Computer Science and Information Systems 17(2):487–507 https://doi.org/10.2298/CSIS191122007O

A Robust Reputation System using Online Reviews*

Hyun-Kyo Oh[1], Jongbin Jung[2], Sunju Park[3], and Sang-Wook Kim[4]

[1] Samsung Electronics
hyunkyo.oh@samsung.com
[2] Stanford University
jongbin@stanford.edu
[3] Yonsei University
boxenju@yonsei.ac.kr
[4] Hanyang University
wook@agape.hanyang.ac.kr

Abstract. Evaluating sellers in an online marketplace is an important yet non-trivial task. Many online platforms such as eBay and Amazon rely on buyer reviews to estimate the reliability of sellers on their platform. Such reviews are, however, often biased by: (1) intentional attacks from malicious users and (2) conflation between a buyer's perception of seller performance and item satisfaction. Here, we present a novel approach to mitigating these issues by decoupling measures of seller performance and item quality, while reducing the impact of malignant reviews. An extensive simulation study shows that our proposed method can recover seller reputations with high rank correlation even under assumptions of extreme noise.

Keywords: reputation, reviews, attacks.

1. Introduction

One of the major challenges for online marketplaces such as eBay.com is that of accurately measuring the reliability of sellers on their platform [3, 10, 14, 17, 25]. The most common implementation of this task takes the form of a reputation system in which buyers are tasked with evaluating their interaction with sellers on some common scale (e.g., a 5-star rating [15, 26–29].) These ratings are then aggregated and presented to future buyers as a proxy for a seller's quality. While such measures have substantial influence on buyer behavior and overall marketplace dynamics, determining how reliable and robust they are to bias is still an open question. The impact of this problem is only getting larger as the presence and significance of online platforms in society increase. With the addition of more complex multi-agent systems and multi-faceted online marketplaces which often span global economies—such as Uber or AirBnB, the question of whether a reputation rating system can be robust to corruption and bias, and what the framework for measuring such robustness could be is becoming more important than ever. [2] Our work aims to address this question by first introducing a novel method for reliably measuring reputation from potentially corrupted ratings, whether maliciously intended or not, and subsequently proposing a general simulation model for quantifying the robustness of various reputation measurement strategies.

* Corresponding author: Sang-Wook Kim (wook@agape.hanyang.ac.kr)

Reputation systems aim to leverage the wisdom of crowds [11, 18], assuming that all participants understand and agree upon the common goal of transparently measuring the quality of a seller. This assumption, however, is flawed, as pointed out by recent studies that identify adversarial behavior in buyer ratings [28]. Various types of cheating behavior have been identified, both in theory and in practice, along with recommendations for how to account for such behavior in aggregating reviews [5, 7, 10, 19, 20, 23, 24].

In addition to the threat of malicious reviews, another—perhaps more subtle—issue for reputation systems is that the common goal of a review may not be immediately obvious to reviewers (e.g., buyers.) By evaluating an interaction, the reviewer is necessarily conflating multiple aspects of a transaction, only one of which is the seller's performance. For example, a buyer could be extremely satisfied with an item, but find the seller's competence in communication and execution problematic. In such a case, the buyer's scoring of the transaction, whether high or low, will be at best a biased measure of seller performance and item quality. We address this issue by first modeling a buyer's review as a combination of evaluating two factors: seller performance and item quality. Taking advantage of the fact that multiple sellers offer similar items and that one seller often offers multiple items, we propose an iterative method, which we call RATING SEPARATION, for teasing out each of the two factors that confound buyer reviews.

In order to create a comprehensive and robust reputation system, we further propose INTEGRITY WEIGHTING, a novel approach to mitigate the adverse effects of dishonest reviews. As the name suggests, the idea is to estimate the level of trustworthiness of each review, based on theories of buyers' cheating behavior. Each review is subsequently re-weighted according to the estimated trustworthiness. We show that, while existing approaches mainly focus on a subset of possible attacks, INTEGRITY WEIGHTING is robust against a large pool of attack types and patterns that have been identified in literature.

Finally, we develop a simulation framework for evaluating the efficiency of a reputation system in online marketplaces. Compared to existing marketplace simulations [4, 12, 16], our model allows for the conflation of multiple factors in a buyer's review. Using this framework, we evaluate the proposed reputation system and find it to be more robust and reliable in measuring seller reputation compared to existing methods.

In summary, contributions of this paper are threefold. First, we propose RATING SEPARATION, a method for disentangling, from a single score given by a buyer, the ratings for a seller and the item sold. Second, we present INTEGRITY WEIGHTING, a scheme for mitigating the risk of malignant agents on the platform. Third, we present a novel and comprehensive simulation approach for evaluating various policies and systems on an online marketplace platform. Using this framework, we are able to evaluate the practical efficacy of complex reputation systems—such as the ones we propose here. Additionally, this simulation framework allows us to better investigate existing methods, and identify their strengths and potential shortcomings.

2. Related work

Many online platforms have their own reputation systems evaluating the reliability of products or sellers by aggregating buyer reviews. However, these reputation systems are intrinsically vulnerable to malicious users who intentionally give unjustified reviews to

products or sellers. Numerous studies have been conducted to improve the robustness of reputation systems by mitigating the influence of such malicious users.

Online platforms can be largely categorized as either single-agent or multi-agent systems. In a single-agent system, a single provider curates a collection of products—such as movies on IMDb.com—and users are tasked with rating their experience with each product. Since there is a single provider, buyer ratings have a clear mapping to each product, and buyers are often limited to rating each product at most once. Existing studies of such single-agent systems give attention to eliminating anomalous (potentially malicious) ratings based on statistical analysis of the distribution of buyer ratings or ranking/grouping users based on their rating patterns in order to derive the weighted mean of ratings given by the users [6, 8, 21, 22].

In a multi-agent system, numerous sellers can provide multiple goods and services—as on Amazon.com. Unlike single-agent systems, in a multi-agent system, buyers can evaluate both the seller and their product. And as a consequence of repeated interactions, it is possible for buyers to provide more than one rating to the same seller, over a variety of products.

Both single-agent and multi-agent systems share the same goal of diminishing the risk of malicious ratings. However, due to the different graph structure between buyers and sellers being more complex, state-of-the-art strategies that work well for single-agent systems are often insufficient for multi-agent systems. One of the often discussed challenges for multi-agent systems is that of identifying malicious buyers (attackers) who—in coordination with associated sellers—aim to artificially manipulate the reputation of target sellers. This can take the form of either increasing the reputation of partnered sellers, or decreasing the reputation of competing sellers. As online platforms become more commonplace and complex, deceptive rating strategies have also evolved, making it harder to identify attackers. In response, most existing studies focus on not only filtering out statistically insignificant ratings but also finding suspicious buyer-seller relationships by detecting malicious behavioral patterns in their rating systems [1, 5, 6, 9, 13, 15, 20, 24, 26–30]. A less discussed issue for rating systems in multi-agent systems, however, is that of confounded ratings. As discussed in the previous section, benign users can still corrupt a reputation system by conflating their evaluation of a seller and a product. While previous studies deal with the issue of malicious users, we have yet to find studies that address the subtle, yet important, issue of confounded buyer ratings. In addition to addressing the more traditional concerns of malicious ratings, our approach aims to achieve robustness against the threat of such ambiguous ratings as well.

3. Proposed methods

Overall, we propose RATING SEPARATION AND INTEGRITY WEIGHTING (RS&IW), a system for retrieving reputations that are robust to confounded ratings and adversarial behavior. The first part of the system, RATING SEPARATION, decomposes the possibly confounded rating into individual components of seller and item ratings. The second part of the system, INTEGRITY WEIGHTING, extends the work of Oh et al. [23] to quantify the trustworthiness of each rating. In the following sections, we present the details of each method.

3.1. Decoupling ratings

A common goal for online marketplace providers is to identify the reputation, trustworthiness, and quality of active sellers and items that are traded on their platform. Formally, given a set of sellers $S = \{s_1, s_2, \ldots\}$ and items $M = \{m_1, m_2, \ldots\}$, a platform provider—such as eBay.com—would like to recover some function $\rho_S : S \rightarrow \mathbb{R}$ that ranks sellers and $\rho_M : M \rightarrow \mathbb{R}$ that ranks items. Since seller and item rankings are not readily measurable, platform providers will often estimate rankings by asking buyers to rate their interactions via some common scale (e.g., 5-star ratings). In other words, given a set of buyers $B = \{b_1, b_2, \ldots\}$, platform providers observe a set of scores $Y = \{y_{s,m}^{(b)}\}$ when a buyer b rates their interaction with seller s to purchase item m. One important, yet subtle issue in this setting is that the observed scores $y_{s,m}^{(b)}$ do not directly measure values of either the seller ratings ($\rho_S(s)$) or item quality ($\rho_M(m)$), but some function of the two.

To motivate our approach, we first consider $y_{s,m}$, the unobserved *true* score corresponding to the evaluation of seller s with regard to item m. Next, we model this true score as a function of two terms, seller performance ($\rho_S(s)$) and item quality ($\rho_M(m)$):

$$y_{s,m} = f\big(\rho_S(s), \rho_M(m)\big). \tag{1}$$

Note that any single observation $y_{s,m}^{(b)}$ is only a noisy approximation of $y_{s,m}$, since different buyers will combine the two components in a different way.[5] The first part of our proposed method involves an iterative clustering of the observed $y_{s,m}^{(b)}$ to decouple and estimate each component $\rho_S(s)$ and $\rho_M(m)$.

Initial clustering and estimation of $\boldsymbol{\rho_S}$ First, define the set $B_{s,m} \subset B$ as the set of buyers who have rated an interaction of purchasing item m from seller s (i.e., $B_{s,m} = \{b_k \in B \mid \exists\, y_{s,m}^{(b_k)} \in Y\}$). For each seller-item pair $(s, m) \in S \times M$, we initially estimate $y_{s,m}$ via the sample mean

$$\bar{y}_{s,m} = \frac{1}{|B_{s,m}|} \sum_{b \in B_{s,m}} y_{s,m}^{(b)}. \tag{2}$$

Next, we utilize the fact that multiple sellers can, and often do, offer the same items, to cluster sellers and estimate $\rho_S(s)$. For a specific seller s_i, let $M_{s_i} = \{m_j \in M \mid \exists\, y_{s_i,m_j}^{(b)} \in Y\}$ be the set of all items for which the seller s_i has been rated. Similarly, for some specific item m_j, let $S_{m_j} = \{s_i \in S \mid \exists\, y_{s_i,m_j}^{(b)} \in Y\}$ be the set of all sellers who have been rated with item m_j. We initially construct $K = |M|$ clusters of sellers, $\mathcal{C}_S$, by collecting sellers who happened to have ratings for the same item m. Formally, we write,

$$\mathcal{C}_S(k) = \{s_i \mid s_i \in S_{m_k}\}.$$

[5] We also note that each buyer ratings $y_{s,m}^{(b)}$ could contain an additional bias component that depends on the specific buyer b. For example, some buyers may intentionally give higher or lower ratings, independent of seller or item quality, to satisfy their idiosyncratic goals. We specifically address this issue in the second part of our method, INTEGRITY WEIGHTING, which is presented in Section 3.2.

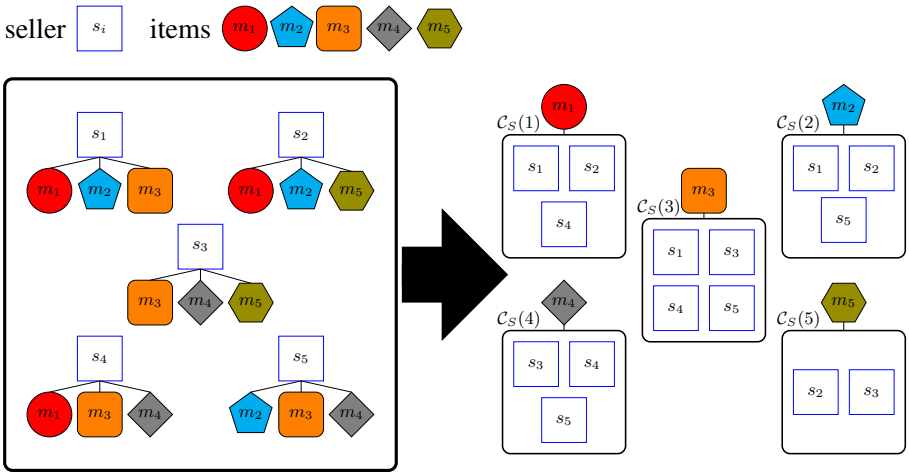

Fig. 1. An example of initial seller clustering

Further, let $\mathcal{L}_{s_i}$ be the set of parameters k such that $\mathcal{C}_S(k)$ contains seller s_i as an element. In other words,

$$\mathcal{L}_{s_i} = \{k \mid s_i \in \mathcal{C}_S(k)\}. \tag{3}$$

An example of this initial clustering is presented in Fig. 1. Fig. 1 represents a platform of five sellers, $\{s_1, s_2, \ldots, s_5\}$, and five items, $\{m_1, m_2, \ldots, m_5\}$. As a result, sellers are initially organized into five clusters based on the items they offer: $\mathcal{C}_S(1) = \{s_1, s_2, s_4\}$, $\mathcal{C}_S(2) = \{s_1, s_2, s_5\}$, $\mathcal{C}_S(3) = \{s_1, s_3, s_4, s_5\}$, $\mathcal{C}_S(4) = \{s_3, s_4, s_5\}$, and $\mathcal{C}_S(5) = \{s_2, s_3\}$. Correspondingly, while not illustrated in Fig. 1, we can write out the sets of clusters that include each seller as $\mathcal{L}_{s_1} = \{1, 2, 3\}$, $\mathcal{L}_{s_2} = \{1, 2, 5\}$, $\mathcal{L}_{s_3} = \{3, 4, 5\}$, $\mathcal{L}_{s_4} = \{1, 3, 4\}$, and $\mathcal{L}_{s_5} = \{2, 3, 4\}$.

Next, we estimate the ranking of each seller $\rho_S(s)$ by averaging relative ratings within each cluster. Define $e_k : \mathcal{C}_S(k) \to \mathbb{R}$ as a scoring function for some seller $s_i \in \mathcal{C}_S(k)$ with respect to each item m_k, *relative to all other sellers in* $\mathcal{C}_S(k)$. Specifically, we define,

$$e_k(s_i) = \bar{y}_{s_i,m_k} - \frac{1}{|\mathcal{C}_S(k)| - 1} \sum_{s_j \in \mathcal{C}_S(k) \setminus s_i} \bar{y}_{s_j,m_k}. \tag{4}$$

Then, for each seller, we subsequently estimate ρ_S by computing

$$\hat{\rho}_S(s) = \frac{1}{|\mathcal{L}_s|} \sum_{k \in \mathcal{L}_s} e_k(s) \qquad \forall s \in S. \tag{5}$$

In other words, a seller's rating, decoupled from item quality, is estimated by taking the average of all the relative scores achieved across clusters. Note that the range of $\hat{\rho}_S$ will vary depending on the range of the original scale implemented in the platform for recording buyer feedback and ratings. However, for the purpose of quantifying rankings amongst a

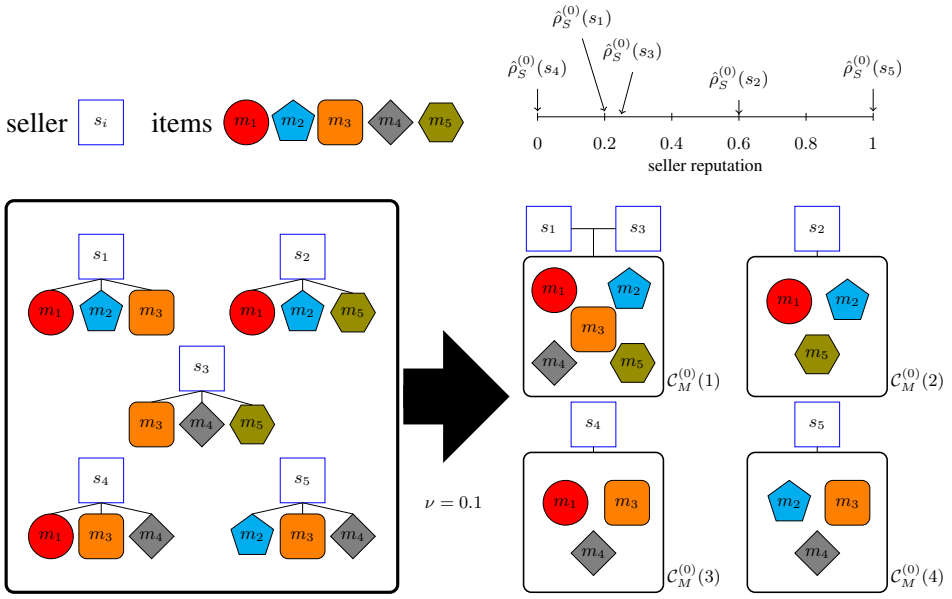

Fig. 2. An example of item clustering based on seller reputation

set of sellers, we can always normalize $\hat{\rho}_s$ to be within some desired range. For the following sections and the experiment described in Section 4, we use min-max normalization to restrict the range of $\hat{\rho}_S$ values to be in $[0, 1]$.

Iterative clustering and estimation of $\boldsymbol{\rho_M}$ Once we have the initial estimates $\hat{\rho}_S$, we can use these values to further cluster items in a similar manner, and subsequently estimate ρ_M. To formalize this iterative approach, we first denote an estimate of ρ_S and ρ_M at the t^{th} iteration as $\hat{\rho}_S^{(t)}$ and $\hat{\rho}_M^{(t)}$, respectively. Hence, our initial estimate from (5) is denoted $\hat{\rho}_S^{(0)}$, and similarly we let $e_k^{(0)}$ be our initial values of e_k computed in (4). At the t^{th} iteration, we create $K \in \mathbb{N}$ clusters of items by first grouping sellers such that seller s_i and seller s_j are in the same group if $|\hat{\rho}_S^{(t)}(s_i) - \hat{\rho}_S^{(t)}(s_j)| < \nu$, where ν is a parameter for determining the granularity and size of clusters and K is determined as a consequence of the distribution of $\hat{\rho}_S^{(t)}$. We define $\mathcal{P}_k$, the set of sellers in group k, such that $|\hat{\rho}_S^{(t)}(s_i) - \hat{\rho}_S^{(t)}(s_j)| < \nu$ for any $s_i \in \mathcal{P}_k$ and $s_j \in \mathcal{P}_k$. Then, items in the k^{th} cluster are defined as the items that have been ranked for the sellers who are in group $\mathcal{P}_k$. Formally, we write,

$$\mathcal{C}_M^{(t)}(k) = \{m_j \mid m_j \in M_{s_i} \; \forall \, s_i \in \mathcal{P}_k\}\,.$$

Similar to (3), let $\mathcal{L}_m^{(t)}$ be the set of parameters k such that $\mathcal{C}_M^{(t)}$ contains item m as a member. In other words,

$$\mathcal{L}_{m_j}^{(t)} = \{k \mid m_j \in \mathcal{C}_M^{(t)}(k)\}.$$

Continuing our illustrative example from the previous section, Fig. 2 presents a numerical example of such item clustering, where $\nu = 0.1$. Based on the numerical values of $\hat{\rho}_S^{(0)}$, presented on the upper-right scale, sellers s_1 and s_3 are grouped together, while the other sellers form singletons, resulting in $K = 4$ clusters. Without loss of generalization, we can arbitrarily assign numbers $1 \leq k \leq 4$ to each cluster, defining sets $\mathcal{P}_1 = \{s_1, s_3\}$, $\mathcal{P}_2 = \{s_2\}$, $\mathcal{P}_3 = \{s_4\}$, $\mathcal{P}_4 = \{s_5\}$; and clusters $\mathcal{C}_M^{(0)}(1) = \{m_1, m_2, m_3, m_4, m_5\}$, $\mathcal{C}_M^{(0)}(2) = \{m_1, m_2, m_5\}$, $\mathcal{C}_M^{(0)}(3) = \{m_1, m_3, m_4\}$, and $\mathcal{C}_M^{(0)}(4) = \{m_2, m_3, m_4\}$. Correspondingly, the clusters that include each item is stored as $\mathcal{L}_{m_1}^{(0)} = \{1, 2, 3\}$, $\mathcal{L}_{m_2}^{(0)} = \{1, 2, 4\}$, $\mathcal{L}_{m_3}^{(0)} = \{1, 3, 4\}$, $\mathcal{L}_{m_4}^{(0)} = \{1, 3, 4\}$, and $\mathcal{L}_{m_5}^{(0)} = \{1, 2\}$.

Similar to (2), we compute a within-cluster mean $\bar{y}_{m,k}^{(t)}$ for each item m in cluster k by taking the average rating over each buyer and seller within the cluster. In other words,

$$\bar{y}_{m,k}^{(t)} = \frac{\sum_{s \in \mathcal{P}_k} \sum_{b \in B_{s,m}} y_{s,m}^{(b)}}{\sum_{s \in \mathcal{P}_k} |B_{s,m}|}. \tag{6}$$

Let $z_k^{(t)} : \mathcal{C}_M^{(t)}(k) \to \mathbb{R}$ be a scoring function for some item $m \in \mathcal{C}_M^{(t)}(k)$, relative to all other items in $\mathcal{C}_M^{(t)}(k)$. In particular, we define

$$z_k^{(t)}(m_i) = \bar{y}_{m_i,k}^{(t)} - \frac{1}{\left|\mathcal{C}_M^{(t)}(k)\right| - 1} \sum_{m_j \in \mathcal{C}_M^{(t)}(k) \setminus m_i} \bar{y}_{m_j,k}.$$

Then, for each item, we subsequently estimate ρ_M at iteration t by computing

$$\hat{\rho}_M^{(t)}(m) = \frac{1}{\left|\mathcal{L}_m^{(t)}\right|} \sum_{k \in \mathcal{L}_m^{(t)}} z_k^{(t)}(m) \qquad \forall m \in M.$$

In other words, the quality of an item, decoupled from a seller's performance rating, is estimated by taking the average of all the relative scores achieved by that item across clusters. As in (5) for $\hat{\rho}_S$, the range of $\hat{\rho}_M^{(t)}$ will vary. For the following sections and the experiment described in Section 4, we use $\min\text{-}\max$ normalization at each iteration t to restrict the range of $\hat{\rho}_M^{(t)}$ values to be in $[0, 1]$.

Iterative clustering and estimation of $\boldsymbol{\rho_S}$ Given values of $\hat{\rho}_M^{(t)}$, we can further improve our estimate of ρ_S by taking additional iterations. An iteration of computing $\hat{\rho}_S^{(t)}$ is very similar to the initial estimation procedure we describe for (5), with the primary difference being in how clusters $\mathcal{C}_S^{(t)}(k)$ are defined for $t > 0$.

Specifically, at the t^{th} iteration for $t > 0$, we create $K \in \mathbb{N}$ clusters of sellers by first grouping items such that item m_i and item m_j are in the same group if $|\hat{\rho}_M^{(t-1)}(m_i) - \hat{\rho}_M^{(t-1)}(m_j)| < \nu$. The set of items in group k, $\mathcal{Q}_k$, is defined such that $|\hat{\rho}_M^{(t-1)}(m_i) - \hat{\rho}_M^{(t-1)}(m_j)| < \nu$ for any $m_i \in \mathcal{Q}_k$ and $m_j \in \mathcal{Q}_k$. The set of sellers in the k^{th} cluster for $t > 0$ are then defined as

$$\mathcal{C}_S^{(t)}(k) = \left\{ s_i \mid s_i \in S_{m_j} \; \forall \, m_j \in \mathcal{Q}_k \right\}, \qquad t > 0.$$

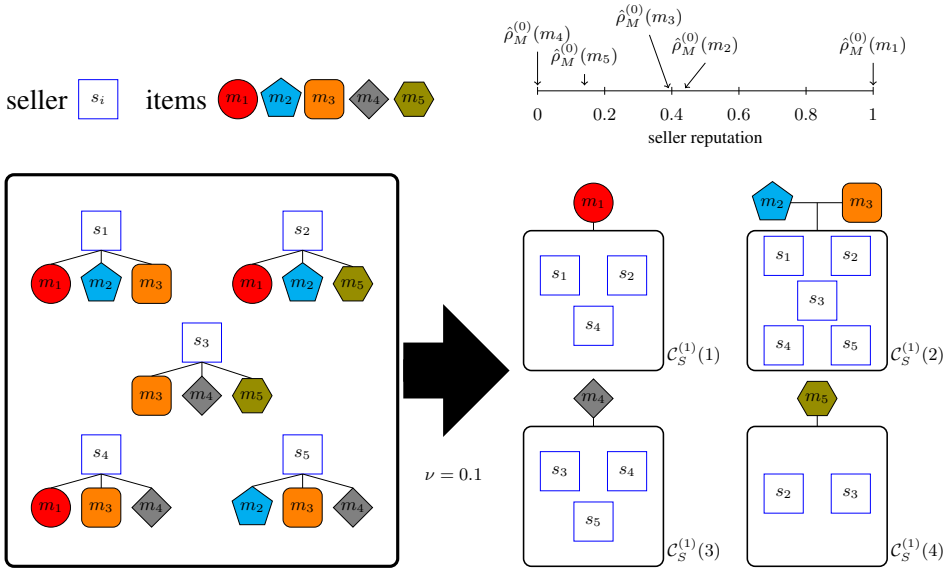

Fig. 3. An example of seller clustering based on item reputation

The set $\mathcal{L}_s^{(t)}$ is trivially defined similar to $\mathcal{L}_s$ in (3).

To continue our example, Fig. 3 illustrates such a clustering of sellers with $\nu = 0.1$ at $t = 1$. Based on the numerical values of $\hat{\rho}_M^{(0)}$, presented on the upper-right scale, items m_2 and m_3 are grouped together, while the other items form singletons, resulting in $K = 4$ clusters. Again, we can assign numbers $1 \leq k \leq 4$ to each cluster, defining sets $\mathcal{Q}_1 = \{m_1\}$, $\mathcal{Q}_2 = \{m_2, m_3\}$, $\mathcal{Q}_3 = \{m_4\}$, $\mathcal{Q}_4 = \{m_5\}$; and clusters $\mathcal{C}_S^{(1)}(1) = \{s_1, s_2, s_4\}$, $\mathcal{C}_S^{(1)}(2) = \{s_1, s_2, s_3, s_4, s_5\}$, $\mathcal{C}_S^{(1)}(3) = \{s_3, s_4, s_5\}$, and $\mathcal{C}_S^{(1)}(4) = \{s_2, s_3\}$. Correspondingly, the clusters that include each seller is stored as $\mathcal{L}_{s_1}^{(1)} = \{1, 2\}$, $\mathcal{L}_{s_2}^{(1)} = \{1, 2, 4\}$, $\mathcal{L}_{s_3}^{(1)} = \{2, 3, 4\}$, $\mathcal{L}_{s_4}^{(1)} = \{1, 2, 3\}$, and $\mathcal{L}_{s_5}^{(1)} = \{2, 3\}$.

Within-cluster mean $\bar{y}_{s,k}^{(t)}$ for each seller s in cluster k is computed by taking the average rating over each buyer and item within the cluster. In other words,

$$\bar{y}_{s,k}^{(t)} = \frac{\sum_{m \in \mathcal{Q}_k} \sum_{b \in B_{s,m}} y_{s,m}^{(b)}}{\sum_{m \in \mathcal{Q}_k} |B_{s,m}|}, \qquad t > 0. \tag{7}$$

Finally, $e_k^{(t)}$ and $\hat{\rho}_S^{(t)}$ for $t > 0$ are defined similar to the initial case of $t = 0$, following (4) and (5), but replacing the initial estimates $\bar{y}_{s,m}$ with the within-cluster average $\bar{y}_{s,k}^{(t)}$.

Complete algorithm for RATING SEPARATION As a stopping condition of the iterative algorithm, we define a tolerance parameter ε. After completing iteration $t > 0$, and given estimates $\hat{\rho}_S^{(t)}$ and $\hat{\rho}_M^{(t)}$, the algorithm is to advance to the next iteration $t + 1$ until $|\hat{\rho}_S^{(t)} - \hat{\rho}_S^{(t-1)}| < \varepsilon$ and $|\hat{\rho}_M^{(t)} - \hat{\rho}_M^{(t-1)}| < \varepsilon$. The overall procedure presented in this section is formally summarized in Algorithm 1.

Algorithm 1: Rating Separation (RS)

Input: set of buyers B, set of sellers S, set of items M, set of ratings Y, clustering range ν, convergence tolerance ε
Output: estimated quality scores for each seller and item $(\hat{\rho}_S, \hat{\rho}_M)$
$t \leftarrow 0$;
repeat
  `// Seller rating separation`
  if $t = 0$ **then**
    cluster sellers by item to build $\mathcal{C}_S^{(0)}$;
  else
    cluster sellers by item score $\hat{\rho}_M^{(t-1)}$ and ν to build $\mathcal{C}_S^{(t)}$;
  end
  foreach $\mathcal{C}_S^{(t)}(k) \in \mathcal{C}_S^{(t)}$ **do**
    foreach $s \in \mathcal{C}_S^{(t)}(k)$ **do**
      compute $e_k^{(t)}(s)$;
    end
  end
  foreach $s \in S$ **do**
    compute $\hat{\rho}_S^{(t)}(s)$;
  end
  `// Item rating separation`
  cluster items by seller score $\hat{\rho}_S^{(t)}$ and ν to build $\mathcal{C}_M^{(t)}$;
  foreach $\mathcal{C}_M^{(t)}(k) \in \mathcal{C}_M^{(t)}$ **do**
    foreach $m \in \mathcal{C}_M^{(t)}(k)$ **do**
      compute $z_k^{(t)}(m)$;
    end
  end
  foreach $m \in M$ **do**
    compute $\hat{\rho}_M^{(t)}(m)$;
  end
  $t \leftarrow t + 1$;
until $t > 0$, $|\hat{\rho}_S^{(t)} - e_{t-1}^{\star}| < \varepsilon$, $|\hat{\rho}_M^{(t)} - z_{t-1}^{\star}| < \varepsilon$;

3.2. Mitigating adversarial reviews

A known issue with many buyer rating systems is that malicious actors may negatively affect the accuracy of scores through various cheating behavior. [5, 7, 10, 19, 20, 23, 24] Here, we mitigate such risk by proposing a method to score each review in terms of an estimated measure of trustworthiness—or *integrity*, which is then used to weigh each observed rating. Our proposed measure of trustworthiness considers three components: engagement, diversity, and anomaly. We calculate each component for every buyer, based on observed rating behavior across item categories. Formal definitions of each component are presented below.

Concretely, we define $\mathcal{G}^\ell \subset M$, a subset of items in category ℓ, as a collection of items which satisfy some predetermined criteria[6]. For example, $\mathcal{G}^1$ might be the collection of all electronics, while $\mathcal{G}^2$ might be all items classified as furniture. Then, let $B^\ell \subset B$ be the set of all buyers who have rated items in $\mathcal{G}^\ell$. Similarly, we use $Y^\ell \subset Y$ to denote the subset of all ratings that were observed for items in category ℓ, while $Y^{\ell[b]} \subset Y^\ell$ further denotes the subset of ratings for items in category ℓ that were given by user b. In other words, we define

$$B^\ell = \{b \in B \mid \exists\, y^{(b)}_{s,m_j} \in Y, m_j \in \mathcal{G}^\ell\}$$
$$Y^\ell = \{y^{(b)}_{s,m_j} \in Y \mid m_j \in \mathcal{G}^\ell\}$$
$$Y^{\ell[b]} = \{y^{(b)}_{s,m_j} \in Y \mid m_j \in \mathcal{G}^\ell, b \in B^\ell\}.$$

Engagement A common measure for quantifying the trustworthiness of users on a typical platform is user engagement. For example, scores provided by a buyer who is highly engaged in the platform—purchasing items and rating interactions on a regular basis—is considered more reliable than that from a one-time visitor. Here, engagement is operationalized as the relative frequency of ratings given by each buyer b within a category ℓ:

$$\alpha_{b,\ell} = \left|Y^{\ell[b]}\right| - \frac{\left|Y^\ell\right|}{\left|B^\ell\right|},$$

where $\left|Y^\ell\right| / \left|B^\ell\right|$ is the average number of ratings provided by each buyer in category ℓ. The corresponding user engagement weights $\alpha_{b,\ell}$ are further normalized to be within the range [0, 1], using min-max normalization for each category ℓ.

Diversity Another consideration for a buyer's trustworthiness is the concentration of ratings. Conceptually, a buyer is considered more trustworthy if they interact with, and rate, a variety of different sellers, as opposed to repeatedly rating a small number of sellers. Thus, we quantify diversity as the proportion of unique sellers that the buyer has rated over all ratings the buyer has given within that category. Formally, let $S^{\ell[b]} \subset S$ be the subset sellers who have received a rating from user b, for at least one item in $\mathcal{G}^\ell$. In other words

$$S^{\ell[b]} = \{s_i \mid \exists\, y^{(b)}_{s_i,m_j} \in Y, m_j \in \mathcal{G}^\ell\}.$$

Then, the diversity weight $\beta_{b,\ell}$ for a buyer b corresponding to item category ℓ is calculated as

$$\beta_{b,\ell} = \frac{\left|S^{\ell[b]}\right|}{\left|Y^{\ell[b]}\right|}.$$

Note that this quantification of diversity is relative to the total number of ratings made by the buyer. For example, a buyer who provided only one rating ($\left|Y^{\ell[b]}\right| = 1$) is considered to have high diversity ($\beta_{b,\ell} = 1$). As the buyer rates more transactions, $\beta_{b,\ell}$ will decrease whenever the buyer rates a seller that they have already rated previously. Similar to engagement weights, we normalize the corresponding diversity weights $\beta_{b,\ell}$ via min-max normalization within each item category $\mathcal{G}^\ell$.

[6] In this study, we use item categories as defined by the lowest-level grouping of items on eBay (https://www.ebay.com/v/allcategories).

Algorithm 2: Integrity weighting (IW)

Input: set of buyers B, set of ratings Y, set of item categories $\mathcal{G}$
Output: integrity-adjusted ratings, $\hat{y}_{s,m}^{(b)}$
`// Compute integrity weights`
foreach *item group* ℓ **do**
  foreach $b \in B^{\ell}$ **do**
    $w_{b,\ell} \leftarrow \alpha_{b,\ell} \times \beta_{b,\ell} \times \gamma_{b,\ell}$;
    foreach $y_{s,m}^{(b)} \in Y^{\ell[b]}$ **do**
      $\hat{y}_{s,m}^{(b)} \leftarrow w_{b,\ell} \times y_{s,m}^{(b)}$;
    end
  end
end

Anomaly We are also concerned with how much a buyer's rating of an item deviates or conforms to that of the general consensus of other buyers, which we refer to as anomaly. To quantify anomaly, we first consider the standardized distance of a buyer's rating for each item, from the overall distribution of ratings for that item. For any given item m, let μ_m and σ_m denote the average and standard deviation of ratings that the item received across all buyers and sellers. Then, for each rating $y_{s,m}^{(b)}$ given by buyer b for item m, we compute the normalized distance from the mean as:

$$\delta_{s,m}^{(b)} = \left| \frac{y_{s,m}^{(b)} - \mu_m}{\sigma_m} \right|,$$

where smaller values of $\delta_{s,m}^{(b)}$ indicate that the ratings given by buyer b for item m is similar and consistent with ratings given by other buyers for that same item. Then, $\gamma_{b,\ell}$, the anomaly weight for buyer b in item group ℓ is computed by taking the average of $\delta_{s,m}^{(b)}$ for all items $m \in \mathcal{G}^{\ell}$:

$$\gamma_{b,\ell} = \frac{1}{\left|Y^{\ell[b]}\right|} \sum_{Y^{\ell[b]}} \delta_{s,m}^{(b)}.$$

As with engagement weights and diversity weights, anomaly weights $\gamma_{b,\ell}$ are subsequently normalized via min-max normalization within each item category $\mathcal{G}^{\ell}$.

Integrity weighted ratings Given the normalized weights $\alpha_{b,\ell}$, $\beta_{b,\ell}$, and $\gamma_{b,\ell}$ for engagement, diversity, and anomaly, respectively, we can compute a comprehensive integrity weight for each buyer b within item category $\mathcal{G}^{\ell}$ as

$$w_{b,\ell} = \alpha_{b,\ell} \times \beta_{b,\ell} \times \gamma_{b,\ell}.$$

Then, for an observed rating $y_{s,m}^{(b)}$ where $m \in \mathcal{G}^{\ell}$, we can compute an integrity-adjusted rating—where the observed rating is weighted by the estimated integrity of user b within category $\mathcal{G}^{\ell}$ as

$$\hat{y}_{s,m}^{(b)} = w_{b,\ell} \times y_{s,m}^{(b)}.$$

This procedure is formally summarized in Algorithm 2.

3.3. A comprehensive reputation score

Finally, we can compute reputation scores for sellers and items that are robust to confounded and adversarial ratings by combining RATING SEPARATION from 3.1 and INTEGRITY WEIGHTING from 3.2. This is achieved by replacing the raw ratings $y_{s,m}^{(b)}$ with their integrity weighted counter parts, $\hat{y}_{s,m}^{(b)}$, in (2), (6), and (7) of RATING SEPARATION.

4. Experiment

To evaluate the efficacy of the methods proposed, we further present a novel and comprehensive simulation framework. The contributions of our new approach are two fold. First, in contrast to existing literature we directly model item-level transactions. This enables our framework to distinguishing between a buyer's rating of sellers versus satisfaction of a specific item. Second, the proposed framework incorporates a comprehensive model of plausible adversarial behavior, allowing us to test how robust our reputation scoring systems are to numerous realistic attack scenarios.

4.1. A simulation framework for online marketplaces

The simulation framework we propose involves four components: three entities—items, sellers, buyers—and a model for how the different entities interact—transactions. A major advantage of our approach is that by explicitly modeling items, in addition to buyers and sellers, we can further capture the realistic dynamics that take place in online marketplaces.

An online marketplace is characterized by the number of items, buyers, and sellers on the platform. For our experiment, we consider two parameter regimes: small-size and large-size marketplaces. For the small-size marketplace, we set 1,000 items, 500 sellers, and 5,000 buyers For the large-size marketplace, we set 2,000 items, 1,000 sellers, and 10,000 buyers. For each setting, we simulate 300 days of marketplace activity, where each buyer is limited to one transaction per day. We describe each component of the simulation in detail below.

Items An item is parameterized by its quality and categorization. The quality of an item is represented by a continuous score in the range $[0, 1]$. We allow for multiple hierarchical item categories.

For the purpose of our simulations in this study, we limit the hierarchy of item categories to three levels—top, middle, and bottom, which we find sufficient to represent many typical online marketplace categorizations realistically. In our experiments, we set three top-level categories,each of which consists of five mid-level subcategory. Each mid-level category is further classified in to six bottom-level subcategories. In total, there are 90 different unique item categories. We further assign items uniformly across different categories, so that the number of items available in each category is similar.

Sellers Sellers are parameterized by their capability and the items that they offer. We assume that seller capabilities follow a truncated normal distribution, bounded in $[0, 1]$, with mean 0.5 and standard deviation 0.25. Higher capability scores correspond to faster

delivery and better service, while lower capability scores correspond to late delivery and poor service.

An important characteristic of sellers, which is not often captured in existing simulation frameworks, is the variety of items that they offer. For example, while some sellers may focus on selectively offering only a few items in major categories, others may choose to offer a wide-selection of items across multiple categories. By modeling items as entities, and parameterizing sellers by the items they offer, the simulation framework we propose is capable of representing this diversity.

In our experiment, we assume that sellers offer items in one major category, along with items from up to three minor categories. To operationalize this assumption, for each seller we first sample one major item category, from which they offer between three and six items. Then, we sample between zero and three minor item categories, from which one to six items are subsequently sampled.

Buyers Buyers are parameterized by item categories of interest and the level of interest for each category. Each buyer is randomly assigned to 3 to 6 item categories of interest. The level of interest for each item category is assigned a continuous value in the range $[0, 1]$. We assume that buyers are more likely to purchase items in categories for which they have a higher level of interest. After each transaction, buyers will leave a single score rating as a function of item quality and seller capability.

Transactions Transactions represent the event in which a buyer purchases an item from a seller, and provides a rating. Each buyer is assigned a random purchase cycle, between zero and three days, which represents how often the buyer will participate in a transaction on the marketplace. Buyers are more likely to purchase items from sellers who offer items for which they have higher levels of interest in. This could result in unrealistically high-frequency transactions between the same buyer-seller pairs. To mitigate this issue, we require a *repurchase waiting time* of three, five, or ten days for the same buyer-seller pairs to have a repeat transaction.

4.2. Simulating malicious ratings

To evaluate the robustness of a reputation system in the presence of adversarial buyers, we model the behavior of malicious ratings. Based on existing literature [5, 15, 20, 24, 27, 29], we categorize adversarial buyers by three behavioral patterns and six attack strategies. The three behavioral patterns and six attack strategies are presented in Tables 1 and 2, respectively, along with a short description and relevant literature reference.

Any attacker will adopt one behavioral pattern and an attack strategy, allowing for a total of 18 possible attacker types. Compared to existing work, which only consider a limited subset of these 18 possible pairs, here we investigate the performance of a rating system under all 18 types of attacks. This is achieved by modeling each type of attack behavior and strategy within the simulation framework presented in Section 4.1. In our experiment, we parameterize the intensity of attacks on a platform as the attack rate—the proportion of all ratings that are malicious. We compare results for varying attack rates, from 10% to 90%, in 10% increments.

Table 1. *Three categories of adversarial behavior patterns.*

Pattern	Description	Reference
Basic	Attackers consistently exhibit adversarial behavior— granting high ratings to conspiring sellers or low ratings to rival sellers.	[5, 7, 15, 20, 24, 26–29]
Camouflage	Attackers attempt to camouflage their adversarial intent by strategically mixing justified ratings with malicious ones. Under this behavioral scheme, attackers typically exhibit benign behavior in early interactions, and transition to adversarial activities at later stages.	[29]
Whitewashing	Attackers behave under the basic scheme, while subsequently creating multiple accounts on the platform to mitigate detection and create an illusion that their malicious ratings are socially validated.	[29]

Table 2. *Six categories of attack strategy. Each strategy is assigned a number which we use as a referense in the text.*

# Name	Description	Reference
1 Ballot stuffing (BS)	Attempting to boost the reputation of conspiring sellers by giving maximum ratings	[5, 20, 27]
2 Bad mouthing (BM)	Attempting to hurt the reputation of rival sellers by giving minimum ratings	[5, 20, 27]
3 BS & BM	Employing a mix of both ballot stuffing and bad mouthing	[5, 20, 27]
4 r-high shifting	Attempting to boost the reputation of conspiring sellers by giving ratings that are r points higher than the average ratings	[15, 20, 24]
5 r-low shifting	Attempting to hurt the reputation of rival sellers by giving ratings that are r points lower than average	[15, 20, 24]
6 r-high/low shifting	Employing both r-high shifting and r-low shifting strategies to boost reputation of conspiring sellers while simultaneously reducing the reputation of rival sellers	[15, 20, 24]

Note that the last three strategies—r-high, r-low, and r-high/low shifting—require that the attackers assign some distribution of benign buyer ratings for the target sellers.

5. Results

We use Spearman's rank correlation coefficient to measure and compare how well a reputation system can recover the true capability rankings of sellers.[7] Our results are presented in two parts. First, we investigate the efficacy of RATING SEPARATION (RS). To do so, we compare RS with a naive baseline approach of computing a seller's reputation via simple average of observed ratings. Second, to test whether INTEGRITY WEIGHTING

[7] Note that here, we focus on seller reputation, but the proposed methods and simulation framework could also be used for estimating item quality via trivial extension of this work.

(IW) and the combined approach of RATING SEPARATION AND INTEGRITY WEIGHTING (RS&IW) is truly robust to adversarial rating activity, we compare performance of each method to existing mitigation techniques.

5.1. RATING SEPARATION (RS) performance

First, we evaluate the performance of RATING SEPARATION in recovering true seller rankings. Because RATING SEPARATION in itself does not mitigate against adversarial ratings, for this section we focus on a simulated platform that assumes no malicious attacks.[8] We compare performance under two different assumptions of marketplace parameters, as described in Section 4.1. As a baseline, we compute a naive measure of seller reputation by taking the average of all ratings that a seller received.

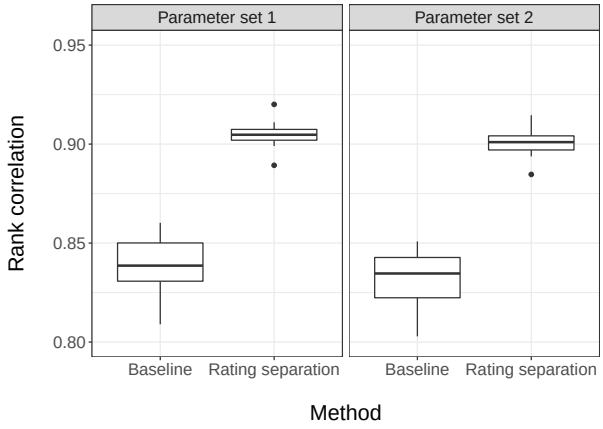

Fig. 4. *Simulation results comparing a single iteration of* RATING SEPARATION *versus the baseline. Each box plot summarizes the results of 10 simulations. The y-axis shows the rank correlation between the estimated seller reputation and true seller capability, for each method. Column panels show the two simulation parameter settings. Overall,* RATING SEPARATION *consistently recovers the true ranking of seller capability more reliably than the baseline, with less variance across each trial.*

In Fig. 4, we compare the rank correlation between estimated seller reputation and true seller capabilities for the baseline and RATING SEPARATION using just a single iteration. The box plot represents the distribution of rank correlation performance achieved for each method, across 10 simulations each.

We find that for every simulation trial, RATING SEPARATION achieved consistently higher rank correlation compared to the baseline. RATING SEPARATION also was more consistent in better recovering true seller rankings, demonstrated by the low variance in performance across simulations, compared to the baseline.

[8] We investigate robustness of our proposed methods to attacks in Section 5.2.

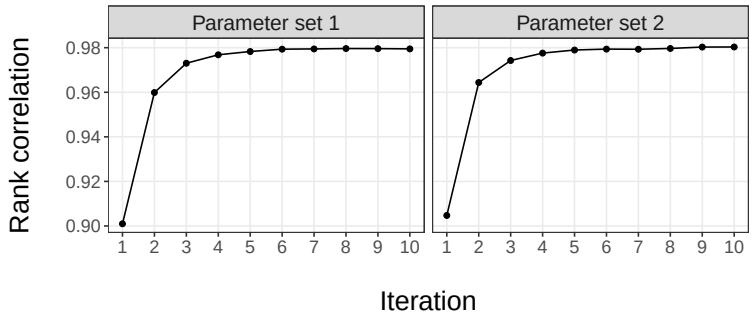

Fig. 5. *Rank correlation between the estimated seller reputation computed via* RATING SEPARATION *and true reputation as a function of the number of iterations. Column panels show the two simulation parameter sets.*

While Fig. 4 shows that just a single iteration of RATING SEPARATION can achieve superior performance compared to the baseline, the iterative nature of RATING SEPARATION allows for further improvement. In Fig. 5, we show that the performance of RATING SEPARATION can be substantially improved by just 3 additional iterations, at which point the rank correlation is close to perfect at about 0.98. This represents a tremendous improvement, considering that the baseline approach, at best, recovers seller rankings with about 0.85 correlation.

5.2. Performance with adversarial ratings

Next, we evaluate the efficacy of INTEGRITY WEIGHTING in mitigating the harms of adversarial ratings. The two approaches of using just INTEGRITY WEIGHTING and using both RATING SEPARATION AND INTEGRITY WEIGHTING are evaluated. We compare performance to three existing methods from previous literature: BRS [27], PA [28], and iCLUB [20]. A baseline approach, which does not explicitly adjust for potential adversarial behavior is included as well. As mentioned in Section 4.2, we conduct simulations for 18 possible attack types, unique pairs of three behavior patterns and six attack strategies. For each attack type, we vary the rate of attack between 10% and 90%, in 10% increments. The results are presented in Fig. 6.

From Fig. 6, we first note that the methods we propose (IW and RS&IW) consistently outperform all alternative methods in every setting that we test. Notably, we find that the baseline method, which assigns equal weight to all ratings and does not account for any malicious behavior, performs better than more sophisticated methods under some conditions.

Overall, as the rate of attacks increases, the performance of all methods in all settings decrease, albeit in varying degrees. One interesting finding is that iCLUB typically achieved either *negative* correlation, or the highest performance among the four benchmark approaches. This suggests that while iCLUB can be a high-performing method under specific assumptions of adversarial behavior, it is not generally reliable.

Under either a bad mouthing strategy (Pattern 2) or r-low shifting strategy (Pattern 5), other methods for mitigating adversarial ratings typically do no better than a naive

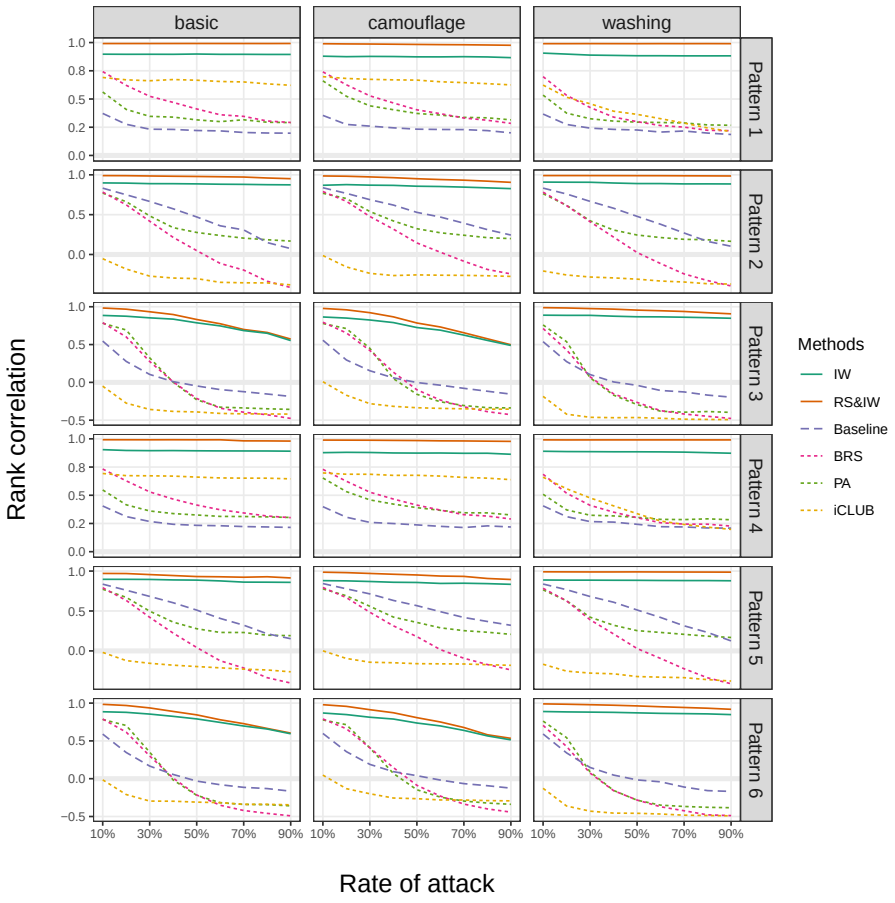

Fig. 6. *Comparison of multiple methods for mitigating malicious reviews. The x-axis shows the proportion of attacks that are assumed in each simulation, while the y-axis shows the rank correlation between the estimated seller reputation and true reputation, for each method. Column panels show different attack types and row panels show different attack patterns. Overall, the proposed methods (*RS *and* RS&IW*) are able to recover the true reputation more reliably than any existing method across all simulated circumstances.*

baseline approach. This indicates that existing methods are tailored to certain types of attack strategies, and do not perform well against a wide range of attacks, in general.

6. Conclusions

In increasingly complex online marketplaces that involve the interaction of multiple agents, evaluating the quality and characteristics of each agent is becoming more important. This paper addresses the issue of the confounded buyer ratings, as well as malicious ratings, in reputation systems and proposes RATING SEPARATION AND INTEGRITY WEIGHTING

(RS&IW), a system for providing agent reputations that are robust to confounded ratings and various types of cheating behavior. Through extensive experiments, we showed that our reputation system can both disentangle scores for sellers from the confounded ratings and are robust to numerous realistic attack scenarios generated by incorporating a comprehensive model of plausible adversarial behaviors.

While, in the interest of clarity and consistency, we have focused our work in this paper on the concrete problem of identifying seller rankings and mitigating malignant buyer behavior, the methods we propose could be extended to a broader family of problems in a more general context of multi-agent platforms. One possible extension would be to apply RATING SEPARATION in matching markets, where participating agents report numerous confounded signals with regard to the quality of other entities. For example, in ride sharing applications, RATING SEPARATION could be applied on ratings to decouple rider satisfaction of driver (e.g., personal, vehicle) and route (e.g., traffic conditions, travel time) characteristics. Or in a three-sided market, such as food delivery services, RATING SEPARATION could be extended to disentangle courier and restaurant ratings from an eater's single score.

Besides seller performance and item quality, item price is the one of the main factors having an influence on conforming a single score. Sometimes, buyers could give a generous score for a seller, even though both this seller's performance and his item quality are not satisfactory, because the price is discovered as the lowest one in an online platform. In a further study, we plan to develop a framework to accurately disentangle this price effect from a user's single score for measuring better purified seller reputation.

Acknowledgments. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. NRF-2020R1A2B5B03001960), the Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7069440), and by Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2020-0-01373, Artificial Intelligence Graduate School Program (Hanyang University))

Bibliography

[1] Bidgoly, A., Ladani, B.: Modeling and Quantitative Verification of Trust Systems Against Malicious Attackers. The Computer Journal 59(7), 1005–1027 (2016)

[2] Biega, A.J., Gummadi, K.P., Weikum, G.: Equity of attention: Amortizing individual fairness in rankings. In: The 41st international acm sigir conference on research & development in information retrieval. pp. 405–414 (2018)

[3] Cabral, L., Hortacsu, A.: The dynamics of seller reputation - theory and evidence from eBay. The Journal of Industrial Economics LVIII(1), 54–78 (2010)

[4] Chandrasekaran, P., Esfandiari, B.: 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. pp. 296–303. IEEE (2011)

[5] Dellarocas, C.: Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior. In: Proceedings of the 2nd ACM conference on Electronic Commerce. pp. 150–157 (2000)

[6] Fan, Z.P.and Xi, Y., Liu, Y.: Supporting consumer's purchase decision: a method for ranking products based on online multi-attribute product ratings. Soft Computing 22, 5247–5261 (2018)

[7] Fang, H., Zhang, J., Sensoy, M., Thalmann, N.M.: SARC : Subjectivity Alignment for Reputation Computation (Extended Abstract) Categories and Subject Descriptors. In: Proceedings of the 11th International Con- ference on Autonomous Agents and Multiagent Systems (AAMAS). pp. 1365–1366 (2012)

[8] Gao, J., Zhou, T.: Evaluating user reputation in online rating systems via an iterative group-based ranking method. Physica A: Statistical Mechanics and its Applications 473, 546–560 (2017)

[9] Ghiasi, H., Brojeny, M., Gholamian, M.: A reputation system for e-marketplaces based on pairwise comparison. Knowledge and Information Systems 56, 613–636 (2018)

[10] Houser, D., Wooders, J.: Reputation in auctions: Theory, and evidence from eBay. Journal of Economics and Management Strategy 15(2), 353–369 (2006)

[11] Howe, J.: The Rise of Crowdsourcing. Wired Magazine (14) (2006), http://archive.wired.com/wired/archive/14.06/crowds.html

[12] Irissappane, A.A., Jiang, S., Zhang, J.: Towards a comprehensive testbed to evaluate the robustness of reputation systems against unfair rating attacks. In: User Modeling Adaptation and Personalizatoin Workshops (2012)

[13] Jiang, W, X.Y.G.H.W.C.Z.L.: Multi agent system-based dynamic trust calculation model and credit management mechanism of online trading. Intelligent Automation & Soft Computing 22(4), 639–649 (2016)

[14] Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems 43(2), 618–644 (2007)

[15] Jøsang, A., Ismail, R., Jsang, A., Ismail, R.: The Beta Reputation System. Proceedings of the 15th Bled Electronic Commerce Conference 160, 324–337 (2002)

[16] Kerr, R., Cohen, R.: TREET: The Trust and Reputation Experimentation and Evaluation Testbed. Electronic Commerce Research 10(3), 271–290 (2010)

[17] Kramer, M.A.: Self-selection bias in reputation systems. In: Etalle, S., Marsh, S. (eds.) IFIP International Federation for Information Processing, vol. 238, chap. Trust Mang, pp. 255–268. Boston: Springer (2007)
[18] Leadbeater, C.: We-think. Profile books (2009)
[19] Lim, E.P., Nguyen, V.A., Jindal, N., Liu, B., Lauw, H.W.: Detecting product review spammers using rating behaviors. In: Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10. pp. 939–948 (2010)
[20] Liu, S., Zhang, J., Mao, C., Theng, Y., Kot, A.: iCLUB: an integrated clustering based approach to improve the robustness of reputation systems. In: Proceedings of 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS). pp. 1151–1152 (2011)
[21] Ma, L., Pei, Q., Xiang, Y., Yao, L., Yu, S.: A reliable reputation computation framework for online items in E-commerce. Journal of Network and Computer Applications 134, 13–25 (2019)
[22] Oh, H., Kim, S., Park, S., Zhou, M.: Can You Trust Online Ratings? A Mutual Reinforcement Model for Trustworthy Online Rating Systems. IEEE Transactions on Systems, Man, and Cybernetics 45(12), 1564–1576 (2015)
[23] Oh, H.K., Kim, S.W., Park, S., Zhou, M.: Trustable aggregation of online ratings. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13. pp. 1233–1236 (2013)
[24] Regan, K., Poupart, P., Cohen, R.: Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change. In: Proceedings of the National Conference on Artificial Intelligence. vol. 21, p. 1206 (2006)
[25] Resnick, P., Zeckhauser, R., Swanson, J., Lockwood, K.: The value of reputation on eBay: A controlled experiment. Experimental Economics 9(2), 79–101 (2006)
[26] Teacy, W.T., Patel, J., Jennings, N.R., Luck, M.: TRAVOS: Trust and reputation in the context of inaccurate information sources. Autonomous Agents and Multi-Agent Systems 12(2), 183–198 (2006)
[27] Whitby, A., Jøsang, A., Indulska, J.: Filtering Out Unfair Ratings in Bayesian Reputation Systems. Icfain Journal of Management Research 6(2), 106–117 (2005)
[28] Zhang, J., Cohen, R.: Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach. Electronic Commerce Research and Applications 7(3), 330–340 (2008)
[29] Zhang, L., Jiang, S., Zhang, J., Ng, W.K.: Robustness of Trust Models and Combinations. Trust Management VI 374, 36–51 (2012)
[30] Zhou, X., Murakami, Y., Ishida, T., Liu, X., Huang, G.: ARM: Toward Adaptive and Robust Model for Reputation Aggregation. IEEE Transactions on Automation Science and Engineering 17(1), 88–99 (2020)

Hyun-Kyo Oh received his B.S., M.S. and Ph.D. degree in Electronics and Computer Engineering from Hanyang University, Seoul, Korea at 2008, 2010 and 2016. He visited the Department of Computer Science of Carnegie Mellon University as a visiting scholar in 2013. He worked with the Knowledge Computing Group at Microsoft Research Asia as a research intern from 2014 to 2015. In 2016, he joined Samsung Electronics, where he currently is a lead data scientist working on creating a new machine learning and deep

learning platform that deals with the health of V-NAND flash memory and the performance of Solid State Disk (SSD). Now, he is also a visiting researcher at Institute for Software Research at Carnegie Mellon University.

Jongbin Jung received a Ph.D. in Computational Social Science and Decision Analysis from Stanford University. He is primarily interested in using quantitative methods and data analytics to help improve and evaluate human decisions. Jongbin studied operations research (M.S.) and business administration (B.B.A.) at Yonsei University (Seoul, South Korea).

Sunju Park received her B.S. and M.S. in computer engineering from Seoul National University, Seoul, Korea, and the Ph.D. degree in computer science and engineering from the University of Michigan, Ann Arbor, MI, USA. She has served on the faculties of Management Science and Information Systems, Rutgers University, NJ, USA. She is a professor of Operations, Decisions and Information at School of Business, Yonsei University, Seoul, Korea. Her current research interests include analysis of online social networks, multiagent systems for online businesses, and pricing of network resources. Her publications include Computers and Industrial Engineering, Electronic Commerce Research, Transportation Research, IIE Transactions, the European Journal of Operational Research, the Journal of Artificial Intelligence Research, Interfaces, Autonomous Agents and Multiagent Systems, and other leading journals.

Sang-Wook Kim received the B.S. degree in computer engineering from Seoul National University, in 1989, and the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology (KAIST), in 1991 and 1994, respectively. In 2003, he joined Hanyang University, Seoul, Korea, where he currently is a professor at the Department of Computer Science and Engineering and the director of the Brain-Korea21-Plus research program. He is also leading a National Research Lab (NRL) Project funded by the National Research Foundation since 2015. From 2009 to 2010, he visited the Computer Science Department, Carnegie Mellon University, as a visiting professor. From 1999 to 2000, he worked with the IBM T. J. Watson Research Center, USA, as a postdoc. He also visited the Computer Science Department at Stanford University as a visiting researcher in 1991. He is an author of more than 200 papers in refereed international journals and international conference proceedings. His research interests include databases, data mining, multimedia information retrieval, social network analysis, recommendation, and web data analysis. He is a member of the ACM and the IEEE.

Received: November 22, 2019; Accepted: May 27, 2020.