Filtering Out Unfair Recommendations for Trust Model in Ubiquitous Environments

Weiwei Yuan[1], Donghai Guan[1], Sungyoung Lee[1], Young-Koo Lee[1,⋆], and Heejo Lee[2]

[1] Department of Computer Engineering, Kyung Hee University, Korea
[2] Department of Computer Science and Engineering, Korea University

Abstract. This paper presents a novel context-based approach to filter out unfair recommendations for trust model in ubiquitous environments. Context is used in our approach to analyze the user's activity, state and intention. Incremental learning based neural network is used to dispose the context in order to find doubtful recommendations. This approach has distinct advantages when dealing with randomly given irresponsible recommendations, individual unfair recommendations as well as unfair recommendations flooding.

1 Introduction

The basis for the trust model to make decision on unfamiliar service requesters are the recommendations given by recommenders who have past interaction history with the requesters. However, in the large-scale, open, dynamic and distributed ubiquitous environments, there may possibly exist numerous self-interested recommenders who give unfair recommendations to maximize their own gains (perhaps at the cost of others). Therefore, finding ways to avoid or reduce the influence of unfair recommendations from self-interested recommenders is a fundamental problem for trust model in ubiquitous environments.

The possible scenarios for unfair recommendations are: (1) Individual Unfair Recommendation: honest recommender gives inaccurate recommendation due to incorrect observation, or the recommender maliciously gives unfair recommendation (the recommender may be a malicious node or a node which acted honest but suddenly gives unfair recommendation due to his own benefits (called Inside Job)). (2) Unfair Recommendations Flooding: a number of recommenders collude to give unfair recommendations (more than 50% of total recommendations), which causes the flooding of unfair recommendations. The flooding may be caused by malicious nodes or those who acted honest (called Inside Job Flooding). (3) Randomly Given Recommendation: the recommender gives random recommendation due to the lack of responsibility.

There are mainly three methods had been proposed for filtering out unfair recommendations in previous works. One is to use polling method, e.g. in [1], the authors used basic polling as well as enhanced polling. The enhanced polling

⋆ Corresponding author.

A. Bagchi and V. Atluri (Eds.): ICISS 2006, LNCS 4332, pp. 357–360, 2006.

differs from basic polling by requesting voters to provide their servent_id to prevent a single malicious user to create multiple recommendations. Another method is to give weighted value to each recommender (also called reputation based method) [2], [3]. This method regards recommendations given by low reputation recommenders as malicious. The third method is to use the combination of filters [4]. It suggests that cluster filtering is suitable to reduce the effect of unfairly high recommendations and frequency filtering can guarantee the calculation of trust not be influenced by the unfair recommendations flooding. However, these methods take at least one of the following assumptions, which makes them disable to deal all the unfair recommendations scenarios: (1) recommendations provided by different recommenders on a service requester will follow more or less the same probability distribution, (2) the higher rank the recommender has, the more authority his recommendation will have. E.g., it is impossible to filter out Inside Job and Inside Job Flooding using reputation based method since it takes assumption (2).

This paper introduces a novel context-based approach using incremental learning algorithm to deal with the possible unfair recommendation scenarios. Instead of taking the assumptions of previous works, context is used in our approach to analyze the user's activity, state and intention. The learning of context is incrementally increased by a Cascade-Correlation architecture neural network.

2 The Proposed Approach

Trust is subjective since it is based on each user's own understanding. Hence it is relatively easy for the malicious recommender to pretend honest and for the honest recommender to be misunderstood as malicious because of the different understandings, which makes it difficult to differentiate between the unfair and fair recommendations. Our key idea for the solution is that: recommenders may give different recommendations due to their different understandings, however, one recommender will follow the rule of himself, i.e., one recommender usually gives similar recommendations in similar context. In case one recommender gives exceptional recommendations compared with his own previous ones in similar context, the reason lies in two aspects. One is that this recommendation is a mischievous one. The other is that the recommender's rule on recommendation giving has changed, e.g. the recommender now only gives positive recommendation to requesters whose past interaction with him is more than 80% successful in stead of 60%.

We use incremental learning based neural network, the Cascade-Correlation architecture in particular, to learn each recommender's rule on recommendation giving since the acquisition of a representative training data for the rule is time consuming and the rule is also possible to dynamically change from time to time. Cascade-Correlation is useful for incremental learning, in which new information is added to an already-trained network. It begins with minimal network, then automatically trains and adds new hidden units one by one, creating a multi-layer structure [5]. Fig. 1 gives the process of training Cascade-Correlation. In

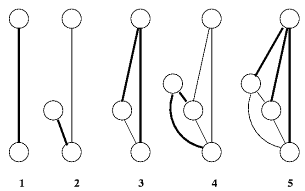

Fig. 1. Training of Cascade-Correlation Architecture

1, we train weights from input to output. In 2, we add a candidate unit and train its weights to maximize the correlation with the error. In 3, we retrain the output layer. We train the input weights for another hidden unit in 4. Output layer is retrained in 5, etc. The usage of Cascade-Correlation architecture has several advantages: it learns quickly; the network determines its own size and topology; it retains the structures it has built even if the training set changes.

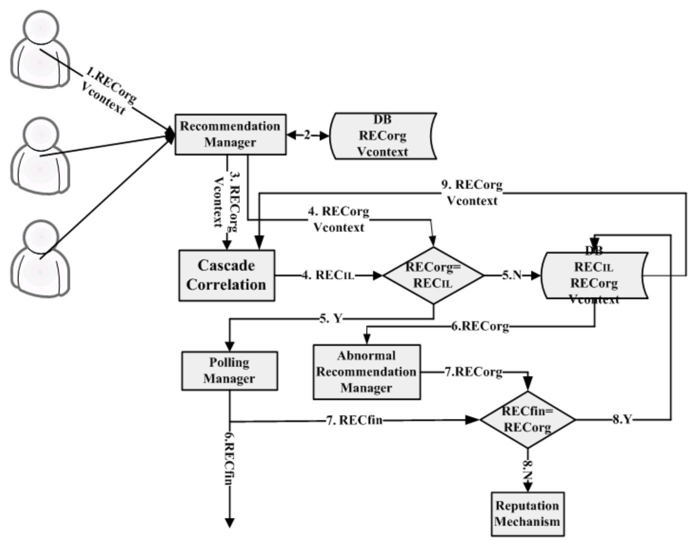

Fig. 2. Architecture for Filtering out Unfair Recommendations

We use the architecture shown in Fig. 2 to filter out the unfair recommendations. Recommendation Manager first collects recommendations (REC_{org}) from all recommenders, along with the context value $V_{context}$ under which recommendations were given. For each recommender, the input of Cascade-Correlation architecture is $V_{context}$ and the output is REC_{IL}, which is the recommendation that one recommender will give due to his past behavior when given $V_{context}$. If REC_{org}=REC_{IL}, it means that the recommender gives the same recommendation as previous behavior. In this case, we regard REC_{com} as a reliable recommendation and use basic voting mechanism to calculate the final recommendation REC_{fin}. Otherwise if $REC_{org} \neq REC_{IL}$, REC_{org} is regarded as a doubtful recommendation. In this case, if $REC_{org} \neq REC_{fin}$, we regard REC_{org} as mischievous or incorrect. Otherwise, if $REC_{org} = REC_{fin}$, the possible situations are: (1) the recommender's rule on recommendation giving has changed, (2) the currently neural network is not enough to reflect the recommender's rule on recommendation giving since the Cascade-Correlation architecture begins with a minimal network and the knowledge on the recommender's rule is incrementally increased. In this case, $V_{context}$ as well as REC_{org} will be given back as retrain data to the Cascade-Correlation architecture.

3 Conclusions

In this paper we propose a robust trust model for ubiquitous environments, in which a context-based approach is used to filter out unfair recommendations. The learning of the context is based on incremental learning neural network. The filtered out recommendations may be the intended unfair recommendations as well as the mis-observation by the recommenders. Since our approach concentrates on the doubtful behaviors of each entity, it has special advantages when dealing with inside job, which is lack of considerations in previous works. In the future work, we plan to simulate our proposed method based on CAMUS [6] middleware. We also plan to add risk analysis in our context-based trust model. We believe that to filter out unfair recommendations by using context-based trust model presents a promising path for the future research.

Acknowledgements. This research is supported by the Ubiquitous Computing and Network (UCN) Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

References

1. F. Cornelli, E. Damiani, S.D.C.D. Vimercati: Choosing Reputable Servants in a P2P Networks. ACM WWW2002, USA.
2. BP. Xu, J. Gao, H. Guo: Rating Reputation: a necessary consideration in reputation mechanism. Proceedings of 2005 International Conference on Machine Learning and Cybernetics.
3. W. Song, V. V. Phoha, X. Xu: An adaptive recommendation trust model in multi-agent system. IEEE/WIC/ACM IAT'04.
4. C. Dellarocas: Building trust online: the design of robust reputation reporting mechanisms for online trading communities. A combined perspective on digital era, Idea Book Publishing (2004).
5. S.E. Fahlman, C. Lebiere: The Cascade-Correlation Learning Architecture. ATechnical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University.
6. H.Q. Ngo, A. Shehzad, S.L. Kiani, M. Riaz, K. A. Ngoc, S.Y. Lee: Developing Context-aware Ubiquitous Computing Systems with a Unified Middleware FrameWork. EUC2004.