A Context-Based Architecture for Reliable Trust Model in Ubiquitous Environments

Weiwei Yuan, Donghai Guan, Sungyoung Lee*, Youngkoo Lee
Department of Computer Engineering, Kyung Hee University, Korea
{weiwei, donghai, sylee}@oslab.khu.ac.kr, yklee@khu.ac.kr

Abstract - This paper presents a novel context-based architecture to filter out unfair and deceitful recommendations for trust model in ubiquitous environments. This approach has distinct advantages when dealing with randomly given irresponsible recommendations, unfair recommendations flooding as well as inside job (recommender acted honest gives unfair recommendations on the benefit of himself), which is lack of consideration in the previous works. In addition we originally give the possible scenarios of recommendations given by recommenders in the trust model to analyze the possible threats of trust model in ubiquitous environments. Finally we summarize the previous methods which were used to choose reliable recommendations and make a comparison with our approach.

1. INTRODUCTION

Trust is an important tool in human life, as it enables people to cope with the uncertainty caused by the free will of others [1]. Computational models of trust are suitable to be used to decide whether to allow unfamiliar entities to use specific services since ubiquitous environments are both highly dynamic and unpredictable.

When making trust decision, the service provider may sometimes not be familiar with the service requester or the service requester does not have enough privilege to access the service. Recommendations given by other entities which had past interactions with the service requester will be needed to help the service provider make trust decision. However, in this large-scale, open, dynamic and distributed ubiquitous environment, there may be numerous self-interested entities, i.e. which interacting in a way to maximize their own gain (perhaps at the cost of others). They may give unfair recommendations on their own benefit. Therefore finding ways to avoid or reduce the influence of unfairly positive or unfairly negative recommendations from self-interested entities is a fundamental problem for the trust model in ubiquitous environments.

The objective of this paper is to contribute to the construction of a trust model in ubiquitous environments which is robust in the presence of unfair and deceitful recommendations. This paper sets the stage by identifying a novel context-based approach in which context is used to analyze the user's activity, state and intentions. Based on the analysis of context, our approach compares the recommender's current recommendation with his past behavior to find the doubtful recommendations. The contributions of this paper are: (1) it originally analyzes the possible recommendation scenarios given by recommenders for trust model in the ubiquitous environments; (2) it uses the novel context-based approach for choosing reliable recommendations which has distinct advantages when dealing with randomly given irresponsible recommendation, unfair recommendations flooding as well as inside job.

The rest of the paper is organized as follows. Section 2 gives the recommendation scenarios in ubiquitous environments. Section 3 introduces the related works on choosing reliable recommendations. Section 4 presents our context-based approach in details. The last section concludes the paper and points out the future work.

2. RECOMMENDATION SCENARIOS IN UBIQUITOUS ENVIRONMENTS

The working procedure of the trust model in a ubiquitous supported smart office is shown in Fig.1. There are 4 categories of devices: (1) Service requester: user who uses his intelligent mobile device (e.g. cell phone, PDA) to request services. (2) Service provider: the device which provides the service in the smart office (e.g. scanner, copy machine, projector). (3) Service agent: the agent which is in charge of several service providers. There is a network of service agents in the smart office that provides different kinds of services. (4) Recommender: user who uses his intelligent mobile device to give recommendation for the service requester. The role of service requester and recommender can interchange in different situations.

The working procedure of the trust model includes 4 steps: (1) Service requester sends a request to service agent to apply a certain service (2) If the service requester is not an acquaintance to the service agent or it does not have enough privilege to access the service, the service agent will ask other users who are now in a certain region to give recommendations for this service requester. (3) If the users who are requested to give recommendations have past interaction history with the service requester, they act as recommenders and give recommendations for the service requester to service agent. (4) Service agent makes trust decision on behalf of the service pro-

* Corresponding Author

vider according to the recommendations given by recommenders.

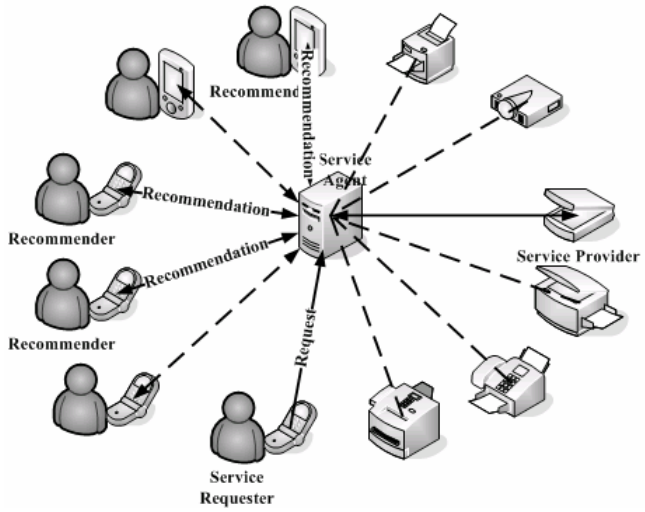

Fig 1. Recommendation Scenarios in Smart Office

We can observe from the above working procedure that recommendations given by recommenders are severe important since they directly affect the trust decision to the service requester. Hence it is exigent to filter out the unfair recommendations given by malicious recommenders which will do harm to the trust decision. Although there are many researches focus on this problem in the e-commerce, a comprehensive analysis of recommendation scenarios in the ubiquitous environments trust model is lacked. The recommendation scenarios in these two environments are totally different since in e-commerce the recommendations are from buyers to sellers, yet in ubiquitous environments the recommendations are from users to users, thus the motivation and scenarios are essentially different. This paper originally presents the recommendation scenarios in ubiquitous environments. The recommendation scenarios will provide guidance to understand the possible threats for trust model in the ubiquitous environments.

(1) Normal recommendation

a. Honest recommender gives an accurate recommendation.

(2) Abnormal recommendation

b. Honest recommender gives inaccurate recommendation due to its incorrect observation.

c. Honest recommender gives exceptional recommendation compared with others due to the changing characteristic of the service requester in front of different recommenders.

d. Recommender gives random value of recommendation at ease due to the lack of responsibility.

e. Recommender intentionally gives unfair high or low recommendation. This recommender acted honest, but suddenly gives unfair recommendation due to the relationship with the service requester or his own benefit.

f. Recommender intentionally gives unfair high or low recommendation, different from the recommender in scenario e, this recommender always gives malicious recommendations.

g. Service requester or some recommender collude a number of recommenders to give unfair recommendations in a certain time window, which causes the flooding of unfair recommendation values.

To be a reliable trust model in ubiquitous environments, it must has the ability to filter out the unfair recommendations in scenario e, f and scenario g, distinguish recommendations in scenario b and scenario d from recommendations in scenarios a, and tell scenario c apart from scenario b, d, e and scenario f.

3. RELATED WORKS

There are already some researches gave some helpful attempt on dealing with the unfair recommendations, especially for scenario f and scenario g. One method is to use polling method, e.g. in [3], the authors used polling method: basic polling and enhanced polling (differs from the basic solution by requesting voters to provide their servent_id). They relied on computing clusters of voters whose common characteristic suggests that the recommendations may have been created by a single, possible malicious, user. Another method is to give weighted value to different recommenders to choose reliable ones e.g. in [4], the authors uses weighted majority algorithm, and a so-called Rating Reputation Feedback is used to train the weighted values in [5]. In [6] [7] [8], the authors used neural network to calculate the reputation in order to filter out unfair recommendations and made the trust model adaptive to the multi-agent system. In [9][10], the authors used probability model to deal with the unfair recommendation, both of the papers used Bayesian analysis and regarded the prior distribution of the trust value as beta distribution. Another method is to use combination of different filters to deal with the unfair recommendations in online trading communities, as mentioned in [11] [12] [13] [14]. It pointed out that using controlled anonymity is an effective way to avoid unfairly low recommendations and negative discrimination. And using cluster filtering is suitable to reduce the effect of unfairly high recommendation and positive discrimination. The author also argued that the frequency filtering can guarantee the calculation of trust not be influenced by the unfair raters flooding (a relatively small number of unfair raters can manage to increase the ratio of unfair recommendation in any given time window above 50% and completely compromise the reliability of the system).

TABLE Ⅰ gives the comparison between the three main methods when dealing with different unfair recommendation scenarios mentioned in section 2. The reason these three previous methods can not deal with some scenarios lies in that the existing methods took one or more of the following assumptions: (1) most recommendations are close in the range to the real quality of the product, (2) recommendations provided by different recommenders on a given service requester will follow more or less the same probability distribution, (3) the top ranked recommenders are the expert recommenders in the trust category, i.e., the higher rank recommender has, the more authority his recommendation will be.

TABLE Ⅰ
COMPARISION OF DIFFERENT RELATED WORK

Scenario / Approach	Polling	Weight-Based	Combination of Filters
a	No effect	No effect	Negative reputation basis
b	√	Useless for high ranked recommender	√
c	×	√	×
d	×	×	Variance should be large
e	√	×	√
f	√	√	√
g (from low ranked recommender)	×	√	√
g (from high ranked recommender)	×	×	√

With assumption (3), it is impossible for the weigh-based method to deal with scenario e. What's more, if scenario e happens, the higher the recommender's rank is, the more serious aftereffect there will be. In addition, as mentioned in [11] [12] [13] [14], the combination of filters is at the cost of negative reputation bias in the absence of unfair recommendations. For scenario d, the disposal is based on the assumption that the distribution of random recommendations is much distinct from the normal distribution, but if the variance in the normal distribution of recommendations is not very large, the random recommendation is not sensible and it will encourage impassive agent [5]. The polling method is unsuitable to dealing with the unfair recommendations flooding since it takes the assumption (1).

4. THE PROPOSED APPROACH

4.1 Context and Context Classification

Based on the above analysis of the recommendation scenarios in ubiquitous environments, we propose a novel context-based approach to choose reliable recommendations. Context is any information that is useful to characterize the state or the activity of an entity. Context-based security is an emerging approach to cope with the new security problems introduced by the high dynamicity and heterogeneity that characterize pervasive and highly dynamic computing environment [2].

We identify contextual information into three types. (1) Simple: The collected information is used in its original format. For example, it can represent the value of a parameter. (2) Interpreted: The collected data cannot be used as it is but needs to be converted in a more meaningful format. For example, the contextual entry is "communication took place 2 days ago" that needs to be converted into "recent communication". (3) Composite: It is a set of simple and/or interpreted entries collected as a whole.

Not only can the isolated subsets of the context affect the efficiency of context-based applications, but also the relationships between different elements of context. Therefore, for the purpose of providing a comprehensive context classification system that includes the key elements of context that have an influence on a user's diverse activities in the ubiquitous environments, the context-based approach should have the following characteristics:

It should provide a standard form for describing human activity. Human cannot fully understand the full moment-to-moment richness of other humans' activities, states, goals and intentions. Yet they manage successfully and fluently to interact in many highly contextualized ways. Hence in attempting to produce better context-based trust model, it is neither possible nor necessary to model all the richness of human activity. To make progress from the current state of the art, we propose that a sufficiently comprehensive context classification may be developed using the relatively simple standard form that covers the key elements that have an influence on human activity.

It should relate individual human activity to society. Users are using the computing services within society and that society will have an influence on the user's activity.

The context-based approach must map the relationships among each element that identifies as having an influence on human activity.

The contexts used in the trust model of ubiquitous environments may include: (1) Time/date of request. (2) Current state. (3) Relationships with other agents. (4) Past interaction history with the service requester. (5) Time of last communication with service requester. (6) Confidence for the service requester in given time window. The first two types of context are specifically bounded to the agent activity or state; the latter are supposed to hold regardless of the agent state because they depend on the relationship with others where the agent is currently situated.

4.2 Choosing Reliable Recommendations Using Context-Based Approach

The key factor for building trust is the user's understanding of the information and the metrics used in trust evaluations. The most challenging aspect of trust is that it is subjective, so it is easy for the malicious recommender to pretend honest and for the honest recommender to be misunderstood as malicious because of the different understandings. Our key idea for the solution is that: though different recommenders have different understandings for the same information or entity, however, from the view of psychology, one recommender has similar understanding as himself in the similar context. Thus by comparing the recommendations with the recommender's own past behavior, it is possible to find the unfair recommendations given by different recommenders. We use the architecture shown in Fig.2 to filter out the unfair recommendations. The detail steps are as follows:

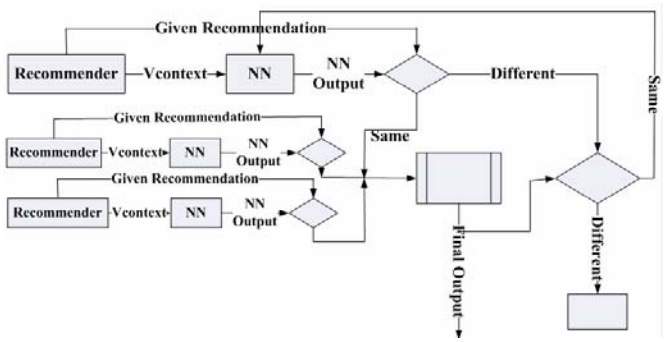

Fig. 2. Filtering Out Unfair Recommendations

First, using its own trust metric, each recommender gives his recommendation value RES_{org} along with the current context information—vector $V_{context}[TD, CS, RE, PI, TL, CF]$, where TD, CS, RE, PI, TL, CF represent the context information we mention in section 4.1 from (1) to (6) respectively.

$$RES_{org} = \begin{cases} 1 & trust \\ 0 & distrust \end{cases}$$

Secondly, for each recommender, its own context trained neural network is used to get the output with the input context vector $V_{context}[TD, CS, RE, PI, TL, CF]$. The output is RES_{NN}.

$$RES_{NN} = NN\{V_{context}[TD, CS, RE, PI, TL, CF]\} = \begin{cases} 1 & trust \\ 0 & distrust \end{cases}$$

Thirdly, compare RES_{org} and RES_{NN} to judge if the given recommendation value is doubtful.

$$RES_{com} = \begin{cases} RES_{NN} & RES_{NN} = RES_{org} \\ -1 & else \end{cases} \quad (1)$$

The reason we choose neural network to train the context data lies in that: (1) its remarkable ability to derive meaning from complicated and imprecise data is suitable to the training of trust since trust is subjective and the training value has probability to be imprecise, (2) its adaptive learning. Since the ubiquitous environments are dynamic, we need to dynamically adjust the parameter of neural network.

Fourthly, use all the normal recommendations from the third step to get the final recommendation. Since based on the context we already filtered out the doubtful recommendations of scenario b, d, e, f and scenario g in the previous step, we regard any $RES_{comi} \neq -1$ as a reliable recommendation. We use simple voting mechanism to calculate the final recommendation RES_{fin}.

$$RES_{fin} = \begin{cases} 1 & NUM[RES_{comi}=1 \,|\, RES_{comi} \neq -1] \geq \dfrac{NUM[RES_{comi} \neq -1]}{2} \\ 0 & else \end{cases} \quad (2)$$

where RES_{comi} is the RES_{com} recommender i. $NUM[RES_{comi} \neq -1]$ is the number of undoubtful recommenders gotten in step 3. $NUM[RES_{comi} = 1 \,|\, RES_{comi} \neq -1]$ is the number of undoubtful recommenders who consider the service requester can be trusted .

Finally, we look back to step3. In formula (1), If $RES_{NN} = RES_{org}$, we consider the current situation is under scenario a and scenario c since the recommender gives the same recommendation in the similar context. Otherwise, if $RES_{NN} \neq RES_{org}$, the possible situations are: A. It belongs to one of the scenario b, d, e, f and scenario g. B. As the changing of the environment or the recommender himself, his judging standards changed, i.e. the recommender makes different decision from previous even in the similar context. However, this recommendation is also an honest one. This kind of situation is reasonable since all the things in the world are always in dynamic movement between balance and imbalance. We use the following step to tell situation B apart from situation A.

$$result = \begin{cases} situationA & REC_{org} \neq REC_{fin} \,|\, REC_{com} = -1 \\ situationB & REC_{org} = REC_{fin} \,|\, REC_{com} = -1 \end{cases} \quad (3)$$

If the result is situation B, our architecture gives the context as well as the given recommendation back to the Neural Network shown in Fig.2 and re-train the neural network. Otherwise if the result is situation A, the record of this recommender is given to a separated disposal unit to mark it as a doubtful recommender. If one user always appears as a doubtful recommender, he will be considered as either a malicious recommender or a recommender who does not have enough ability to give correct recommendations. The recom-

mendations given by this recommender will be filter out directly next time.

5. CONCLUSION

In this paper we propose a robust trust model for ubiquitous environments, in which a context-based approach is used to filter out the unfair recommendations including the intended unfair recommendations as well as the mis-observation of the recommenders. We also focus on the flooding of unfair recommendation in this paper. Since our approach concentrates on the abnormal behavior of each recommender, it has special advantages when dealing with inside job, which is lack of consideration in current trust models. What's more, we give the analysis of recommendation scenarios in the ubiquitous computing environment, which is different from in the e-commerce environment because of the distinct intentions.

The advantages of our context-based architecture are: (1) It is able to filter out incorrect observations by honest recommenders. (2) It is able to filter out the randomly given irresponsible recommendations even the variance of distribution is not very large. (3) It is able to filter out the suddenly appear malicious recommendations from the recommenders who acted honest (inside job). (4) It is able to defend the unfair recommendations flooding, no matter the flooding is from the recommenders who acted honest or malicious. (5) When there is no unfair recommendation, it has no negative bias on the recommendations.

Compared with other methods, the cost of our approach is that our approach needs more computation, because our architecture needs to dispose each context by the neural network to judge the validity of the recommendations. However, since these calculations take place in the service agent (as shown in Fig.1) which has enough computing ability, we believe that it does not distinctly affect the efficiency of the trust model.

In the future work, we plan to add the risk analysis in our context-based trust model and implement our trust model to use in our CAMUS [15] middleware. Based on the analysis of our context-based approach and other methods, we believe that the usage of context-based trust model in ubiquitous environments applications presents a promising path for the future research.

ACKNOWLEDGMENT. This research was supported by the Driving Force Project for the Next Generation of Gyeonggi Provincial Government in Republic of Korea.

6. REFERENCES

[1] Sini Ruohomaa, Lea Kutvonen , "Trust Management Survey", iTrust 2005, Paris, France.

[2] Rebecca Montanari, Alessandra Toninelli, Je ffrey M. Bradshaw, "Context-based security management for multi-agent systems", MAS&S 2005, Philadelphia, USA.

[3] Damiani, Vimercati, Paraboschi, Samarati, and Violante, "A reputation-based approach for choosing reliable resources in peer-to-peer networks", 9th ACM CCS 2002

[4] Bin Yu, Munindar P. Singh, and Katia Sycara, " Developing trust large-scale peer-to-peer systems", First IEEE Symposium on Multiagent Security and Survivability, 2004.

[5] Ping Xu, Ji Gao, Hang Guo, "Rating Reputation: a necessary consideration in reputation mechanism", Proceedings of 2005 International Conference on Machine Learning and Cybernetics.

[6] Weihua Song, Vir V. P hoha, and Xin Xu, "An adaptive recommendation trust model in multiagent system", IEEE/WIC/ACM IAT'04.

[7] Weihua Song, Vir V. Phoha, "Neural network-based reputation model in a distributed system", pp. 321-324, 2004 IEEE International Conference on E-Commerce Technology (CEC'04), 2004.

[8] Huang Baohua; Hu Heping; Lu Zhengding, "Identifying local trust value with neural network in p2p environment", The First IEEE and IFIP International Conference in Central Asia on Internet, 2005

[9] Whitby, A., Josang, A., and Indulska, J. "Filtering out unfair ratings in Bayesian reputation systems", AAMAS 2004, New York, USA.

[10] Patel J., Teacy W. T. L., Jennings N. R. and Luck M., "A probabilistic trust model for handling inaccurate reputation sources", In Proceedings of Third International Conference on Trust Management, pp. 193-209,2005

[11] C.Dellarocas, "The design of reliable trust management systems for electronic trading communities", MIT Working Paper.

[12] C. Dellarocas , "Building trust online: the design of robust reputation reporting mechanisms for online trading communities" A combined pe rspective on the digital era, Doukidis, G., Mylonopoulos, N. and Pouloudi, N. (Eds.), Idea Book Publishing (2004).

[13] C. Dellarocas. "Immunizing online Reputation Reporting systems against unfair ratings and discriminatory behavior", In Proceedings of the ACM Conference on Electronic Commerce, pages 150--157, Minneapolis, Minnesota, USA, 2000.

[14] Chrysanthos Dellarocas , "Mechanisms for coping with unfai r ratings and discriminatory behavior in online reputation reporting systems", In ICIS, pages 520--525, 2000.

[15] Hung Q. Ngo, Anjum Shehzad, Saad Liaquat Kiani, Maria Riaz, Kim Anh Ngoc, Sungyong Lee.: Developing Context-aware Ubiqutious Co mputing Systems with a Unified Middleware FrameWork. The 2004 Internation Conference on Embedded & Ubiquitous Computing(EUC2004)