huaian 223300, China
kryolith.xiabin@gmail.com

Journal of Digital Information Management

ABSTRACT: In existing trust models, the recommending services often provide a single trust value (computed by them during their interaction with the unknown entity in question) as a recommendation. However, a single trust value recommended by a recommender represents its subjective opinion about the unknown entity and cannot depict the real trust level very well under certain circumstances. To solve this problem, a fuzzy based credibility evaluation method for indirect trust computation is proposed. The calculation method cannot just incorporate a mechanism to determine the weight each valid recommendation should carry on aggregation process, but also distinguish between honest and dishonest recommendations in the meantime.

Subject Categories and Descriptors:
K.6.5 [Security and Protection]; H.5 [Information Interfaces and Presentation] ; Evaluation
General Terms: Trust, Security, Security Evaluation

Keywords: Trust Mechanism, Reputation, Indirect Recommendation, Security Evaluation, Credibility Evaluation, Fuzzy based Evaluation

Received: 11 November 2013, Revised 13 February 2014, Accepted 26 February 2014

1. Introduction

Direct trust can assess the credibility of each node based on the interactive experience with known nodes are part of known nodes. However, when a node is completely unknown, the reliability of it is calculated in accordance with recommendation received from other nodes. Reliance on indirect recommendations may also lead to wrong decisions with a dishonest recommendation. Recommendations can enhance the trust values of suspicious nodes, as well as reduce the trust values of honest nodes, which provide a dishonest recommendation. Thus, it is a fundamental question in the trust model, whether there is a mechanism which can avoid the impact of dishonest recommendations [5].

In the current trust model, recommendation service only provides a single trust value (obtained by the interaction with evaluating nodes). However, such single trust values represent the provider's subjective view of the unknown entity which cannot well reflect the trust level under certain circumstances.For example, S requests recommendation of unknown entity and receives recommendations provided by A and B, the recommended values $T_A = 5$, $T_B = 2.5$. Does this means the same weight should be given to both sides in the complex recommendation trust calculation? Assuming TA is obtained at a time t_A, and TB is obtained at a time t_B. tcurrent - t_A > tcurrent - t_B. Compared with B, A has less interaction history with C. Obviously, not only does B has a recent interaction, but also B has more interactive experience. So when assessing the credibility of C, the recommendation provided by B should be more reliable than that by A. In addition, the credibility of B's recommendation should be strengthened, if the interaction between B and C is more sensible than that between A

and *C*, in the context of the interaction [6].

To solve this problem, this paper presents an efficient method for calculating indirect trust. The premise of the model is that only honest recommendation can be referenced in the recommended trust calculation. Further, when calculating the complex recommendation trust value of an unknown entity, more weight should be given to honest reliable recommendations than unreliable ones. The effectiveness of this model has been proven in the experiment. In the framework of the proposed model, when an end-entity requests a kind of service, the service provider seeks recommendations from another peer service provider. All recommendations must go through the filtering mechanism "*outlier detection engine*". "*Outlier detection engine*" uses deviation detection proposed in [1] to determine whether the recommendation is dishonest, then uses the "*recommended assessment model*" to calculate credibility of every honest recommendation. This model uses the method of weighted average (credibility as weight) to calculate the complex recommendation trust value. The basic function of "*policy analysis*" is to process the request, deciding whether the end-entity that requests the service would be allowed to do request activities, according to the recommendation trust value after calculation and strategies provided for this service.

2. Outliers detection engine

In order to make other service providers provide recommendations, the original service provider *Sq* creates a "*recommendation request message*" (R_{REQST}), and broadcasts the news to the neighbor nodes, trusted nodes and other unknown nodes. Then the original service provider *Sq* waits to receive recommendations. Recommender nodes that have had interaction with the target end-entity may answer R_{REQST} message (R_{RESP}) with the recommendation response message (R_{RESP}). Recommendation providers feedback some attributes (these attributes define some historical interaction information between recommendation providers and assessed terminal) rather than just a single recommendation trust values to *Sq*.

Request and the corresponding message is defined as follows:

$$R_{REQST}\,[\text{Svc\_ID, Entity\_ID, Req\_Time}]$$

$$R_{RESP}\,[\text{Recomm\_ID, Entity\_ID, } Ti, t, n_t, SS]$$

In RREQST, Svc_ID represents the identity of service provider Sq that requests recommendations; Entity_ID represents the identity of target end-entity; Req_Time represents the time of recommendation request. In RRESP, Recomm_ID says the recommender's identity, Ti is the recommended value of recommender i, t is the time recommended value recorded, n_t said the number of interactions that the recommendation is based on, SS is the sensitivity of the recommender.

The purpose of the indirect trust calculation is to determine the trustworthiness of a strange entity from a group of recommendations, thus narrowing the gap between the recommended trustworthiness and actual trustworthiness of the entity. In this framework, the "*outlier detection engine*" is responsible for detecting dishonest recommendations. "*Outlier detection engine*" defines a dishonest recommendation as an abnormal value, which appears inconsistent with other recommendations and does not meet the data distribution of other recommendations. In the field of database and data mining, this method has been fully aware of the importance of detecting outliers [2]. First proposes the method based on the deviation from the abnormal value: positioning the abnormal value by the method of dynamic programming. The model in this paper has extended the qutlier detection technique, so that he can filter dishonest recommendations.This method is based on the following fact: if a recommendation deviates from the median value of the given recommendations set, and the likelihood of this happening is low, then this recommendation is filtered as a dishonest recommendation. If an entity X requests for access to service A, A does not have enough interactive experience with X, then it will broadcast on the recommendation request from X. R is defined as collected recommendations set, wherein n is the total number of recommendations.

Because a smart attacker's recommendation may have little deviation, so as to not easily detected, this model divides all the possible recommended values into ten intervals [3]. For example, Rc_1 consists of recommendations recommended by the values between [0, 0.5]; Rc_2 consists of recommendations recommended by the values between [0.5, 1];Rc_3,..., Rc_{10}; and so on. We think recommendations are similar if they fall within the same intervals. When all the recommendations have been assigned to the respective intervals, filtering engine computes a histogram, which indicates the recommendation number falls within each corresponding interval. H is defines as the histogram of a set of recommendation class [4].

H (R) = {<Rc1, f1>, <Rc2, f 2>, <Rc3, f 3>, <Rc4, f4>, <Rc5, f 5>, <Rc6, f6>, <Rc7, f 7>, <Rc8, f 8>, <Rc9, f 9>, <Rc10, f10>}
Where, *fi* is the frequency of recommendations that fall in the. From this histogram, remove all *fi* = 0 recommendation class, and get a domain set (Rdomain) and a frequency set (*f*).

$$Rdomain = \{\text{Rc1, Rc2, Rc3, ... , Rc10}\}$$

$$f = \{f1, f2, f3, \ldots\ldots, f10\}.$$

Definition 1: Define the dissimilarity function $DF\,(xi)$ as:

$$DF\,(x_i) = \frac{|\,x_i - median\,(x)\,|^2}{f_i} \qquad (1)$$

Where x_i is a recommendation category of the recomme

ndation x. In this method, DF value of x_i is proportional to the absolute median deviation (Median Absolute Deviation, MAD, i.e., $| x_i - median\ (x) |^2$). MAD is used to detect the degree of deviation because of being sensitive to outliers. Abnormal value may significantly change the value of MAD. Further, DF value of x_i and is inversely proportional to its frequency f_i. So if a recommendation value deviates from other recommendation value, and its frequency is very low, then the corresponding dissimilarity DF of the recommendation is a great value. Similarly, if a recommendation value close to other recommendation values, and its frequency is very high, then the corresponding dissimilarity DF of the recommendation is a small value.

For each Rc_i, use the formula (1) to calculate its dissimilarity DF, i.e. the dissimilarity of other recommendation frequency. Put all recommendation classes of Rdomain in reverse order by dissimilarity $DF\ (Rc_i)$. $DF\ (x_i)$ maximum recommendation class is considered most likely to be dishonest recommendation, and it should be filtered out. Once Rdomain has sorted, the next step is to determine a set of dishonest recommendation classes.

Definition 2: Define the SF function of SRdomain as:

$$SF\ (SRdomain_j) = C\ (Rdomain - SRdomain_j) * (DF(Rdomain) - DF\ (Rdomain - SRdomain_j) \quad (2)$$

Where $j = 1, 2, 3, ..., m$; m is the number of elements in $SRdomain$. C is the number of elements in $(Rdomain - SRdomain_j)$. DF(Rdomain) is obtained by adding the DF value of each element in $Rdomain$. SF shows how much the dissimilarity reduces after a group of suspected recommendations ($SRdomain$) have been removed from $Rdomain$.

Definition 3: If $SF\ (SRdomainx\ ) > SF(SRdomainj)$, j is any value of $1, 2, 3\ldots, m-1, m$ except x, then $SRdomainx$ is a group of dishonest recommendations of $SRdomain$.

To find a group of dishonest recommendation domain $Rdomain_{dishonest}$, this paper defines the following mechanisms:

(1) Rc_k is the k-th recommendation class in $Rdomain$; $SRdomain$ is one of the suspect's recommendation class. $SRdomain \subseteq Rdomain$

(2) Initialize $SRdomain$ = { }

(3) For each $SRdomaink$, calculate $SF\ (SRdomain_k)$. $SRdomain_k$ is obtained from merging $SRdomain_{k-1}$ and Rc_k. That is $SRdomain_k = SRdomain_{k-1}$*" Rc_k, where $k = 1, 2, 3\ldots, m-1, m$ is that number of recommendation classes after sorting.

3. Credibility measuring

For each recommendation, there is a certain degree of credibility to determine whether the recommendation is credible. The credibility of a recommended is determined by the following three aspects: (1) Experience (E); (2) Time Base Experience (TBE) (3) Sensitivity of the Service (SS).

TBE representatives the time the recommender last had interaction with the target terminal. In calculating the credibility of recommendations, the introduction of TBE is to integrate constantly decay experience. Change not only means to adapt to new things, but also means that the old things gradually forgotten. Anthropological studies show that as time goes on, new things will gradually slow decay rate. Indirect trust calculation mechanism, based on an anthropological perspective, uses interactive historical data. Therefore, the interactive experience with the target terminal will inevitably affect real-time view of the target terminal, and the newer experience should account for a higher weight in decision-making. To calculate TBE, recommendation service extracts the last interaction time from each RRESP. Make t and t_c respectively represent the last interaction time and the present time, the TBE of target end-entity should be calculated as

$$TBE = \alpha (1 - \beta)^{\frac{\Delta t}{\alpha}} \quad (3)$$

Among them $\Delta t = t_c - t$, α and β are normal adjusted values that can be adjusted through the definition of recession rate. (One day is considered as a period of time).

E represents the total number of interaction the service provider has with a terminal entity in his life cycle. It measures the number of activities between the end-entity and the service provider. The more interactions a service provider has with the target end-entity, the more experience he gets. Thus, the more interactions with the destination terminal number are, the more credible the recommendation is. In this method, the recommendation service extracts interaction number n_t from RRESP. Because credibility is the value within [0,1], we need a format function that is used to limit the number of interaction between [0,1]. The function we used to format the interactive value is as follows:

$$E = \frac{n_t - n_t^{min}}{n_t^{max} - n_t^{min}} \quad (4)$$

Wherein n_t^{min} and n_t^{max} especively represent the minimum and the maximum number of interactions. $n_t^{min} = 1$, $1 \le n_t^{max} \le \infty$.

SS: Ubiquitous environment is conceived as a physical space wealthy of equipment and service and it is able to carry out interaction service with the user, the physical environment and the external network. We can classify these services depending on the type of various service provided by ubiquitous environments. For example, a

simple scanning device has a lower sensitivity than the file service. More sensitive recommender has higher weight than less sensitive ones. The hidden meaning of idea is that a malicious service requester can have large number of interactions with low sensitive service to enhance his trust value. However, malicious intent can be effectively detected after using recommendation system that combines experience and sensitivity.

4. Credibility evaluation based on fuzzy reasoning

Because of the ambiguity, ambiguity and subjectivity of the measure of credibility, so it could not be measured as a discrete value. This method introduces a fuzzy inference engine for assessing the credibility of recommendations. Fuzzy reasoning is using fuzzy logic to develop a given input to output mapping process. This mapping provides a basis for decision or pattern recognition. In this method, E, TBE and SS are input as fuzzy inference engine. Output of fuzzy inference engine is the credibility assessment. Fuzzy reasoning process has three main steps: fuzzification, rule evaluation, and defuzzification, as shown in Figure 1.

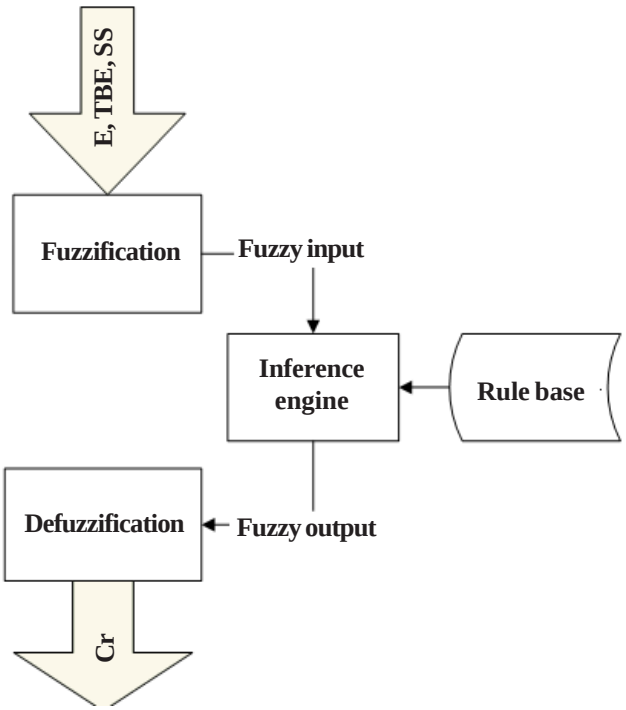

Figure 1. *Cr* fuzzy reasoning process

(1) **Fuzzy:** using fuzzy control to convert the clear input to the fuzzy language value of corresponding membership function. Put E, TBE, SS into a vector, as $X^* = [x1^* \; x2^* \; x3^*] = [E \; TBE \; SS]$.Wherein, X^* represents a real clear input. Each input variables are divided into three categories of fuzzy sets: *Low* (*L*), *Medium* (*M*), *High* (*H*). X between each fuzzy set is characterized by a membership function (MF) and an associated x between $[0,1]$. In the model, we define the input Gaussian membership function MF: $\mu_{TBE}{}^{d}(x_1)$, $\mu_{E}{}^{d}(x_1)$ and $\mu_{SS}{}^{d}(x_1)$.

$$\mu_{TBE}{}^{d}(x_1) = e^{-\left(\frac{x_1 - \bar{x}_1^{d}}{\sigma_1^{d}}\right)^2} \quad (5)$$

$$\mu_{E}{}^{d}(x_2) = e^{-\left(\frac{x_2 - \bar{x}_2^{d}}{\sigma_2^{d}}\right)^2} \quad (6)$$

$$\mu_{SS}{}^{d}(x_3) = e^{-\left(\frac{x_3 - \bar{x}_3^{d}}{\sigma_3^{d}}\right)^2} \quad (7)$$

Where, $d = \{L, M, H\}$. $x_1{}^{d}, x_2{}^{d}, x_3{}^{d}$ and $\sigma_1{}^{d}, \sigma_2{}^{d}, \sigma_3{}^{d}$ are constant parameters, representing the mean and variance of input fuzzy set. In the model, using a Gaussian blur mapping $x^* \in R3$ to fuzzy set X.

$$\mu_X(x_1, x_2, x_3) = e^{-\left(\frac{x_1 - x_1^*}{a_1}\right)^2} \times e^{-\left(\frac{x_2 - x_2^*}{a_2}\right)^2} \times e^{-\left(\frac{x_3 - x_3^*}{a_3}\right)^2} \quad (8)$$

Where $a1, a2, a3$ are positive integer, and using $a_1 = 2 \max_{d=\{L,M,H\}} \sigma_1^{d=2}$, $a_2 = 2 \max_{d=\{L,M,H\}} \sigma_2^{d=2}$, $a_3 = 2 \max_{d=\{L,M,H\}} \sigma_2^{d}$, t-norm"$\times$" operation is the algebraic product.

Output variables consist of five fuzzy sets (Cr^g): VL (Very Low), L, M, H and VH (Very High). The Gaussian MF for five fuzzy sets is defined as

$$\mu_{Cr^g}(y_m) = e^{-\left(\frac{y_m - y^{-g}}{\rho^g}\right)^2} \quad (9)$$

Where $g = \{VL, L, M, H, VH\}$, y^{-g} and ρ^g are respectively the mean and variance of the output fuzzy set.

(2) **Rule evaluation:** Fuzzy reasoning is estimate about fuzzy relations, which is based on logic rules in the fuzzy rule. In this method, we choose to use the Product Inference Engine (PIE) to deal with the fuzzy output. Fuzzy rules library that is used to assess the credibility of recommendations is based on the following axioms:

Axiom 1: For highly sensitive services (SS), i.e., a large number of interactions with the target end-entity (E) and most of them are recently interaction (TBE), Cr is a high reliability.

Axiom 2: For low-sensitive services (SS), i.e., a small number of interactions with the target terminal (E) and most of them are over a long time (TBE), Cr is a low reliability.

Based on these axioms, this paper presents 27 kinds of fuzzy inference rules, based on three fuzzy input functions and a fuzzy output function, to measure credibility, and these 27 rules are in the following table 1. This paper uses PIE, as the formula (10) defines, based on fuzzy rules and linguistic to process fuzzy input. PIE structure, based on the individual rules of inference, consists of the Mamdani method, t-norm and the maximum operation [25] of s-norm. PIE is defined as

$$\mu_{Cr}(y_m) = \max_{i=1}^{M}\Big[\sup_{\{x_1,x_2,x_3\}} \mu_X(x_1, x_2, x_3)\,\mu_{TBE^l}(x_1)\,\mu_E(x_2)\,\mu_{SS^l}(x_3)\,\mu_{Cr^l}(y_m)\Big] \quad (10)$$

Wherein for a given model, the fuzzy rule is represented by l. $l = 1, 2, \ldots, 27$. Through decomposition and simplification, the above equation simplifies to:

$$\mu_{Cr'}(y_m) = \max_{l=1}^{M}\Big[\Big[e^{-\left(\frac{x_{1P}^l - x_1^{-l}}{\sigma_1^l}\right)^2}.e^{-\left(\frac{x_{1P}^l - x_1^{-l}}{a_1}\right)^2} \times e^{-\left(\frac{x_{1P}^l - x_1^{-l}}{\sigma_2^l}\right)^2}.e^{-\left(\frac{x_{1P}^l - x_1^{-l}}{a_2}\right)^2} \times e^{-\left(\frac{x_{3P}^l - x_3^{-l}}{\sigma_3^l}\right)^2}.e^{-\left(\frac{x_{3P}^l - x_3^{*}}{a_3}\right)^2}\Big]\mu_{Cr^l}(y_m)\Big] \quad (11)$$

Wherein

$$x_{1P}^l = \frac{a_1^2\, x_1^{-l} + (\sigma_1^l)^2 x_1^*}{a_1^2 + (\sigma_1^l)^2},\ x_{2P}^l = \frac{a_2^2\, x_2^{-l} + (\sigma_2^l)^2 x_2^*}{a_2^2 + (\sigma_2^l)^2},$$

$$x_{3P}^l = \frac{a_3^2\, x_3^{-l} + (\sigma_3^l)^2 x_3^*}{a_3^2 + (\sigma_3^l)^2},$$

(3) Defuzzification: This is the process of converting fuzzy output to exact output. Use the blur eliminator, a map output PIE to discrete points y^*, y^* represents the best point. Blur elimination by the method of gravity center specified y^* by collating the area covered by the value of Cr. But because of the complexity of the calculations, bur elimination by the method of center average, proposed in the literature, can calculate the approximation of y^*. The formula is as follows:

$$Cr = y^* = \frac{\sum_{g=1}^{5} \overline{y^g}\, w^g}{\sum_{g=1}^{5} w^g} \quad (12)$$

Wherein y^g and g^{th} respectively represent the center of either fuzzy sets, and w^g is its height.

After calculating the credibility of each recommendation, the service provider Sq uses the credibility and trust values to aggregate compute the recommendation trust value (T_{recom}) of the end-entity. As the weight the heavy trust value, each recommended credibility calculated indicates the influence degree of each recommendation in the polymerization process. Therefore, the higher the recommended reliability is, the greater weight it holds in the polymerization. The service provider uses each recommended credibility and recommendation to calculate T_{recom}, with the following formula:

Rule name	Rule description
Ru(1)	IF E is L AND TBE is L AND S is L THEN Cr is VL
Ru(2)	IF E is M AND TBE is L AND S is L THEN Cr is L
Ru(3)	IF E is L AND TBE is M AND S is L THEN Cr is L
Ru(4)	IF E is L AND TBE is L AND S is M THEN Cr is L
Ru(5)	IF E is M AND TBE is M AND S is L THEN Cr is L
Ru(6)	IF E is L AND TBE is M AND S is M THEN Cr is L
Ru(7)	IF E is M AND TBE is M AND S is L THEN Cr is L
Ru(8)	IF E is H AND TBE is L AND S is L THEN Cr is L
Ru(9)	IF E is L AND TBE is H AND S is L THEN Cr is L
Ru(10)	IF E is L AND TBE is L AND S is H THEN Cr is M
Ru(11)	IF E is L AND TBE is M AND S is H THEN Cr is M
Ru(12)	IF E is L AND TBE is H AND S is M THEN Cr is M
Ru(13)	IF E is M AND TBE is L AND S is H THEN Cr is M
Ru(14)	IF E is M AND TBE is H AND S is L THEN Cr is M
Ru(15)	IF E is H AND TBE is L AND S is M THEN Cr is M
Ru(16)	IF E is H AND TBE is M AND S is L THEN Cr is M
Ru(17)	IF E is M AND TBE is M AND S is M THEN Cr is H
Ru(18)	IF E is H AND TBE is H AND S is L THEN Cr is H
Ru(19)	IF E is L AND TBE is H AND S is H THEN Cr is H
Ru(20)	IF E is H AND TBE is L AND S is H THEN Cr is H
Ru(21)	IF E is H AND TBE is M AND S is M THEN Cr is H
Ru(22)	IF E is M AND TBE is M AND S is H THEN Cr is H
Ru(23)	IF E is M AND TBE is H AND S is M THEN Cr is H
Ru(24)	IF E is M AND TBE is H AND S is H THEN Cr is H
Ru(25)	IF E is H AND TBE is H AND S is M THEN Cr is H
Ru(26)	IF E is H AND TBE is M AND S is H THEN Cr is H
Ru(27)	IF E is H AND TBE is H AND S is H THEN Cr is VH

Table 1. Fuzzy inference rules of Cr

$$T_{recom} = \begin{cases} undefined, & if\,(i = 0) \\ 0, & if\,(i > 0 \text{ and } \sum_{i}^{n} Cr_i = 0) \\ \dfrac{\sum_{i=1}^{n} Cr_i * T_i}{\sum_{i=1}^{n} Cr_i} & if\,(i > 0 \text{ and } \sum_{i=1}^{n} Cr_i = 0) \end{cases} \quad (13)$$

Where I represents the number of recommended. According to formula (13), if the recommendation requester did not receive any recommendation information which equals to $i = 0$ then $T_{recom} = 0$. This situation is likely to occur. For example, a service requestor is a new network node in the network, and not with any other service providers have interaction. Not only that, if the recommender's SS is very low, and only long ago has several interactive with entity. Then the credibility of recommendation provided by recommender may be 0.

Experiments and Analysis.

5. Experiment and Analysis

In order to explain the calculation process of establishing trust recommended value between the neighbor nodes (strangers' nodes), this part gives an example to explain. Suppose A is a service provider on the Internet, A uses this model to detect the dishonest recommendation and calculate an unknown entity recommendation trust value.

For the application of this model, some parameters to configure the model needs first. In the calculation in the recommendation credibility, the parameters which models need to use are in table 2.

Parameters	Value
Present Time tc	"30 Nov 2012, 10:20:30:45"
The minimum number of interaction nmin	1
The maximum number of interaction nmax	50
α	1.1
β	0.1

Table 2. Indirect trust initial parameters

In this scenario, assuming for the service provider X, A is an unknown node, and X wants to ask A for service. Because A and X no interaction before, in order to determine the X trusted values and access level, A will be prior to those with X had the same terminal interactive request recommended. Table 3 shows that A sends RREQST information and A receives feedback from the same level's terminal. The information will be handled by model. Recommendation trust value computation steps are as follows:

Through the "*anomaly detection engine*", we can know that, R2, R5, R8 were found to be dishonest recommendations.

When receiving a recommendation, it is passed to the qutlier detection engine. Outlier filtering engine uses a detection mechanism based on deviation to filter dishonest recommendations. Experiments and Analysis.

Step 1: After receiving a recommendation from the peer-to-peer service provider, the recommendations are grouped into different intervals according to the recommendation value Ti. Table 4 shows the RDomain formed after received recommendations have been assigned to different intervals.

Step 2: After dividing received recommendations into different categories, remove recommendation whose value is 0 and calculate the $DF(Rc_i)$of each recommendation.

Recommendation request information:	
RREQST [SvcI D : 1023, Enti tyI D : 8745, ReqTime : "Nov302013, 10 : 20 : 30 : 45"]	
Corresponding recommendation information:	
R1	RRESP[RecommI D : 1067, Enti tyI D : 8745, 1.15, "Nov 20 2013, 10 : 20 : 30 : 47", 30, 0.11]
R2	RRESP[RecommI D : 1238, Enti tyI D : 8745, 35, "Nov25 2013, 12 : 45 : 21 : 41", 45, 0.7]
R3	RRESP[RecommI D : 2154, Enti tyI D : 8745, 0.95, "Nov 19 2013, 08 : 16 : 38 : 15", 17, 0.6]
R4	RRESP[RecommI D : 3152, Enti tyI D : 8745, 1.4, "Oct 30 2013, 18 : 40 : 39 : 13", 37, 0.8]
R5	RRESP[RecommI D : 1863, Enti tyI D : 8745, 9, "Nov 17 2013, 13 : 29 : 12 : 55", 45, 0.9]
R6	RRESP[RecommI D : 2169, Enti tyI D : 8745, 0.55, "Nov 27 2013, 21 : 58 : 30 : 27", 45, 0.9]
R7	RRESP[RecommI D : 2041, Enti tyI D : 8745, 0.45, "Nov 06 2013, 11 : 34 : 04 : 42", 21, 0.2]
R8	RRESP[RecommI D : 1009, Enti tyI D : 8745, 4.75, "Oct 20 2013, 15 : 20 : 59 : 08", 36, 0.4]
R9	RRESP[RecommI D : 7130, Enti tyI D : 8745,1.3, "Nov 26 2013, 06 : 33 : 09 : 45", 29, 0.5]
R10	RRESP[RecommI D : 5308, Enti tyI D : 8745, 0.7, "Nov 09 2013, 19 : 45 : 01 : 31", 31, 0.7]

Table 3. Recommend and recommended the corresponding instance

Table 5 shows the recommended class in reverse order sorted according to Dissimilitude Degreedissimilitude degree.

Rci	Rec Value rci	Frequency fi
Rc1	0.5	1
Rc2	1	3
Rc3	1.5	3
Rc4	2	0
Rc5	2.5	0
Rc6	3	0
Rc7	5	1
Rc8	4	1
Rc9	4.5	0
Rc10	4	1

Table 4. Recommendation partition

Step 3: Calculation of each RDomain's SF are derived from the sorted RDomain. Because the Rc10's deviation value is the maximum, it is the most suspicious recommendation. Add it to the suspicious recommendati on domain (SRDomain1), then calculate the SF value of

SRDomain1. Then, combined with SRDomain1 and a recommended Rc8 to form SRDomain2, and then calculate the SF. For each Rci in the Rdomain ($i < m$, m is the number of recommendation classes in Rdomain), the process is repeated. As shown in Table 6, SRdomain3 has the largest SF value. So the SRdomain3 recommendation {5, 4, 5} is considered to be a dishonest recommendation, and should be removed from the Rdomain.

Rci	Rec Value rci	Frequency fi	DF(Rci)
Rc10	5	1	12.25
Rc8	4	1	6.25
Rc7	5	1	4
Rc1	0.5	1	1
Rc2	1	3	0.083
Rc3	1.5	3	0

Table 5. Recommendation class order

SRdomain	Rdomain-SRdomain	DF(Rdomain-SRdomain)	SF
{5}	{4,5,0.5,1,1.5}	11.333	110.25
{5,4}	{5,0.5,1,1.5}	6.583	136
{5,4,5}	{0.5,1,1.5}	0.333	162.75
{5,4,5,0.5}	{1,1.5}	0.083	141
{5,4,5,0.5,1}	{1.5}	0	70.749

Table 6. The calculation results of SF

5.1 Calculation of Credibility

If R_v represents the recommendation set after removing dishonest recommendations. The steps below explain the process of calculating the recommendation trust value by the "*recommendation evaluation engine*".

Step 1: Because the credibility and the recommendation trust value are defined in the [0, 1], we need to define the number and time of interaction between [0, 1]. For each effective recommendation in the R_v, we use type (3), (4) to calculate TBE, E. Table 7 shows the E, TBE, SS after calculating the recommendations in R_v.

Step 2: For each honest recommendation, calculate its credibility by the use of fuzzy logic to the E, TBE, SS to. First we use three inputs (E, TBE, SS), followed by type (5), (6), (7) to determine the membership degree of their own through the Gauss MF. These inputs are thought to be limited to numbers between [0, 1].

Step 3: When the input is fuzzy, we can know membership degree of each input to meet the demand. There are a total of 27 rules in this model. As shown in Table 2, each of the rules are composed of three inputs, and each input is divided into L, M, H. Each rule uses logic operation AND (= prod) to calculate the fuzzy input. The calculation of R1 for ru1 is as follows:

Ri	nt	ti	E	TBE	SS
R1	30	Nov 20 2012, 10 : 20 : 30 : 45	0.6	0.422	0.11
R3	17	Nov 19 2012, 08 : 16 : 38 : 15	0.34	0.383	0.6
R4	37	Oct 30 2012, 18 : 40 : 39 : 13	0.74	0.062	0.8
R6	45	Nov 27 2012, 21 : 58 : 30 : 27	0.9	0.825	0.9
R7	21	Nov 06 2012, 11 : 34 : 04 : 42	0.42	0.11	0.2
R9	29	Nov 26 2012, 06 : 33 : 09 : 45	0.58	0.749	0.5
R10	31	Nov 09 2012, 19 : 45 : 01 : 31	0.62	0.147	0.7

Table 7. The E, TBE, after recommendation calculation in R_v

$$Ru^1 = prod\{\mu_{E^L}, \mu_{TBE^L}, \mu_{SS^L}\} = prod\{0.226, 0.482, 0.951\} = 0.1036$$

Step 4: Because the model uses the Gauss fuzzifier, according to the formula (11), we can get:

$$\mu_{Cr'}(y_m) = \max_{i=1}^{M}\Big[0.1036\mu_{Cr^{VL}}(y_m),\ 0.439\ \mu_{Cr^L}(y_m),\ 0.208\mu_{Cr^L}(y_m),\ldots,\ 0.008\mu_{Cr^M}(y_m), 0.028\mu_{Cr^M}(y_m),\ldots,\ 0.122\mu_{Cr^H}(y_m), 0.002\mu_{Cr^H}(y_m),\ldots,\ 0.005\mu_{Cr^{VH}}(y_m)\Big]$$

Step 5: After rules in the formula grouped according to fuzzy output set (VL, L, M, H, VH), the maximum value can be chosen to represents the group.

$$\mu_{Cr'}(y_m) = \Big[0.1036\mu_{Cr^{VL}}(y_m),\ 0.884\ \mu_{Cr^L}(y_m),\ 0.309\mu_{Cr^M}(y_m),\ldots, 0.256\mu_{Cr^H}(y_m), 0.0048\mu_{Cr^{VH}}(y_m)$$

Step 5: The output of the PIE is an aggregated fuzzy output set containing a group of output values. So in order to obtain a unique value, we must carry on the defuzzification. Such as using formula (12) to calculate the reliability of R1:

$$Cr = \frac{0.103*0 + 0.884*0.25 + 0.309*0.5 + 0.256*0.75 + 0.0048*1}{0.103 + 0.884 + 0.309 + 0.256 + 0.0048} = 0.367$$

Each recommendation in the R_v are repeated using the recommendation evaluation process to calculate their own credibility. The credibility is the weight of each recommendation in the trust value assessment. Table 8 shows the credibility of each recommendation in R_v.

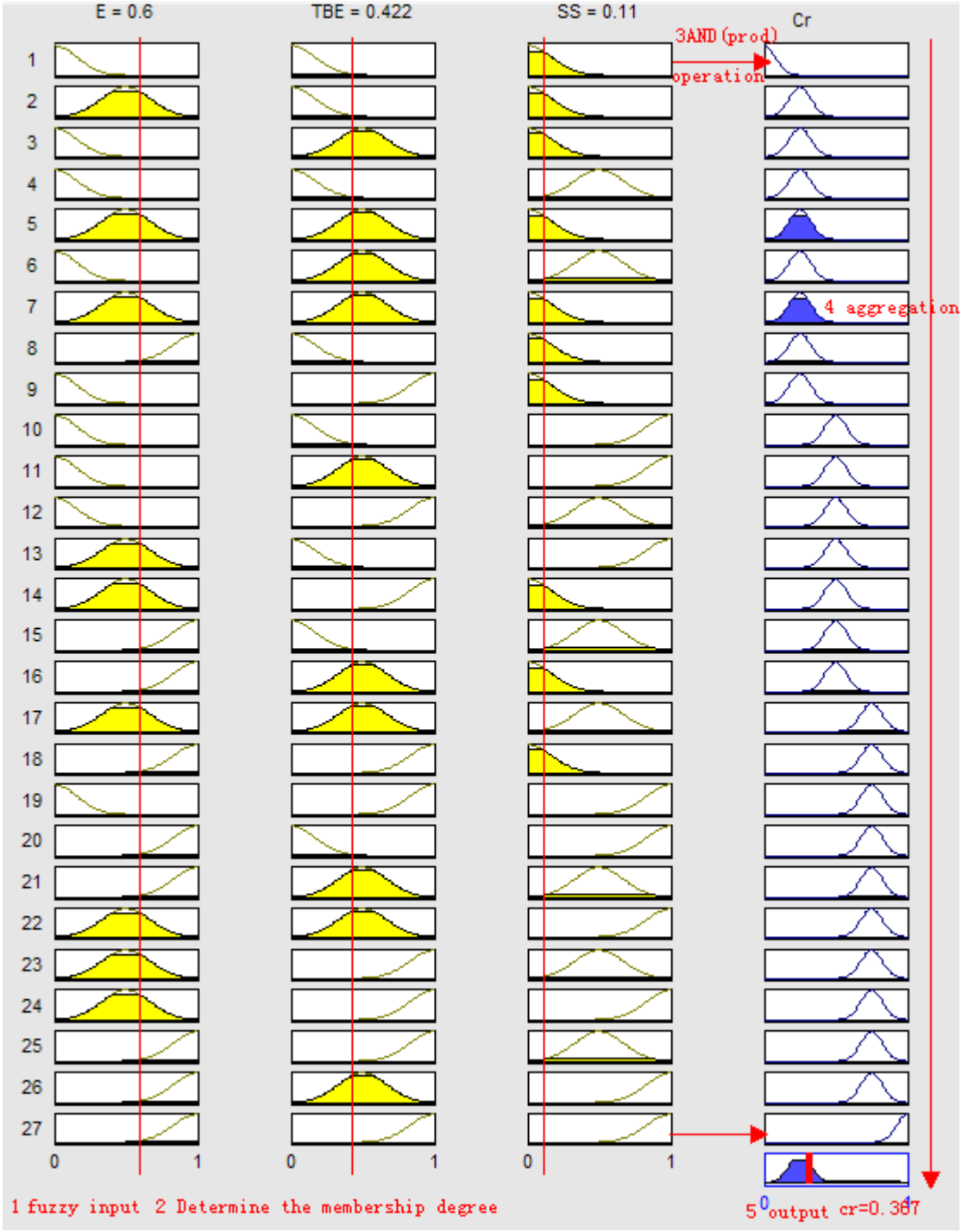

Figure 2. The calculation process in R1 Matlab

Ri	E	TBE	SS	Cr
R1	0.6	0.422	0.11	0.367
R3	0.34	0.383	0.6	0.475
R4	0.74	0.062	0.8	0.54
R6	0.9	0.825	0.9	0.805
R7	0.42	0.11	0.2	0.291
R9	0.58	0.749	0.5	0.613
R10	0.62	0.147	0.7	0.448

Table 8. Each R_v recommendation credibility

5.2 Calculation of recommendation credibility

Integrate each recommendation value (Ti) and each recommendation credibility in the collection R_v, according to equation (13), the recommendation trust value of the entity Trecom is calculated as:

$$T_{recom} = \frac{0.367 * 1.15 + 0.475 * 0.95 + 0.54 * 1.4 + 0.805 * 0.55 + 0.291 * 0.45 + 0.613 * 1.3 + 0.448 * 0.7}{0.367 + 0.475 + 0.54 + 0.805 + 0.291 + 0.613 + 0.448}$$

$$= 0.935$$

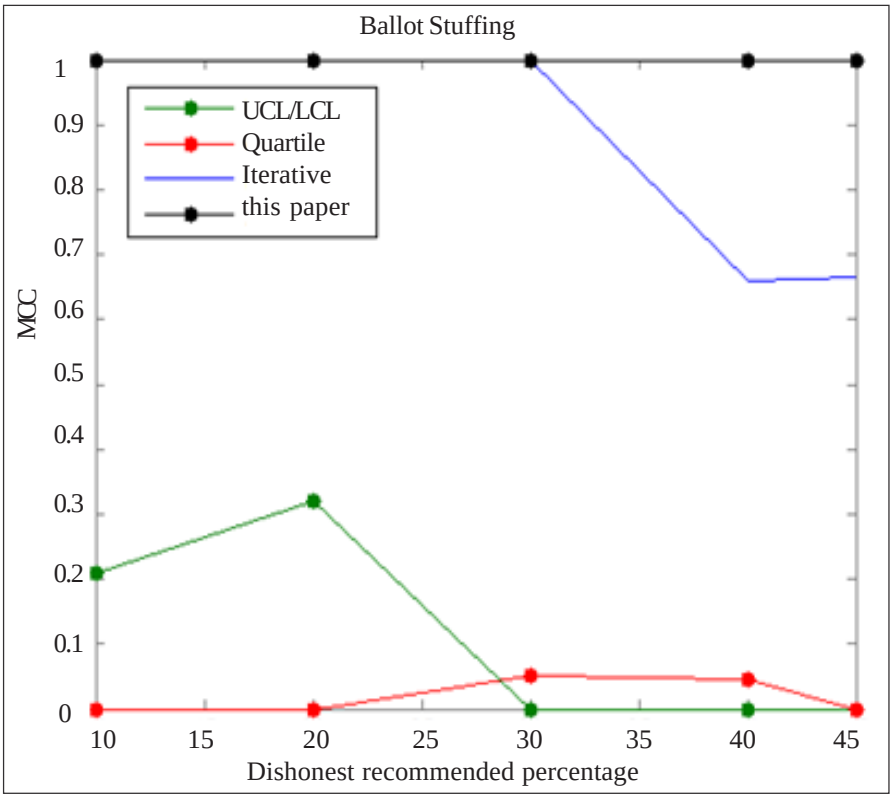

Figure 3. Effect of bad mouthing on the anomaly detection engine

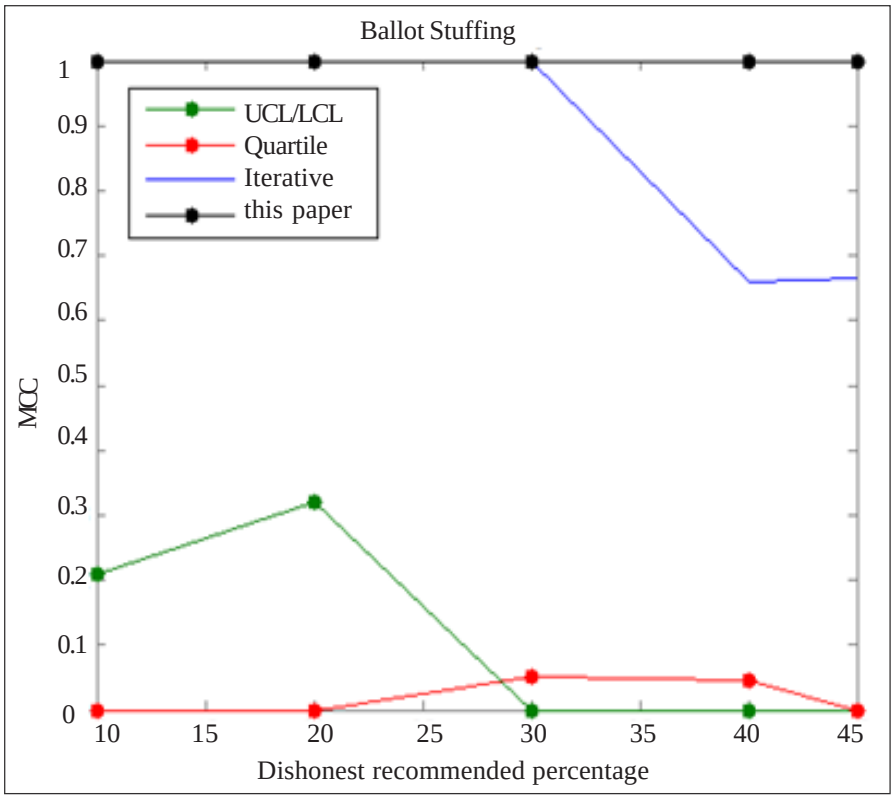

Figure 4. Effect of Ballot Stuffing on the anomaly detection engine

5.1 Verification of abnormal value detection engine

In order to verify the affectivity of abnormal value detection engine, this paper compares the Quartile [33], Control Limit Chart [31], Iterative Filtering [34] and other detection methods1 and this method to verify their performance in dishonest recommendation detection. This paper applies these methods to detect dishonest recommendation in two different scenarios. In the first experiment, assuming that there is a certain proportion of dishonest recommendation. They give the recommended value is between 0.5-1.5, they use bad mouthing attack, the experimental results in figure 2. The second set of

experiments, the recommender hypothesis is not honest to highly recommended value (4-5), they use Ballot Stuffing attack, the experimental results in Figure 3. In the two experiments, dishonest recommenders' ratio is between 10%-45%. In order to compare, this paper uses the Mathews correlation coefficient (Mathews Correlation Coefficient (MCC)) [35] measurement for the detection of four dishonest recommendations. The calculating formula of type MCC:

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Where TP is the number of really high reputation values, TN is the number of really low reputation value, FP is the number of false high reputation values, FN is the number of false low reputation value. MCC returns the -1 to a value of 1, 1 represents perfect filter, 0 representatives of stochastic filtering, -1 representatives were not filtered. In order to avoid infinite results in the calculation of MCC, so that if the denominator of four TP, FP, TN and FN and has a value of zero, the denominator is set to 1. When the dishonest recommenders increased from 10% to 45% changes, compared four methods of MCC as shown in Figure 3 and figure 4. According to the results, the method can effectively detect the dishonest recommendation.

Type	Dishonest & detection	Evaluation of recommendation
Average ([26] [27])	NA	NA
Reputation based model ([28] [29] [30])	NA	The reputation of recommender
Malicious recommenda-tion filtering ([31]-[34])	Control Limit Charts,Quartile, Iterative Filtering	NA
This method	The filtering mechanism based on deviation	The credibility of recommender

Table 9. The classification of recommended method

5.2 Verification of recommendation value

To demonstrate the validity of the methods, this part simulates the experiment. The simulation is to verify whether the target entity credibility method to determine the effective computing trust value based on fuzzy inference. Because of the need to compare, methods are divided into four types as shown in table 9. In the experiment, we simulate the multi terminal environment, each terminal to provide or request recommended and

Simulation	Identifier	scenario 1	Scenario 2
The simulation run times	N_STEPS	7	7
The number of recommender's RT	N_RT	500	500
The percentage of honest recommender RT	N_honest_RT(%)	90	90
Honest recommendation value range	Honest_range	[3,5]	[0,1.5]
The percentage of dishonest recommendation RT	N_Dishonest_RT(%)	10	10
The percentage of dishonest recommendation RT	Dishonest_range	[0,1.5]	[3,5]
Dishonest each simulation RT percentage increase	N_Dishonest_RT_inc	5	5
The evaluation of RPT real trust value	TRPT	4	0.75
Each collection of recommended number VH	Nrecomm	100	100
The percentage of VH recommendation H	CR_VH	20	20
The percentage of H recommendation H	CR_H	20	20
The percentage of M recommendation H	CR_M	20	20
The percentage of L recommendation H	CR_L	20	20
The percentage of VL recommendation H	CR_VL	20	20

Table 9. The classification of recommended method

continuous leave or join in the network environment. Accordingly, the terminal can be divided into two categories: recommendation requester (Recommendation Requestor Terminal, RRT) and recommended provider (Recommendation Provider Terminal, RPT). Each test, the new RRT requests to other RPT network recommendation to the unknown RPT, repeated the process. All RPT are likely to become a presenter terminal of other RPT (Recommendation Terminal, RT).

When the RRT send request to RPT, RT will take on recommendation. Recommendation values are between [0, 5]. According to the recommendation credibility to judge whether the RT is honest. Honest RA according to his own experience true provide recommended, however, dishonest RT according to his malicious intent to disguise his experience, provide a high, low or unstable recommended. Assume that each of RRT and RT in

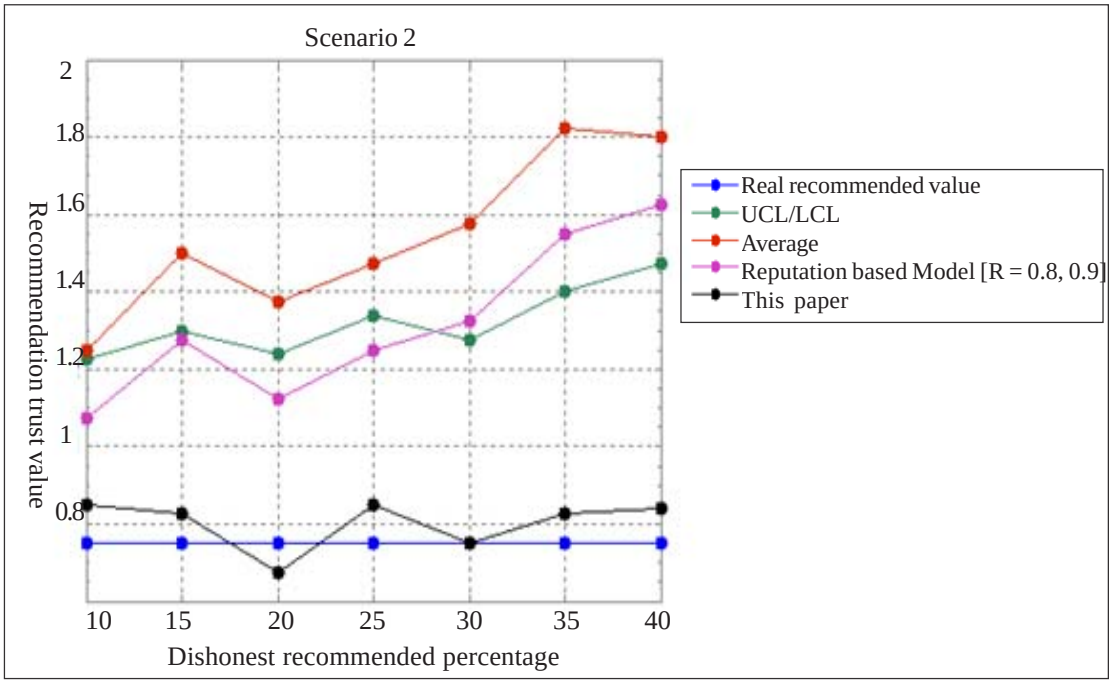

Figure 5. Comparison chart of recommended trust value of scenario 1

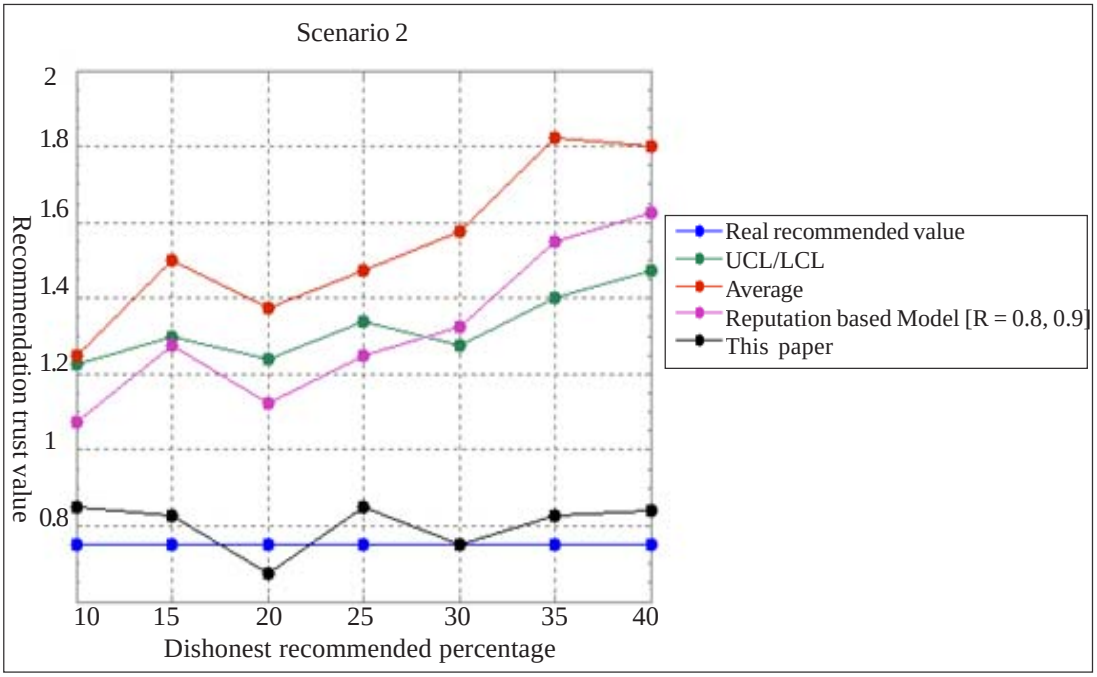

Figure 6 Comparison chart of recommended trust value of scenario 2

RREAST and RRESP format to exchange recommended. In order to define the recommendation credibility, random for each recommended distribution type: V L [0 1], L [1 2], M [2 3], H [3 4] and V H [4 5].

Test define a group of honest recommendation N_honest_RT and a group of dishonest recommendation N_dishonest_RT. The simulation is divided into N_ST E PS steps, each step, the percentage of dishonest_RT increased gradually. Each step, RRT radio and RREAST, then randomly select a group of RT. After each simulation run, recommendation trust RRT to table four methods and 9 respectively in the calculation of the value of SRA.

Validity of the method proposed in this paper for the analysis of the simulation environment, the definition, and this paper introduced two attack scenarios for recommendation method (bad mouthing and Ballot Stuffing).

5.2.1 Attack scenario 1

The attack scenario is dishonest recommender use bad mouthing attack on RRT, in order to reduce the reliability of RPT. The scene simulation variables as shown in table 10. In this scenario, a RRT collection RA recommendation about an unknown RPT. The real trust hypothesis RPT value is 4. Simulation of the initial, 10% in the environment

are not honest RA, by giving them the recommendation trust low value (between [0, 1.5]), bad mouthing attack on RPT. In the seven step of each step, the percentage of RA increased 5% dishonest. Table 9 Comparison of recommend trust through the categories in the calculated values as shown in figure 5.

5.2.2 Attack scenario 2

The attack scenario is recommended dishonest Ballot Stuffing attack on RRT, in order to improve the reliability of RPT. The scene simulation variables as shown in table 10. In this scenario, the simulation is divided into seven step operation. Each step, a dishonest RA recommended by the high trust value (between [3, 5]) way launched Ballot Stuffing attack. The actual trust assumption RPT the value 0.75. Table 9 Comparison of recommend trust through the categories in the calculated values as shown in figure 6.

Two attack scenarios show that the method proposed in this paper to deal with some attack is more effective than the other methods. The recommendation method based on average value, the received recommended, do the calculated average value, get the Trecom, and then, with the increase of dishonest recommendation, Trecom also tend to be dishonest recommendation. In Figure 5, you can see, the effect of bad mouthing attack, recommendation trust value calculated trust value smaller than the actual. The recommendation method based on credibility, the credibility of precision affects the reliability and credibility of the recommendation. But in Figure 5, 6 shows, this method cannot accurately judge the entity credibility, because he cannot detect the entity of bifurcation behavior has a very high reputation. But this method can use technology to filter the dishonest recommendation, to obtain accurate results. Because, an honest recommendation, he may be obtained according to the long ago or inadequate experience, so these methods cannot be measured by an honest recommendation reliability. The proposed method can detect the dishonest recommendation, and consider the measure of each honest recommendation trust, recommendation trust so as to calculate the optimal value.

In summary, indirect trust recommendation based on computing plays a very important role in the comprehensive trust method. Calculation method for the indirect trust can not only distinguish dishonest recommendation, but also can calculate each credible honest recommendation. The method of the recommendation credibility measure concept can determine the influence degree of each recommendation. Credibility is measured by introducing the recommended three new parameters to calculate the corresponding information, these three parameters are E, TBE, and SS. Reliability is calculated by fuzzy inference engine completed.

6. Conclusion

This paper presents a method to calculate the indirect trust based on fuzzy inference. Due to the single trust recommendation to provide value represents the subjective view of his unknown entity, cannot well reflect the trust level in certain circumstances. The method can solve this problem. In the method, for each recommendation, have a certain degree of confidence, to decide whether the recommendation is credible. The recommendation credibility by interactive number, the time of the last update trust value sensitivity, provide recommendation service of the three decisions. This design method of an "*anomaly detection engine*", the engine is responsible for detecting dishonest recommendation. The experimental results can be well proved the validity of the method.

7. Acknowledgments

This work has been funded by College Natural Science Foundation of Jiangsu Province (11KJB520002), Jiangsu 973 Scientific Project (BK2011023, BK2011022), the National Natural Science Foundation of China (61272419,

60903027), China postdoctoral Foundation (2012M521089), Jiangsu Postdoctoral Foundation (1201044C), Jiangsu Natural Science Foundation (BK2011370), Research Union Innovation Fund of Jiangsu Province (BY2012022).

References

[1] Ramchurn, S D., Jennings, N R., Sierra, C., et al. (2004). Devising A Trust Model for Multi-agent Interactions Using Confidence and Reputation. *Applied Artificial Intelligence*, 18 (9-10) 833-852.

[2] Arning, A., Agrawal, R., Raghavan, P. (1996). A Linear Method for Deviation Detection in Large Databases. KDD, 164-169.

[3] Qianmu, Li., Hong, Zhang. (2012). Information Security Risk Assessment Technology of Cyberspace: a Review. *Information- an International Interdisciplinary Journal*, 15 (11) 4677-4684.

[4] Qianmu, Li. (2011). Multiple QoS Constraints Finding Paths Algorithm in TMN. *Information- an International Interdisciplinary Journal*, 14 (3) 731-738.

[5] Qianmu, Li., Jia, Li. (2012). Rough Outlier Detection Based Security Risk Analysis Methodology. *China Communications*, 5 (7) 14-21.

[6] Sugier, J. (2013). Computational Methods for Adaptation of Markov Models to Requested Maintenance Policies. *Computer Modelling and New Technologies*, 17 (1) 14-24.