Towards a Comprehensive Testbed to Evaluate the Robustness of Reputation Systems against Unfair Rating Attacks

Athirai Aravazhi Irissappane, Siwei Jiang, and Jie Zhang

School of Computer Engineering
Nanyang Technological University, Singapore
{athirai001, sjiang1, zhangj}@ntu.edu.sg

Abstract. Evaluation of the effectiveness and robustness of reputation systems is important for the trust research community. However, existing testbeds are mainly simulation based and not flexible to perform robustness evaluation, and none of them is specifically designed to evaluate the robustness of reputation systems against unfair rating attacks. In this paper, we propose a novel comprehensive testbed by simulating three types of environments (simulated environments, real environments with simulated unfair rating attacks, and real environments with detected unfair ratings). The testbed incorporates sophisticated deception models and unfair rating attack models, and introduces several performance metrics to fully test and compare the effectiveness and robustness of different reputation systems. We also provide two case studies to demonstrate the usage of partial features of our proposed testbed.

Keywords: Testbed, Robustness, Reputation Systems, Unfair Ratings

1 Introduction and Motivation

Reputation systems strengthen the quality of electronic marketplaces by providing incentives for good behavior of sellers and their high quality services and by sanctioning their bad behavior and low quality services [3]. However, the performance of reputation systems may be affected by unfair rating attacks from dishonest buyers (also called advisors). To detect unfair ratings and to assist buyers in accurately evaluating the reputation of sellers, different approaches such as BRS [4, 9], iCLUB [7], TRAVOS [8], WMA [11], and the Personalized approach [12] have been proposed for reputation systems.

However, the majority of the reputation systems with those unfair rating detection approaches have only been evaluated using experimental frameworks of their authors' own. They have been compared with only a very few other approaches. And, most of the experimental frameworks are based on simple simulated scenarios, which often cannot be considered as reliable evidence for how the reputation systems would perform in a realistic environment.

Some unified testbeds (such as ART [2] and TREET [5]) have been proposed for the evaluation of trust and reputation systems. However, they are mainly

simulation-based and also cannot reflect real environmental settings. Besides, they are not specifically designed for evaluating the robustness of reputation systems in coping up with unfair rating attacks. Another shortcoming of those testbeds is that they often propose only one performance metric. Hence, there arises an urgent need to develop a comprehensive testbed to evaluate reputation systems in order to fully analyze their actual effectiveness and robustness.

In this paper, we propose a comprehensive testbed to evaluate and compare different reputation systems with their unfair rating detection approaches. We simulate three types of environments, including simulated environments, real environments with simulated unfair rating attacks, and real environments with detected "ground-truth" about which ratings are unfair. The testbed incorporates sophisticated deception models of sellers and various attack models of buyers to fully test the effectiveness and robustness of reputation systems. We also introduce some novel performance metrics to represent the robustness of reputation systems against unfair rating attacks. In the current work, we have implemented two environments, simulated environments and real environments with simulated unfair rating attacks. We have also conducted experiments in those two environments to evaluate several existing reputation systems, to demonstrate the usage of partial features of our proposed testbed. While this initiative stands to model a comprehensive testbed, it may not be seen as a complete implementation of the proposal. This work is meant to be a primary step for the proposed testbed. We believe that our work would be beneficial for the researchers in the field to analyze and compare their approaches with the purpose of improving their performance by providing a comprehensive testbed that offers flexibility to adjust parameters of experiments according to their needs.

2 Related Work

In this paper, we focus on evaluating the robustness of reputation systems against unfair rating attacks. Next, we provide a brief summary of some approaches for handling unfair ratings in reputation systems and some existing testbeds.

2.1 Approaches for Handling Unfair Ratings

To handle unfair ratings in reputation systems, various approaches have been proposed. For example, the Beta Reputation System (BRS) [4, 9] iteratively filters out unfair ratings based on a *majority rule*. If the calculated reputation of a seller based on the ratings of the set of honest buyers falls in the rejection area (q quantile or $1-q$ quantile) of the beta distribution of a buyer's ratings to that seller, this buyer will be filtered out from the set of honest buyers. TRAVOS [8] copes with unfair ratings by accomplishing two tasks. The first task is to estimate the accuracy of the current ratings based on the amount of fair and unfair previous ratings which are similar to the current ratings. The second task is to adjust the current ratings according to the accuracy. The aim of this task is to

reduce the effect of unfair ratings. In the Personalized approach [12], the trustworthiness of advisors takes into account both the buyer's personal experience with the advisor's ratings and the public knowledge about the advisor. When the buyer has enough private information about (personal experience with) the advisor, the buyer uses private knowledge alone otherwise uses an aggregation of both the private and public knowledge to compute the trustworthiness of the advisor. The iCLUB approach [7] can handle multi-nominal ratings. It applies clustering approaches and considers buyer's local and global knowledge about the sellers to filter out unfair ratings. Yu and Singh [11] propose a Weighted Majority Algorithm (WMA) to adjust the trustworthiness of advisors. If a rating provided by an advisor deviates from the majority of other advisors' ratings, the trustworthiness of the advisor will be decreased.

2.2 Existing Testbeds

The Agent Reputation and Trust (ART) testbed [2] is an example of a testbed that has been specified and implemented by an international group of researchers. The ART testbed specification is an artwork appraisal domain where appraisers need to buy artwork about which they may have limited knowledge. However, it is currently not flexible enough for carrying out realistic simulations and robustness evaluation for many trust and reputation systems [3]. Also, the integration with the testbed is quite challenging [5]. Furthermore, the winning approach in the ART testbed does not consider reputation ratings from other appraisers. This decision raises concern about the importance of an approach for coping with unfair ratings in this testbed, and whether the results of comparing unfair rating detection approaches based on this testbed will be significant. The testbed is also not flexible to support centralized trust and reputation systems [1].

TREET [5] is a testbed which models a general e-marketplace scenario. TREET supports both centralized and decentralized reputation systems and allows collusion attacks to be implemented. But like ART, TREET is a simulated environment which may not exactly depict the realistic environment. It is also not specifically designed to evaluate unfair rating detection approaches.

3 High Level Architecture of the Proposed Testbed

In this paper, a comprehensive experimental testbed is proposed to specifically evaluate and compare reputation systems for detecting unfair rating attacks. Some of the unique features of the testbed are listed below:

- Robustness evaluation: The testbed evaluates the robustness of reputation systems in handling unfair ratings. Metrics (*e.g.*, number of unfair ratings required by attackers to change a target's reputation, transaction volume difference [13]) have been designed specifically for robustness evaluation.
- Multiple types of environments: Three types of environments, including simulated environments, real environments with simulated unfair rating attacks,

and real environments with detected "ground-truth" about which ratings are unfair, will be considered in the testbed.
- Unified platform: The testbed is a comprehensive unified platform as it supports both centralized and decentralized reputation systems. It also supports simple attacks as well as complicated attacks like collusion attacks.
- Scalability: The testbed allows to freely add reputation systems, attack models and performance metrics for the purpose of comprehensive evaluation.
- Flexibility: The testbed can deal with various types of ratings namely binary, multi-nominal and continuous ratings. If offers the flexibility to choose any environmental settings or experimental parameters.
- Comparison and experimentation: Apart from evaluating the effectiveness of reputation systems, the testbed also allows to compare their performance in a variety of experimental settings. Comparison of reputation systems under the same attack model but in different environments or that under different attack models but in a same environment are both supported by the testbed.

Figure 1 shows the high level architecture of the proposed testbed. The detailed design of the major modules is described below.

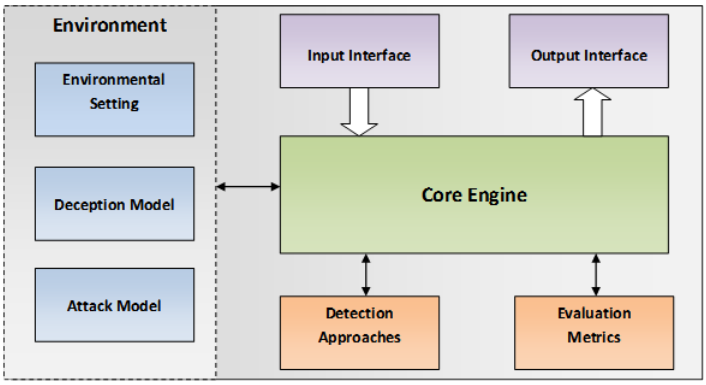


Fig. 1. High Level Architecture of the Testbed.

Input Interface: The input interface provides a convenient way to configure different environmental settings as well as set up customized experiments.

Output Interface: The output interface gives ease of information consumption, including visualization tools. Graphical representation and analysis of the results which is necessary to comprehend the evaluation results in an easy manner is supported by the output interface.

Core Engine: The core engine is responsible for managing all the related interfaces of the system. The main functionalities of the core engine include: bootstrapping experimentation process and displaying input interface; creating an environment based on configurations; coordinating all components of the testbed to accomplish experimentation tasks, collecting results and sending data to the output interface for display.

Detection Approaches: It involves the implementation of the reputation systems with their unfair rating detection approaches which are to be evaluated and compared against one another based on their robustness and performance against unfair rating attacks.

Evaluation Metrics: There are some conventional evaluation metrics, such as the mean absolute error of estimating reputation of targets, or the precision and recall of whether a rating is unfair. Besides these, we introduce several novel evaluation metrics to specifically evaluate the robustness of reputation systems against unfair rating attacks. In [13], we proposed robustness metric as the transaction volume difference between honest and dishonest duopoly sellers in an electronic marketplace environment. This is based on the fact that if a reputation system is completely robust against a certain attack, it will always suggest honest buyers to transact with honest duopoly sellers. Otherwise, under a certain attack, if the system always suggests honest buyers to transact with dishonest duopoly sellers, it means that this system is completely vulnerable to the attack. Another proposed robustness metric is the number of unfair ratings required by attackers (dishonest advisors) to change a target's reputation. This is because a more robust reputation system costs more efforts from attackers.

Environment: The most important component in the testbed is the environment which includes the environmental setting, attack models to model the behavior of dishonest buyers, and deception models to model the behavior of dishonest sellers. It is not easy to obtain a dataset from a real environment with ground truth about which ratings are unfair because 1) it is costly for system managers to find out the ground truth if human subjects are hired to inspect every rating, whatever interaction is rated by the rating, and whoever is involved in the interaction; 2) system managers with such information may not be willing to share it. Hence, in the proposed testbed, we use three ways to generate the environment for experimentation as described below:

- Simulated Environment: This environment is entirely based on simulations, but it has several unique and better design decisions compared to other simulation-based environments in the literature: a) the testbed offers flexible selection of environmental settings that follow those of the real world environments. We extract statistics in several real environments and generate data distributions for environmental settings; b) the testbed incorporates different deception strategies for malicious targets to choose from, including a sophisticated adaptive deception strategy where the malicious target may learn from the environment and accordingly adjust its deception type and frequency; c) various attack models, such as Constant attack, Camouflage attack, Whitewashing attack, and combinations of them are integrated for buyers to decide what ratings to provide [13]. A particularly challenging attack is the Collusion attack where strategic buyer may work together to provide unfair ratings to some target sellers. Compared with the real environment with simulated attacks that is introduced below, the simulated environment has the advantage of allowing users of the testbed to vary the deception strategies of the targets.

- Real Environment with Simulated Attacks: In this case, we collect data from real environments, such as IMDB, TripAdvisor, Amazon, and eBay. Various attacks from users (or buyers) are simulated based on the attack models in order to be mixed with the real data.
- Real Environment with Detected "Ground Truth": Here, we rely on spam review detection tools to detect spam reviews in the collected real data [6]. Ratings associated with spam reviews will then be treated as unfair ratings.

Reputation systems can be evaluated and verified for their robustness and effectiveness in all the three kinds of environments. In this paper, we evaluate some existing reputation systems in simulated environments (**Case Study 1**) and real environments with simulated unfair rating attacks (**Case Study 2**).

4 Case Study 1: Simulated Environment

In this case study, the environment is entirely based on simulation of a *Duopoly Market* with a reasonable competition scenario, where a dishonest duopoly seller tries to beat its honest competitor in transaction volume by hiring or collaborating with dishonest buyers to perform unfair rating attacks.

4.1 Environmental Settings

In the context of the simulated e-marketplace, when a buyer evaluates the reputation of a potential seller, it may need to ask for other buyers' opinions (advisor' ratings) towards that seller.

Attack Models: Dishonest advisors may provide unfair ratings to sellers.

- *Constant Attack*: Dishonest advisors constantly provide unfairly positive ratings to dishonest sellers, while giving negative ratings to honest sellers.
- *Camouflage Attack*: Dishonest advisors may camouflage themselves as honest ones by providing fair ratings strategically. *e.g.*, dishonest advisors can give fair ratings for a period, and then exploit their trustworthiness later.
- *Whitewashing Attack*: In e-marketplaces, it is difficult to establish buyers' identities: users can freely create a new account as a buyer. This presents an opportunity for a dishonest buyer to whitewash its low trustworthiness by starting a new account with the default initial trustworthiness value (0.5 in our investigated reputation systems).
- *Sybil Attack*: Dishonest buyers obtain larger amount of resources (buyer accounts) than honest buyers to constantly provide unfair ratings to sellers.
- *Sybil Camouflage Attack*: As the name suggests, this attack combines both the Camouflage attack and Sybil attack.
- *Sybil Whitewashing Attack*: Here, the number of dishonest buyers is larger than that of honest buyers and they perform the Whitewashing attack.

Detection Approaches: BRS, iCLUB, TRAVOS, WMA and the Personalized approach are evaluated to cope with the above attacks. The set of buyers

are defined as $B = \{B_i | i = 1, \cdots, l\}$, advisors as $A = \{A_j | j = 1, \cdots, m\}$ and sellers as $S = \{S_k | k = 1, \cdots, n\}$. The actual and predicted reputation of seller S_k is $Rep(S_k)$ and $\widehat{Rep}(S_k)$, respectively. The rating to seller S_k from buyer B_i is R_{B_i,S_k}. The trustworthiness of advisor A_j from the view of buyer B_i is $T_{B_i}(A_j)$. The estimated reputation of the seller S_k, $\widehat{Rep}(S_k)$, is then calculated as:

$$\widehat{Rep}(S_k) = \frac{\sum_{j \neq i} T_{B_i}(A_j) \times pos_j(S_k) + \epsilon}{\sum_{j \neq i} T_{B_i}(A_j) \times (pos_j(S_k) + neg_j(S_k)) + 2\epsilon} \tag{1}$$

where $pos_j(S_k)$ and $neg_j(S_k)$ are the number of positive and negative ratings from each advisor A_j to the seller S_k, and $T_{B_i}(A_j) \in [0, 1]$. When S_k does not receive any ratings, its initial reputation is 0.5. For BRS and iCLUB, $T_{B_i}(A_j) = 1$ when B_i selects A_j as its honest advisor; otherwise, $T_{B_i}(A_j) = 0$. The parameter $\epsilon \neq 0$ is a small constant to avoid the case of dividing by zero.

Simulation Settings: We set the number of honest duopoly sellers as 1, number of dishonest duopoly sellers as 1, number of honest common sellers as 99, number of dishonest common sellers as 99, number of honest buyers/advisors ($|B^H|$) as 28 for non-Sybil-based attack or 12 for Sybil-based attack, number of dishonest buyers/advisors or attackers ($|B^D|$) as 12 for non-Sybil-based attack or 28 for Sybil-based attack, number of simulation days ($|Days|$) as 100 and the ratio of duopoly sellers' transactions to all transactions ($ratio$) as 0.5.

Evaluation Metric: The *Robustness of a reputation system* (defense, Def) against an unfair rating attack model (Atk) is:

$$\mathcal{R}(Def, Atk) = \frac{|Tran(S^H)| - |Tran(S^D)|}{|B^H| \times |Days| \times ratio} \tag{2}$$

where $|Tran(S^H)|$ and $|Tran(S^D)|$ denote the total transaction volume of the honest and dishonest duopoly seller, respectively. If a reputation system is completely robust against an attack, $\mathcal{R}(Def, Atk) = 1$. On the contrary, if Def is *completely vulnerable* to Atk, $\mathcal{R}(Def, Atk) = -1$. When $\mathcal{R}(Def, Atk) > 0$, the greater the value is, the more robust Def is against Atk. When $\mathcal{R}(Def, Atk) < 0$, the greater the absolute value is, the more vulnerable Def is to Atk.

4.2 Experimental Results

The robustness of the reputation systems is calculated based on Eq. 2 and the results are presented in Table 1 which shows the mean and standard deviation ($mean \pm std$) over 50 independent runs. The best results are in bold font. Here, we test reputation systems when the rating type is real. The rating given by buyer B_i to seller S_k is $R_{B_i,S_k} \in [0, 1]$. Since the BRS, TRAVOS and Personalized approaches are designed to deal with binary ratings, we convert the real ratings to binary ratings. If $R_{B_i,S_k} \in [0, 0.5)$, it is translated as a negative rating $neg_i(S_k)$; otherwise, a positive rating $pos_i(S_k)$ is assigned.

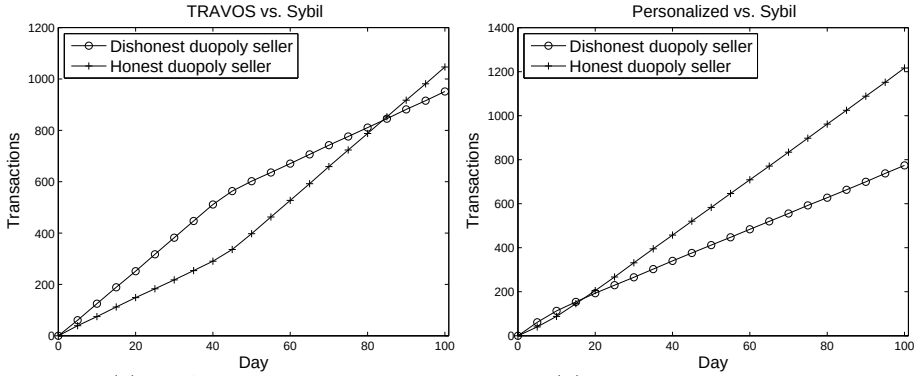
From Table 1, none of the reputation systems is completely robust against all the attacks. iCLUB obtains 2 best results for Sybil Camouflage and Sybil

Table 1. Robustness of Reputation Systems against Attacks

	Constant	Camouflage	Whitewashing	Sybil	Sybil Cam	Sybil WW
BRS	0.87±0.03	0.89±0.02	-0.18±0.07	-0.99±0.08	-0.47±0.07	-0.30±0.07
iCLUB	0.98±0.03	0.99±0.03	0.79±0.14	0.21±0.32	**0.94±0.10**	**0.20±0.29**
TRAVOS	0.97±0.02	0.82±0.03	0.87±0.03	0.16±0.09	-0.57±0.07	-0.98±0.07
WMA	0.89±0.04	0.69±0.04	-0.95±0.08	**0.82±0.06**	0.63±0.08	-0.98±0.07
Personalized	**0.99±0.03**	**0.99±0.03**	**0.98±0.03**	0.74±0.45	**0.94±0.08**	-1.00±0.08

*Sybil Cam: Sybil Camouflage Attack; Sybil WW: Sybil Whitewashing Attack

Whitewashing attacks. But $std = 0.29$ signifies that iCLUB cannot arrive to the stable robust state to handle the Sybil Whitewashing attack. WMA obtains 1 best result for Sybil. Personalized obtains 4 best results. All the Reputation systems are robust against Constant (baseline). Sybil Whitewashing is the most powerful attack. None of reputation systems is completely robust against it (*i.e.*, $\mathcal{R}(Def, \text{Sybil Whitewashing}) = 1$).



(a) TRAVOS vs. Sybil (b) Personalized vs. Sybil

Fig. 2. Transactions along Days under the Sybil Attack

From Figure 2, TRAVOS is not completely robust against Sybil. In the early period, TRAVOS cannot find enough reference sellers so the discounting is not effective (called *soft punishment*). For instance, if the trustworthiness of dishonest/honest advisor is 0.4/0.6, and a buyer gives one rating to a seller, according to Eq. 1, an honest seller's reputation is $0.39 < 0.5$ ($0.39 = (0.6 \times 12 + \epsilon)/(0.4 \times 28 + 0.6 \times 12 + 2\epsilon)$, suppose $\epsilon = 1e - 6$) and that of the dishonest seller is $0.61 > 0.5$; both suggest inaccurate decisions. However, if a reputation system is able to set the dishonest/honest advisor's trustworthiness as 0.1/0.9, the evaluation of sellers' reputation will become accurate. For Personalized against Sybil, at the beginning, it suffers *soft punishment* when the buyer relies on public information to evaluate advisors' trustworthiness. Figure 2 shows that, as transactions increase, TRAVOS and Personalized become effective after Day 80 and Day 15, respectively.

5 Case Study 2: Real Environment with Simulated Unfair Rating Attacks

In this case study, we collect dataset from a real environment (*e.g.*, an online rating system) instead of using simulations. Attack models are adopted to generate unfair ratings. The mixed data that combines the real dataset and the simulated unfair ratings is then used to assess the performance of reputation systems.

5.1 Environmental Settings

Real Dataset: Real data is obtained from IMDB (`http://www.imdb.com`). The information extracted includes userID, ratings, date, movieID, movie name, usefulness, director name, directorID, etc. This data is first pre-processed to remove noise[1] and filter out users and directors according to some predefined thresholds (*e.g.*, users who provided less than 5 ratings to directors are removed).

Attack Models: The RepBad, RepSelf and the Reptrap attack models [10] are implemented to generate unfair ratings. A complementary unfair rating type is used in the three attack models for the purpose of verifying whether the attack is successful or not. The main goal of these attacks is to overturn the quality of the target director by providing unfair ratings.

- *RepBad*: The attacker registers multiple userIDs and gives unfair ratings to the target item directly such that the items' quality is overturned.
- *RepSelf*: The attacker gives honest ratings to uninterested items to boost its trust value (self promotion) before giving unfair ratings to the target item.
- *Reptrap*: The attacker first gives unfair ratings to some unpopular items. Next, the attacker chooses some non-target items to give fair ratings and then gives unfair ratings to the target items. The unfair ratings are provided such that the items' quality is overturned.

Detection Approaches: We evaluate the robustness of the BRS, TRAVOS and Personalized approaches. Here, movie directors are considered to be sellers and the users as buyers and advisors in an e-marketplace environment. The formulation of the seller's (director's) reputation is as per Eq. 1.

Simulation Settings: We consider the directors as targets. The quality of a director is high, if the number of his movies which have high qualities is not less than the number of his movies which have low qualities. The quality of a movie *(item I_k)* is considered high ($Q(I_k) = 1$), if the number of its ratings which are high is not less than the number of its ratings which are low. Otherwise, the quality of this movie is low ($Q(I_k) = 0$). The goal of successfully attacking one director is to change his quality to be opposite (complementary). Dishonest users will be created to give unfair ratings to movies in order to attack directors.

[1] Though we cannot guarantee 100% removal of noise, the presence of noise to a certain extent is acceptable as it helps in determining the robustness of trust models in a better way.

Evaluation Metric: The robustness of the reputation systems is evaluated using the *number of unfair ratings* required by attackers (dishonest users) to change the reputation of the target director. This is because, a more robust reputation system costs more efforts from attackers. Thus, if the attackers need more unfair ratings to change the targets' reputation, the reputation system is more robust against the attack. For each movie under a director, the actual number of unfair ratings provided by the dishonest userIDs is determined. These numbers from the individual movies under each director is then aggregated for the director. Based on the number of unfair ratings provided by the attacker to each director, the reputation systems are evaluated for their robustness.

5.2 Experimental Results

The primary targets are the first 20 directors out of the 40 directors. The attacks are conducted in consecutive and ascending order. The plots below are generated after running through reputation systems on the corrupted data generated by attack models, including RepBad, RepSelf and RepTrap.

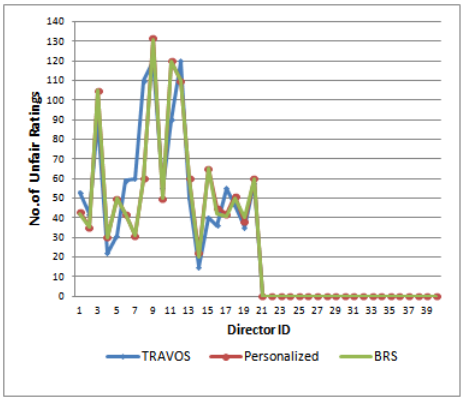
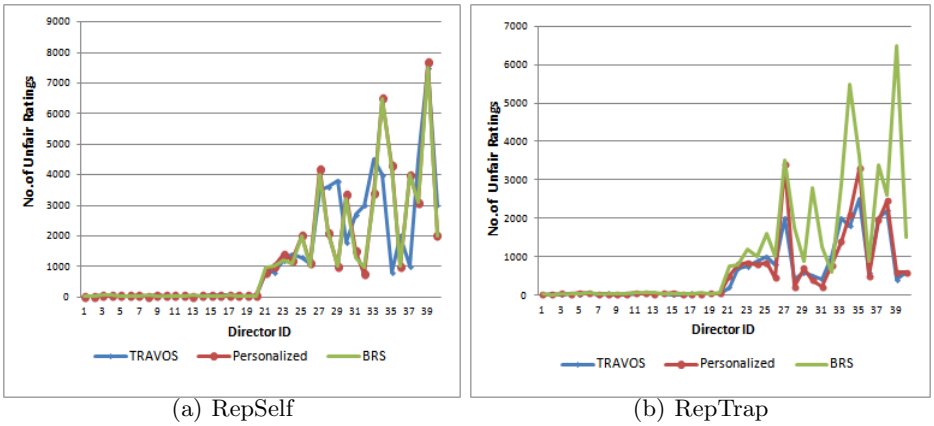


Fig. 3. Number of Unfair Ratings Needed by RepBad

Figure 3 shows the number of unfair ratings required by the RepBad attack to change the reputation of the target directors (directorID 1-20) when the BRS, TRAVOS and Personalized approaches are used, respectively. It can be seen that the number of unfair ratings required for a successful attack hits a maximum of 120 for TRAVOS and BRS, while it is 131 for the Personalized approach. However, the average number of unfair ratings required for a successful attack is 60 for all the three reputation system which signifies that all the three of them are equally robust against the RepBad attack.

Figure 4 shows the number of unfair ratings required by RepSelf and RepTrap. For RepSelf, the average number of unfair ratings needed to change a director's reputation is 2654, 2662, and 2607 for TRAVOS, Personalized and BRS, respectively. The maximum number of unfair ratings needed for a successful attack is 7700 for the Personalized approach. Thus the Personalized approach is found to be more robust against RepSelf attack than BRS and TRAVOS.



(a) RepSelf (b) RepTrap

Fig. 4. Number of Unfair Ratings needed by RepSelf and RepTrap

For RepTrap, BRS needs the maximum number of unfair ratings of 6500. The average number of unfair ratings required for a successful attack is 1102, 1182 and 2246 for TRAVOS, Personalized and BRS, respectively. Thus, BRS is more robust than TRAVOS and Personalized against the RepTrap attack.

6 Conclusion and Future Work

In this paper, we proposed a comprehensive testbed to evaluate the robustness and effectiveness of reputation systems. The proposed testbed performs thorough evaluation of the robustness of the various reputation systems against unfair rating attacks, a feature which is not available in the other existing testbeds. The testbed supports three different kinds of environments which makes it highly flexible for experimentation in a variety of settings. It employs simple as well as strategic attack models to effectively analyze reputation systems. Novel robustness metrics have also been proposed to accurately evaluate their robustness.

The testbed is composed of many components. This paper is a primary step in building the entire testbed. In this paper, we have presented two kinds of environments (simulated environment and real environment with simulated attacks). Implementation of the real environment with detected "ground truth" (third type of environment) is intended to be taken up as future work. The current implementation of the testbed employs various attack models (Constant, Camouflage, Whitewashing, etc) for evaluating some existing reputation systems employing different unfair rating detection approaches like BRS, iCLUB, TRAVOS, WMA, and Personalized. In the future, we plan to implement more challenging attack models and deception models to evaluate reputation systems. The sophisticated input and output interfaces are to be integrated in the later stage along with the development of the testbed. Scalability and usability issues are also to be considered for the better performance of the testbed. After the complete implementation of the testbed, the API for the testbed will be released so that the researchers can use it for their own purposes.

Acknowledgements. This work is supported by the NTU Start-up Grant and the MOE AcRF Tier 1 Grant.

References

1. P. Chandrasekaran and B. Esfandiari. A model for a testbed for evaluating reputation systems. In *Proceedings of the 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2011.
2. K. Fullam, T. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. Barber, J. Rosenschein, L. Vercouter, and M. Voss. A specification of the agent reputation and trust (ART) testbed: experimentation and competition for trust in agent societies. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 512–518. ACM, 2005.
3. A. Jøsang and J. Golbeck. Challenges for robust trust and reputation systems. In *Proceedings of the 5th International Workshop on Security and Trust Management (SMT), Saint Malo, France*, 2009.
4. A. Jøsang and R. Ismail. The beta reputation system. In *Proceedings of The 15th Bled Electronic Commerce Conference*, pages 41–55, 2002.
5. R. Kerr and R. Cohen. TREET: The trust and reputation experimentation and evaluation testbed. *Electronic Commerce Research*, 10(3):271–290, 2010.
6. E. Lim, V. Nguyen, N. Jindal, B. Liu, and H. Lauw. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 939–948. ACM, 2010.
7. S. Liu, J. Zhang, C. Miao, Y. Theng, and A. Kot. iCLUB: an integrated clustering-based approach to improve the robustness of reputation systems. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, volume 3, pages 1151–1152, 2011.
8. W. Teacy, J. Patel, N. Jennings, and M. Luck. TRAVOS: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.
9. A. Whitby, A. Josang, and J. Indulska. Filtering out unfair ratings in bayesian reputation systems. In *Proceedings of the 3rd International Joint Conference on Autonomous Agenst Systems Workshop on Trust in Agent Societies (AAMAS)*, 2004.
10. Y. D. Yafei Yang, Qinyuan Feng. Yan Lindsay Sun. Reptrap: A novel attack on feedback-based reputation systems. In *Proceedings of International Conference on Security and Privacy in Communication Networks (SecureComm'08)*, page 111. ACM, 2008.
11. B. Yu and M. Singh. Detecting deception in reputation management. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 73–80. ACM, 2003.
12. J. Zhang and R. Cohen. Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach. *Electronic Commerce Research and Applications*, 7(3):330–340, 2008.
13. L. Zhang, S. Jiang, J. Zhang, and W. K. Ng. Robustness of trust models and combinations for handling unfair ratings. In *Proceedings of the 6th IFIP WG 11.11 International Conference on Trust Management (IFIPTM)*, 2012.