Dynamic Sliding Window Model for Service Reputation

Xin Zhou
Department of Social Informatics
Kyoto University
Kyoto, Japan
xin@ai.soc.i.kyoto-u.ac.jp

Toru Ishida
Department of Social Informatics
Kyoto University
Kyoto, Japan
ishida@i.kyoto-u.ac.jp

Yohei Murakami
Unit of Design
Kyoto University
Kyoto, Japan
yohei@i.kyoto-u.ac.jp

***Abstract*—Reputation plays a crucial role in the success of e-commerce. In a commercial transaction, it is necessary to present reputation values of web services in a timely and a robust manner so as to counter the unfair ratings of malicious users. To address the time lag problem, most popular web sites use an averaging algorithm with fixed sliding windows; window size is constant and older ratings are dropped upon the arrival of new ratings. Herein, we propose a dynamic sliding window model that is capable of reflecting the reputation values yielded by the latest changes in services. Furthermore, we implement a statistical strategy to filter out unfair ratings by calculating the standard deviation of the ratings after transposing the two-dimensional linear window into the constant one-dimensional window by using linear regression. Experiments confirm the effectiveness of the proposed model, it outperforms the existing reputation system by 40% on average based on the 5 test cases examined, and also show that it can asymptotically converge to the underlying reputation value as ratings accumulate.**

***Keywords*-Reputation system; dynamic sliding window model; unfair rating; time lag;**

I. Introduction

The booming popularity of electric markets such as eBay, Amazon and Taobao, inevitably attracts malicious users intending to benefit from the systems illegally. As service providers, they can deliberately increase their reputation to mislead consumers who have no or little experience and consequently boost their transaction volume. A service consumer, when using an existing service or creating a new service by combining several existing services, may discover several functionally equivalent service providers with, it is assumed, different level of qualities. Usually, a service consumer has no or little direct experience with the candidate service providers. Thus decisions must be made as to which service provider will maximize the consumers benefit. Existing markets use reputation systems to allow the service consumer to evaluate the candidates.

The value of reputation systems has been well supported by both research and the success of reputation-centric e-commerce [1]. We define reputation as a public metric that is visible to all users, such as the reputation system operated by Amazon, eBay etc. Usually, service providers with high reputation achieve better average outcomes. A study of eBay conducted by Resnick et al. revealed that consumers were willing to pay 8% more to sellers with established reputations than to new sellers [2]. Unfortunately, open e-commerce systems can make service providers vulnerable to attack from malicious providers who collude with to deliberately give unfair ratings to specific competitors; examples include blackening the reputation of targeted service provider for personal gain [3]. Despite ongoing research on this problem, even advanced reputation models have difficulty in distinguishing the truthful changes caused by service updates from malicious ratings.

Beyond the unfair rating problem, another problem that has been ignored by most researchers is the time lag in reputation value. The simple and widely used reputation engine in current e-commerce is the averaging method, which is easily understood by both service providers and service consumers. Unfortunately, it cannot timely reflect the dynamic changes of service providers. In some e-commerce companies, a variation of the averaging algorithm, called the fixed sliding window algorithm (with window size of 30 days or so), is adopted to deal with this problem. Previous works [4], [5] on the unfair rating problem either assume the users have personal experience or ignored the time lag problem. A recent work by Wu et al. [6] proposed an olfaction-based algorithm that takes both problems into consideration. However, they assume that there must exist one fair rating at least in 10 consecutive ratings. In this paper, we relax the constraint and propose a dynamic sliding window model that addresses the unfair rating and time lag problems simultaneously. Here, we define the dynamic sliding window as a window whose size varies from time to time. The key issue is how to find the appropriate window size to correctly evaluate recently received ratings. We show that the proposal covers all online services, not just e-market services like eBay or Amazon, and so suits various services such as the service platform Language Grid [7] and Amazon Mechanical Turk.

Our main contributions are summarized as follows:

- A dynamic sliding window mechanism is proposed to eliminate the time lag problem; window size is changed to match the latest service changes. The algorithm continuously monitors the distribution of ratings to adjust the dynamic sliding window size. When the

978-1-4673-7281-7/15 $31.00 © 2015 IEEE
DOI 10.1109/SCC.2015.14

behavior of the service provider changes, the algorithm creates a new window in order to remove the influence of previous behavior so that the reputation can reflect the latest behavior of the service provider.

- The behavior pattern of a service provider is evaluated by Bayesian linear regression, which can learn from the observed ratings and update its parameters accordingly. Moreover, malicious ratings are detected based on probability theory. The proposed algorithm tags a continuous sequence of unfair ratings as a small probability event allowing its influence to be mitigated entirely.
- We refine the basic dynamic sliding window based on observations of e-market services to facilitate the identification of unfair ratings.

The rest of our paper is organized as follows. We discuss related work in Section II and introduce the background of the sliding window algorithm in Section III. The validity and performance of the model are evaluated in Section IV by comparing it to other algorithms in the literature. The research is rounded off with a conclusion in Section V.

II. Related Works

Since Resnick et al. [8] pointed out the issues dogging the reputation of web sites, such as eBay, various reputation systems based on feedback have been published to capture the reputation of service providers. A group of approaches calculate the reputation value by integrating several metrics, e.g., response time, availability, transaction volume, and demand. Reputation studied by [9], [10], [11] falls into this group. Those approaches fail to address the unfair rating problem, and do not distinguish the veracity of the feedback provider. Donato et al. [12] proposed new metrics to limit the impact that malicious users can have on network systems based on the EigenTrust model [13]. EigenTrust manages the reputation of peers by assigning global trust values and using this value to choose reliable peers for interaction; the system can isolate malicious peers from the network. Moreover, some reputation systems robust to malicious behavior have been proposed, such as the hierarchical and Bayesian inferred-trust model [4] and robust linear markov [14]. The first model is robust even in malicious environments by learning from all observed information, such from direct experience or third-parties. The latter updates the reputation in a hidden markov model based on new observations. In conclusion, those systems intend to manage the reputation of individual peers in a decentralized way based on the behavior of each individual. The reputation value is various from the rating experience of each individual. Hence, consumers with no or little experience can not infer the reputation of the service provider.

Jøsang and Ismail proposed a Bayesian reputation system based the beta probability distribution [15]. The reputation value is updated and learned from behavior history based on the beta distribution. The beta reputation model may suffer from coalition attacks wherein a group of malicious users modify the reputation value deliberately with fake feedback. The reputation algorithm used by Amazon is an averaging algorithm [16]. While it easy to understand by both service providers and service consumers, it also suffers heavily from unfair ratings and the time lag problem. To overcome the negative lag effect of the averaging method, Amazon averages the rating score in different time ranges, such as ratings in the latest 30 days. This mechanism can help users in evaluating the latest tendencies of service reputation. Although the fixed time window averaging algorithm can mitigate the lag effect of the averaging method to some degree, it is highly vulnerable to unfair ratings. That is, the fair ratings collected in the latest 3 days cannot mitigate the adverse effect caused by previous unfair ratings. Therefore, a model that not only represents the latest behavior of service providers, but that is also robust against unfair ratings is highly desired.

Wu et al. proposed a olfaction-based algorithm (OACR) for the time lag and unfair rating problems [6]. OACR allocates a fixed pre-defined weight value p to the latest received rating r_i when r_i is fair. Unfair ratings are detected when the absolute value of $R_{i-1} - r_i$ exceeds the preset threshold h, where R_{i-1} denotes previous reputation value. Olfactory response starts on unfair ratings, it first enters the perception stage for α iterations and continues to the fading stage if unfair ratings are still exist. Relative low weight is assigned to the perception stage, and high weights to the fading stage when an unfair rating is discovered. The key issue is how to determine the boundary between these two stages. OACR contains two algorithms OACR1 and OACR2. The only difference is that OACR2 uses preset weight values in its olfactory phase. The OACR model can detect unfair ratings and then mitigate their adverse effect because it assumes that at least one in 10 consecutive ratings is fair. If OACR finds the ratings to be fair, it moves to the fading stage, in which high weight is assigned to the latest received ratings.

Our proposed dynamic sliding window model is built on the observation that malicious users always intend to increase or decrease the reputation of a service provider deliberately. Usually, they rate every transaction they conduct. As a result, the ratio of transactions that have been rated is relatively high compared to the behavior of normal users. The dynamic sliding window algorithm also captures the dynamic changes made by the service provider, and moves to a new window when the behavior pattern is different from the previous one.

III. Dynamic Sliding Window Model

Fixed sliding window reputation model is popular used in commercial system, however, its hard to decide the appropriate window size for good performance. That is, using wide sliding window will suffer heavily from time lagging, while narrow sliding window is vulnerable to unfair rating attacks. The Figure 1 represents this dilemma situation. This leads us to design a dynamic sliding window in service computing domain.

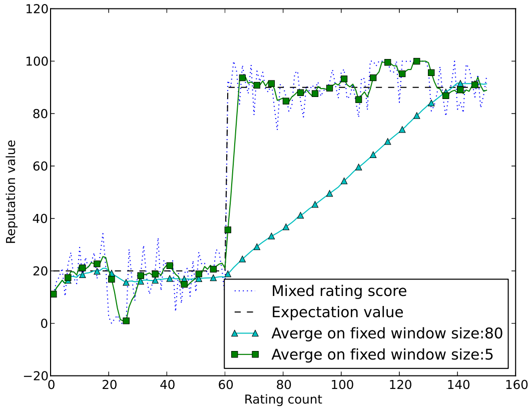


Figure 1. The dilemma of fixed sliding window: when given a relatively wide window, it suffers heavily from the time lagging; otherwise, a narrow window will be vulnerable to unfair rating.

Whitby et al. assume that ratings provided by different raters on a given service will follow more or less the same probability distribution. When service provider S_i changes its behavior, it is assumed that all honest raters who interact with S_i will change their ratings accordingly [17]. The model proposed herein adopts the same assumption that in the dynamic case, the rating distribution changes with a change in service provider behavior. This assumption is the foundation of our dynamic sliding window model.

A. Basic Dynamic Sliding Window Model

A popular approach to defining reputation models is to base, in part, the current reputation value on the reputation value generated in the previous time slice [6], [18]. We define the following formula to aggregate the most recent received rating r_n, given the rating r_n is the n-th rating in current window W_c:

$$R_n = \rho f(n) + (1-\rho)R_{n-1}, and\ f(n) = k \cdot n + C. \quad (1)$$

where ρ is the weight assigned to the predictive reputation value of latest received rating in window W_c. $f(n)$ is the linear regression function on ratings. When the behavior of a service changes, current reputation value R_n will be updated in a new window, therefore current window may be updated as the new window. For example, the ratings received by service provider s are denoted as $r_1, r_2, ...r_n$, and we assume they are received in the same window W_c. However, ratings r_{n-2}, r_{n-1}, r_n need to be identified as fair or not because their probability deviates relatively a lot from the standard deviation. When new rating r_{n+1} is received, and the dynamic sliding window algorithm has determined that it should move to new window W_n because the service provider has changed its behavior pattern, then all prior pending ratings r_{n-2}, r_{n-1}, r_n will be used as the initial ratings for window W_n, and the algorithm is applied to the new window. Index i is assigned as 4 in the new window, because it is the 4-th rating received in the new window. Consequently, weight ρ is refreshed based on the new window. Thus, the dynamic changes and reputation calculation can be evaluated in one equation. The scheme for determining the window moves and ρ is addressed in the next subsection.

B. Filtering Unfair Rating based on Probability Theory

In the dynamic sliding window model, the key issue is to detect the behavior pattern changes of service providers and identify the unfair ratings of malicious users. The dynamic property of the model can adapt to the latest behavior of a service. When aggregating the latest received rating r_i, the current ratings distribution is used to determine window size w and weight ρ. First, we define window size in our model. Window size w is the maximum number of ratings in window w following the same distribution with fewer than WND consecutive accumulated unfair ratings. The window is terminated when the count of newly received accumulated unfair ratings exceeds WND; subsequent ratings are allocated to a new window.

Our proposal uses the Bayesian model to solve the linear regression problem. We limit the linear function to a two-dimensional function, that is, $f(x) = kx + C$. For a given sequence of ratings $r_1, r_2, ..., r_n$, the Bayesian model can find the maximum likelihood of k, C based on the ratings. That is, by maximizing the posterior distribution:

$$p(k, C|\mathbf{r}) \propto p(\mathbf{r}|k, C)p(k, C). \quad (2)$$

parameter k, C can be derived based on the observed ratings $\mathbf{r}$. If the prior distribution of k, C follows a normal distribution, according to [19], the log of posterior distribution takes the form:

$$ln\ p(k, C|\mathbf{r}) = -\frac{\beta}{2}\sum_{i=1}^{n}\{r_i - f(i)\}^2 - \frac{\alpha}{2}[k, C]^T[k, C] \quad (3)$$

Maximizing the posterior distribution with respect to k, C is equivalent to the minimization of the sum-of-squares error function plus a parameter regularization term, where $\beta = 1/\sigma^2$ is the noise precision parameter and α is the regularization parameter. The last part in the above equation is mainly to limit the parameter k, C in order to avoid the over-fitting problem. By partial differentiation on k and C respectively, the parameter can be derived as:

$$k = \frac{\partial ln\ p(k, C|\mathbf{r})}{\partial k}; C = \frac{\partial ln\ p(k, C|\mathbf{r})}{\partial C} \quad (4)$$

When the distribution of ratings changes with time, the parameters can learn from the changes and so suit the latest observations. To detect unfair ratings, we eliminate rating r_n with the linear function $f(i)$ in its window, so the span value of ratings $f(1)-r_1, f(2)-r_2, ..., f(n)-r_n$ is converted into the constant reputation situation. The standard variance of $f(1)-r_1, f(2)-r_2, ..., f(n)-r_n$ is calculated and used to determine the occurrence probability, p_{n+1}, of next rating r_{n+1}. If p_{n+1} exceeds threshold $sigma$, the model records index $n+1$ as the new rating that has high probability of indicating the start of a new window. We observe the next several ratings, if their accumulated probability exceeds the maximum threshold then the ratings from $n+1$ are abnormal in the current window. At this point, those ratings are either unfair ratings or fair ratings, and should be mitigated or allocated to another window. Here, we first adopt the assumption that the system security strategy ensures that there is at least one fair rating in WND consecutive ratings. That is, when the detected abnormal ratings count exceeds WND, the conclusion can be drawn that the rating changed because the quality of the service provider was updated. However, we can omit this assumption by introducing the transaction volume variable in next section.

Weight of the predictive reputation value for r_n: assuming that the trend underlying the ratings is given by $f(x) = kx + C$ and $f(i)$ is the expectation value of rating i. Using the property of linear regression on ratings, as $i \to \infty$, the error between predicted and the real reputation value R_i approaches 0 in the same window. That is:

$$\lim_{i \to \infty} |f(i) - R_i| \to 0 \tag{5}$$

Similar to the averaging algorithm, as the number of accumulated ratings increases, the variation in rating has less impact on the current reputation value. The weight of the latest received rating is given by this formula:

$$\rho = e^{-|f(i)-r_i|/w} \tag{6}$$

where w the window size of current window, and $f(i)$ is the predicted value of r_i.

C. Rating Ratio Based Dynamic Sliding Window Model

The classic approach to distinguish quality updates from unfair ratings uses a fixed number of consecutive ratings. There are two reasons why we use the rating rate on transaction volume to facilitate this process: 1) Some researchers argue that raters cannot express their opinion accurately if only numeric rating scores can be assigned [20]. They proposed some methods to facilitate the collection of opinions. Here, the transaction volume of an agent is used to verify the ratings given by raters. For example, if the transaction volume is proportional to ratings, then we have a solid belief that the ratings are fair. However, if the transaction volume is excessive relative to the rating value, then the latest received ratings may be unfair and should be given low weight. 2) Observations of the data of Taobao, Amazon and Hotels.com showed that not all customers have the time or interest to input their ratings.

In fact, nearly 50%∼60% of EBAY customers did not leave a rating for various reasons [21]. Which means transaction volume was much larger than the number of received ratings. Some customers did not express their opinion of the service, but their opinion can be discerned from the transaction volume. Normal customers, those who pay for the service, have little interest in entering their ratings unless they are extremely satisfied or unsatisfied. On the contrary, malicious users tend to seize every opportunity to increase or decrease the reputation of the interacted service deliberately. We try to facilitate the detection process for unfair ratings by examining the ratio of rating number to transaction volume. We denote the improved DSW algorithm as Rating Ratio based Dynamic Sliding Window algorithm (RRDSW). Given service provider s_i, with rating scores from 0 to 60, the ratio $r_r = N_r/T_r$ is around 50%∼60%, where N_r and T_r are the number of ratings received and the transaction volume in a given time span, respectively. After a service update, the difference between the previous ratio r_r and the new ratio $r_{r+1} = N_{r+1}/T_{r+1}$ will stay within the range $THRT$. The reason is that the users cannot change their behavior mode immediately. Based on this observation, we can better identify unfair ratings.

The results in figure 2(a) are for the improved DSW algorithm, RRDSW, and show that it has better performance than OACR in all test cases

IV. Evaluation

This section presents a series of numerical experiments designed to evaluate accuracy of our model in a comparison with the state-of-the-art competitor [6].

A. Experimental Environment

The best way to evaluate a reputation algorithm is use actual reputation values. However, no general test set is available to evaluate the reputation system. Here, we adopt the popular method of using simulations. In paper [6], the authors use a set of cases and each case has 150 expectation values and 150 rating scores. We adopt the same simulation environment. In that paper, expectation values are generated by expectation functions such as linear, quadratic, sinusoidal, exponential and logarithmic function, while the rating scores are generated by the expectation function plus a median distribution of fair ratings. We use normal distribution $N(\mu, \sigma^2)$ with parameter $\mu = 0, \sigma = 6.5$ in the tests, the noise precision β can be learnt from the observed ratings, and α is predefined as 0.005. To simulate the unfair rating, some of the fair ratings are replaced with unfair rating. In the experiments, the distribution of the unfair rating is restricted as at most 9 consecutive unfair ratings must be followed by 1 fair rating. The test data set is generated by mixing fair ratings with unfair ratings. The mean absolute error (MAE) is calculated to evaluate the accuracy of the algorithm.

$$e = \frac{(\sum_{i=1}^{t} |x_i - E(x_i)|)}{t} \tag{7}$$

where t is the total set of ratings, x_i is the $i-th$ rating, and $E(x_i)$ is the expectation value of $i-th$ rating.

B. Design of Test Cases

The proposed dynamic sliding window algorithm (DSW) is implemented and compared with the novel method called Olfaction-based Algorithm (OACR) [6] on various data cases. The proposed model and OACR are evaluated on the following patterns, which are widely used in the related works to reflect the different behaviors of service providers:

1) **Constant:** This group of service providers either behaves consistently with high reputation value or low reputation value. The providers with high reputation value are rational, while those with low reputation may always provide low quality service to take advantage of the consumer [22], [23].
2) **Linear and pairwise:** The service providers change their strategies halfway, either from high reputation value to low in order to rip off the attained reputation [24], [25], or they learn from their previous mistakes and ameliorate their behavior [6], [22], [23], [26].
3) **Sinusoidal:** This type of service provider acts in a random manner or deliberately creates and then abuses the reputation value periodically [23]. The quality of the service may degrade with time and the service provider updates it periodically.

The above behavior patterns are mentioned in the literature, but no concrete test cases were described for experiments. We thus design the following test cases for benchmark tests. Each test case is executed 10 times and the average value is compared in Figure 2(a). In the detailed case figure, we did not draw the RRDSW algorithm line as it basically mirrors that of DSW.

- Test case 1: $y = 50, 1 \leq x \leq 150$. Figure 2(b) plots the results of the three algorithms under constant reputation mode. The results prove that the averaging algorithm works best if ratings are consistently fair with a small number of unfair ratings.
- Test case 2: $y = 0.6x, 1 \leq x \leq 150$. In this test case, the reputation value of a service provider is updated in linear mode. When a service provider realizes that more benefit can be derived from the reputation value, it continues updating its service quality periodically. Figure 2(c) plots the results of the three algorithms under linear reputation mode with $y = 0.6x$. The results are the average of 10 tests in Figure 2(a).
- Test case 3: Reputation value is changed midway. The expectation function is:

$$y = \begin{cases} 20, & 1 \leq x \leq 60 \\ 90, & x > 60. \end{cases} \tag{8}$$

This test case demonstrates the behavior pattern that the service providers learn from their experience and update their service quality halfway. The opposite behavior pattern is also possible where the service providers abuse their established reputations. 10 tests were performed and the average, maximum, and minimum values are plotted in Figure 2(a). Detailed reputation values for each algorithm are illustrated in Figure 2(d).

- Test case 4: Rating distribution is changed midway. The expectation function is:

$$y = \begin{cases} 50, \sigma = 6.5, & 1 \leq x \leq 60 \\ 50, \sigma = 18.0, & x > 60. \end{cases} \tag{9}$$

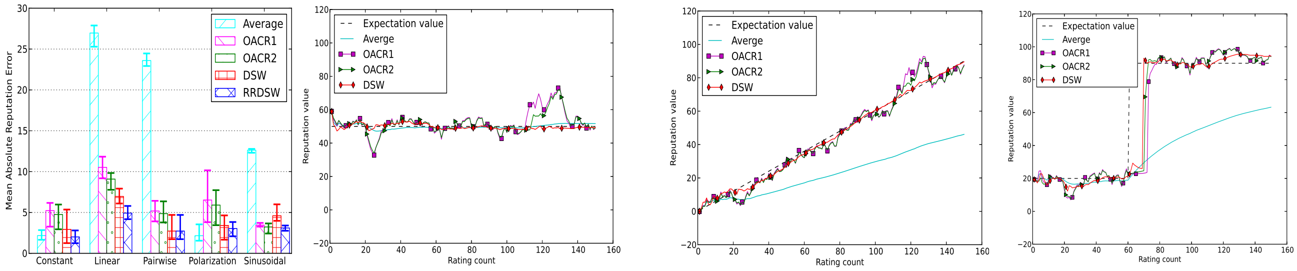
Some service providers update the quality of their services successfully, while updates are not so successful. This means the update is accepted by some people, but not everyone. A characteristic of this the preponderance of 5-star and 1-star ratings. In this test case, the variance of ratings changed from 6.5 to 18 midway. The results of the three algorithms are illustrated in Figure 3(a). The σ value for DSW model, denoted by the red dashed line, verifies that the proposed model dynamically adapted its threshold value in response to the change in the distribution of ratings. The original ratings are plotted by the blue dashed line. In the figure, the observed rating values fluctuate after rating count 60.

- Test case 5: $y = 20.0 * sin(x/60.0) + 30.0$, $1 \leq x \leq 500$. The test case here examines the sinusoidal behavior of some service providers. The performance of the three algorithms is shown in Figure 3(b). We observe that OACR1 adapted better than OACR2 because the weight changes with the rating count in the olfactory phase. The proposal, DSW, is better than other algorithms because after it set a new window, it approached the expected value as more new ratings were received.

However, the results in figure 2(a) are worse than those yielded by OACR, and we address this problem in the next subsection by introducing the transaction volume parameter.

C. Unfair attack model

TO simulate the unfair rating attack, we use the similar attack model in [6]. The attack appears with the following



(a) MAE reputatin (b) Constant: $y = 50$ (c) Linear: $y = 0.6x$ (d) Pairwise reputation function

Figure 2. Results of the average algorithm, olfaction-based algorithm (OACR) and dynamic sliding window algorithm (DSW) compared on b) Constant reputation value c) Linear reputation value and d) Pairwise reputation function. Unfair rating attacks from rating index 20 to 26, intervals [110, 120) and (120, 130].

Table I
MEAN ABSOLUTE ERROR OF REPUTATION EVALUATION FOR MODELS UNDER COALITION ATTACK.

Cases	Average	OACR1	OACR2	RRDSW
Constant	**0.80 ± 0.12**	3.85 ± 0.25	3.62 ± 0.21	**0.80 ± 0.14**
Linear	75.42 ± 0.40	4.40 ± 0.25	4.42 ± 0.20	**1.25 ± 0.25**
Pairwise	18.19 ± 0.19	5.61 ± 0.20	5.02 ± 0.26	**1.58 ± 0.23**
Sinusoidal	12.30 ± 0.11	4.05 ± 0.27	3.86 ± 0.24	**2.73 ± 0.24**

pairwise function:

$$y = \begin{cases} y_t - 20, & 20 \le x \le 26 \\ y_t + 20, & 110 < x < 120 \; and \; 120 < x < 130. \end{cases} \tag{10}$$

where y_t is the fair rating function in the above test cases. Each test is evaluated to examine the robustness of the reputation model.

D. Performance Evaluation

1) Convergence: In order to reduce the algorithm computational complexity from $O(w)$ to constant level, where w is the window size, we introduce the concept of cutting the window length at the point at which DWS becomes stable. The following convergence tests are conducted:

i. The averaging algorithm and the Olfaction-based Algorithm (OACR) [6], both OACR1 and OACR2, are evaluated against the dynamic sliding window (DSW) algorithm in terms of mean absolute error metric by formula (7). Observing a large data set of ratings will allows us to determine whether the reputation system can converge to a fixed mean absolute error or not.
ii. The mean absolute error against rating count is plotted to determine the convergence rate of the four algorithms.
iii. We focus on the constant reputation value scenario, with a normal distribution $N(0, 6.5)$ of fair ratings. The result is shown in Figure 3(d).

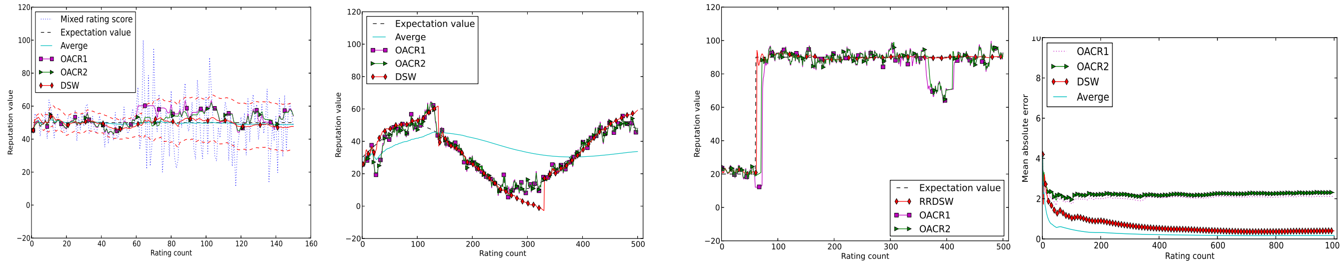
In Figure 3(d), despite the fluctuation in DSW at the beginning, DSW converged quickly to the averaging algorithm in the above environment. Ideally, the mean absolute error of DSW algorithm and averaging algorithm should approach 0 when the ratings approach infinity. Neither OACR1 nor OACR2 matched this property. The reason is that DSW algorithm uses a linear regression algorithm to detect the real reputation values that underlie the ratings. Thus, as more observations are accumulated, the error between the expectation reputation value and the calculated value approaches 0. OACR allocates a fixed weight to the latest rating, as the rating follows $N(0, 6.5)$, and so is not assured of converging to the underlying reputation value. The mean absolute error of reputation values at rating index 1000 are 0.16, 2.27, 2.50 and 0.48 for averaging algorithm, OCAR1, OACR2, and DSW, respectively. The results prove that our proposed dynamic sliding window model, and the averaging algorithm, can accurately converge to the underlying real reputation value.

2) Robustness: As previously mentioned, the OACR algorithm depends on the assumption that there must be at least one of 10 consecutive ratings must be fair. We lift this constraint by introducing the rating ratio based dynamic sliding window algorithm. We use the pairwise reputation function to confirm robustness as it is widely used in the literature. The scenario is similar to the previous tests except that more than 10 consecutive unfair ratings were possible. The result on the pairwise case is shown in 3(c) indicating RRDSW can mitigate the unfair ratings in rating period [360, 400] effectively. While OACR2 can partially mitigate the influence of unfair ratings from 360 to 370, it enters the fading stage after rating index 370, and thereafter misjudges the unfair ratings.

Table I presents the mean and standard deviation (over 10 tests) for mean absolute error of reputation evaluation on 4 test cases. We find the RRDSW is robust against coalition attacks and outperforms the OACR algorithm by 62% on average.

E. Analysis and Discussion

The results shown in Figure 2(a), show the proposed basic DSW model outperforms all OACR variants in almost all test cases. Figures 2 and 3 clearly show that the plots of DSW



(a) Dynamic rating distribution (b) Sinusoidal reputation function (c) Robustness (d) Convergence

Figure 3. Results of the average algorithm, olfaction-based algorithm and dynamic sliding window algorithm compared on a) Dynamic rating distribution, the ratings towards polarized on the service. b) Sinusoidal reputation value. Unfair rating attacks from rating index 20 to 26, intervals [110, 120) and (120, 130]. . c) Robustness of OACR and DSW. Unfair rating attacks from rating index 360 to 400. d) Convergence comparison: DSW converges more closely to the real reputation value than OACR.

are closer to the expected values than the OACR variants. This is because the DSW algorithm is based on the following theoretical advantages:

1) The averaging algorithm. The results from the first test case proved that the averaging algorithm has better performance than the others given constant reputation values. It is understandable because the averaged value can neutralize the noise of fair ratings and the impact of unfair ratings. In the extreme situation that unfair ratings predominate, the averaging algorithm yields poor performance.
2) Bayesian linear regression. The Bayesian theory offers a sound probability foundation for choosing parameters k, C. By maximizing the posterior distribution with respect to k, C on fair ratings X will yield a plausible value for k, C [19].
3) Probability theory. After subtracting the fair ratings with their expected rating calculated by linear function $f(n)$, the variance σ of the data is calculated. For newly arrived rating r_n, if its occurrence probability less than $\sigma(\approx 0.68)$, DSW continues to accumulate the following ratings. If the accumulated occurrence probability P_a exceeds 0.01, which indicates small probability events in the same direction (all the ratings below or above 0), DSW begins to mitigate the unfair ratings. Probability P_a is changed with the distribution of ratings. This process is different from the OACR algorithm in [6], in which the author uses a predefined threshold value for detecting unfair ratings, and so is robust against dynamic changes in the environment.

However, the basic DSW is inferior to the OACR variants in the sinusoidal test case. In Figure 3(b), DSW assumes the service provider is in its current behavior pattern when the algorithm cannot distinguish the changes by service updates from those by unfair ratings. As mentioned in Section III, in order to distinguish the changes caused by service updating from those by unfair ratings, a constraint parameter, WND, is introduced in our model as well as OACR [6]. In the WND ratings, DSW assumes the latest received ratings are unfair and eliminates the impact of this suspicious period. Therefore, in Figure 2(a), the rating ratio based dynamic sliding window algorithm can improve the result to equal the performance of the OACR algorithm in the worst case. However, DSW has the advantages of convergence to the real reputation value and robustness against coalition attacks.

Figure 3(d) shows that DSW stabilizes at rating index 100. That means, by limiting the maximum sliding window size to the value at which DSW becomes stable, DSW can run in constant time. This will suit environments that are sensitive to computation complexity.

V. Conclusions

This paper tackled the time lag and unfair rating problems in reputation systems by introducing a new algorithm. In the proposed algorithm, the distribution of ratings is continuously monitored to dynamically resize the sliding window. When the service provider changes its behavior, the algorithm sets a new window to remove the influence of previous behavior such that the latest reputation reflects the latest quality of the service provider. In order to facilitate the detection of unfair ratings, we improve the basic dynamic sliding window based on the observation that 50%∼60% of consumers fail to rate their transactions. This mechanism lifts the restrictive assumption made by the existing algorithm that a fixed number of consecutive ratings must be observed before unfair ratings can be identified.

Simulations showed that our algorithm was more accurate than published algorithms and a method used by current commercial services The proposed algorithm adapts itself to dynamic changes on the rating distribution unlike the existing algorithm that uses a fixed threshold for unfair rating detection. Furthermore, by introducing the ratio of rating number to transaction volume as an indicator, the improved algorithm outperformed the compared algorithm by 40% on average and was proven to be more robust than the compared algorithm by 62% under coalition attacks.

Acknowledgments

This research was partly supported by a Grant-in-Aid for Scientific Research (S) (24220002, 2012-2016) from Japan Society for Promotion of Science (JSPS).

References

[1] J. A. Livingston, "How valuable is a good reputation? a sample selection model of internet auctions." *Review of Economics and Statistics*, vol. 87, no. 3, pp. 453–465, 2005.

[2] P. Resnick, R. Zeckhauser, J. Swanson, and K. Lockwood, "The value of reputation on ebay: A controlled experiment," *Experimental Economics*, vol. 9, no. 2, pp. 79–101, 2006.

[3] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," in *Proceedings of the 2nd ACM conference on Electronic commerce*. ACM, 2000, pp. 150–157.

[4] W. Teacy, M. Luck, A. Rogers, and N. R. Jennings, "An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling," *Artificial Intelligence*, vol. 193, pp. 149–185, 2012.

[5] S. Liu, J. Zhang, C. Miao, Y.-L. Theng, and A. C. Kot, "i-club: An integrated clustering-based approach to improve the robustness of reputation systems," in *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, 2011, pp. 1151–1152.

[6] Y. Wu, C. Yan, Z. Ding, G. Liu, P. Wang, C. Jiang, and M. Zhou., "A novel method for calculating service reputation," *Automation Science and Engineering, IEEE Transactions on*, vol. 10, no. 3, pp. 634–642, 2013.

[7] T. Ishida., "Language grid: an infrastructure for intercultural collaboration." *Proc. IEEE/IPSJ Symp. Applications and the Internet (SAINT ',06)*, pp. 96–100, 2006.

[8] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Communications of the ACM*, vol. 43, no. 12, pp. 45–48, 2000.

[9] P. Xiong, Y. Fan, and M. Zhou, "Web service configuration under multiple quality-of-service attributes," *Automation Science and Engineering, IEEE Transactions on*, vol. 6, no. 2, pp. 311–321, 2009.

[10] E. Al-Masri and Q. H. Mahmoud, "Discovering the best web service," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 1257–1258.

[11] Z. Shen and N. Sundaresan, "Reprank: reputation in a peer-to-peer online system," in *Proceedings of the 22nd international conference on World Wide Web companion*. International World Wide Web Conferences Steering Committee, 2013, pp. 163–164.

[12] D. Donato, M. Paniccia, M. Selis, C. Castillo, G. Cortese, and S. Leonardi, "New metrics for reputation management in p2p networks," in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*. ACM, 2007, pp. 65–72.

[13] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th international conference on World Wide Web*. ACM, 2003, pp. 640–651.

[14] X. Wang, L. Liu, and J. Su, "Rlm: A general model for trust representation and aggregation," *Services Computing, IEEE Transactions on*, vol. 5, no. 1, pp. 131–143, 2012.

[15] A. Jøsang and R. Ismail, "The beta reputation system." in *Proceedings of the 15th bled electronic commerce conference*, 2002, pp. 41–55.

[16] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision." *Decision Support Systems*, vol. 43, no. 2, pp. 618 – 644, 2007.

[17] A. Whitby, A. Jøsang, and J. Indulska, "Filtering out unfair ratings in bayesian reputation systems." in *Proc. 7th Int. Workshop on Trust in Agent Societies, vol. 6*, 2004.

[18] B. Khosravifar, J. Bentahar, and A. Moazin, "Analyzing the relationships between some parameters of web services reputation," in *Web Services (ICWS), 2010 IEEE International Conference on*, July 2010, pp. 329–336.

[19] C. M. Bishop, *Pattern recognition and machine learning.* New York: springer, 2006, vol. 1.

[20] H. Ramn, R. Centeno, and M. Fasli., "From blurry numbers to clear preferences: A mechanism to extract reputation in social networks." *Expert Systems with Applications*, vol. 41, no. 5, pp. 2269–2285, 2014.

[21] L. Cabral and A. Horiaçosu, "The dynamics of seller reputation: Evidence from ebay*," *The Journal of Industrial Economics*, vol. 58, no. 1, pp. 54–78, 2010.

[22] M. Zaki and A. Bouguettaya, "Rateweb: Reputation assessment for trust establishment among web services." *The VLDB Journal – The International Journal on Very Large Data Bases*, vol. 18, no. 4, pp. 885 – 911, 2009.

[23] G. Vogiatzis, I. MacGillivray, and M. Chli., "A probabilistic model for trust and reputation." in *In Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, vol. 1-Vol.1. International Foundation for Autonomous Agents and Multiagent Systems, 2010, pp. 225 – 232.

[24] L. Xiong and L. Liu., "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities." *Knowledge and Data Engineering, IEEE Transactions on*, vol. 16, no. 7, pp. 843–857, 2004.

[25] S. Wang, Z. Zheng, Q. Sun, H. Zou, and F. Yang, "Evaluating feedback ratings for measuring reputation of web services," in *Services Computing (SCC), 2011 IEEE International Conference on*, July 2011, pp. 192–199.

[26] J. Sabater and C. Sierra., "Regret: A reputation model for gregarious societies." in *Fourth workshop on deception fraud and trust in agent societies, vol. 70*, 2001.