Mining Deficiencies of Online Reputation Systems: Methodologies, Experiments and Implications

Hong Xie, *Member, IEEE* and John C.S. Lui, *Fellow, IEEE*

Abstract—Online reputation systems serve as core building blocks in various Internet services such as E-commerce (e.g., eBay) and crowdsourcing (e.g., oDesk). The flaws and deficiencies of real-world online reputation systems have been reported extensively. Users who are frustrated about the system will eventually abandon such service. However, there is no systematic and formal studies which examine such deficiencies. This paper presents the first attempt, which develops a novel data analytical framework to uncover online reputation system deficiencies from data. We develop two novel measures to quantify the efficiency of online reputation systems: (1) ramp up time of a new service provider, (2) long term profit gains for a service provider. We present a new data analytical framework to evaluate these two measures from data. We show that inherent preferences or personal biases in expressing feedbacks (or ratings) cause the computational infeasibility in evaluating the ramp up time and the long term profit gains from data. We develop two computationally efficient randomized algorithms with theoretical performance guarantees to address this computational challenge. We apply our methodology to analyze real-life datasets (from eBay, Google Helpouts, Amazon and TripAdvisor). We extensively validate our model and we uncover the deficiencies of online reputation systems. Our experimental results uncovers insights on why Google Helpouts was eventually shut down in April 2015 and why eBay is losing some sellers heavily.

Index Terms—Online reputation systems, ramp up time, long term profit gains, approximation algorithms

1 Introduction

With the advancement of Internet technologies, a variety of online services are booming. E-commerce systems such as eBay [7] and Taobao [25] of Alibaba are representative examples. In an E-commerce system, buyers can purchase products from strangers and transactions are conducted online. Another typical Internet service is online product review website such as TripAdvisor [27], etc. In such websites, customers share their experiences on products, so that other customers can make purchasing decisions based on these experiences. Crowdsourcing services such as Google Helpouts [9] and oDesk [20] are another form of Internet services, where requester can outsource a task to different workers. One common characteristic of the above services is that transactions are usually carried out between two "strangers", and there is a risk because sellers may sell low quality goods while workers may provide low quality solutions. To overcome such risk, Internet service companies deploy reputation systems [21].

In general, an online reputation system involves two parties: "*service providers*" and "*customers*". A service provider can be a seller in eBay, a worker in Google Helpouts, or a hotel chain in TripAdvisor. A customer can be a buyer in eBay, a task requester in Google Helpouts, or a traveller in TripAdvisor. A transaction can be a buyer purchasing a product from a seller, a requester paying a worker to solve a task, or a customer spending an evening in a hotel. When a transaction is completed, a customer gives a feedback rating to indicate the quality of a service. For example, eBay adopts a three-level cardinal rating metric: {"negative", "neutral", "positive"}. Each service provider is associated with a reputation score, which is the aggregation of all its feedback ratings. The reputation score reflects the "*overall quality*" of service providers, and each service provider's reputation is accessible by all customers.

Many reports have indicated that existing online reputation systems have critical flaws, which result in losing users and putting Internet service companies at the risk of significant revenue loss. For example, it was reported in [29] that the eBay reputation system frustrates sellers. More concretely, the eBay reputation system forces some sellers out of the business because it makes them difficult to attract customers. In fact, eBay was reported to have a significant user loss [12], [23], [26]. Similarly, Google Helpouts was eventually shut down in April 2015 due to poor business [8]. It is important to formally explore these phenomena: *What are the key factors which influence the efficiency of online reputation systems? How to uncover the deficiencies of online reputation systems from data?* Exploring these questions not only can help us to uncover potential risks of online reputation systems, but we also can gain important insights to improve them.

• *The authors are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong. E-mail: hongx87@gmail.com, cslui@cse.cuhk.edu.hk.*

Manuscript received 2 Jan. 2017; revised 10 June 2017; accepted 10 July 2017. Date of publication 21 July 2017; date of current version 13 Oct. 2020. (Corresponding author: Hong Xie.)
Digital Object Identifier no. 10.1109/TSC.2017.2730206

Despite its importance, there is no formal study to explore deficiencies of online reputation systems. This paper aims to fill this void. However, there exists at least three challenges: (1) What is the right performance measure to quantify the *efficiency* of online reputation systems? (2) How to efficiently process online reputation datasets and apply these measures to analyze them? (3) How to address computational challenges arose in large scale reputation data analysis? This paper addresses these challenges. Our contributions are:

- We propose two measures: *ramp up time* and *long term profit gains*, to quantify the efficiency of feedback-based online reputation systems.
- We show that preferences or personal biases in assigning feedbacks (or ratings) cause the computational infeasibility in evaluating our proposed measures from data. We propose computationally efficient randomized algorithms (with theoretical performance guarantees) to address the above computational challenge.
- We apply our methodology to real-life datasets from eBay, Google Helpouts, Amazon and TripAdvisor. We extensively validate our model. We discover the deficiencies of online reputation systems: (1) the ramp up time is more than 500 days; (2) reducing ramp up time can improve the long term profit gains significantly, e.g., an 80 percent reduction on ramp up time leads to at least 50 percent (as high as 100 percent) improvement in long term profit gains. Our experimental results also uncovers insights on why Google Helpouts was eventually shut down in April 2015 and why eBay is losing sellers heavily.

This paper organizes as follows. In Section 2, we present the system model for online reputation systems. In Section 3 we formulate our problem. In Section 4, we present a novel data analytical framework to evaluate reputation systems. In Section 5 We then extend our framework to incorporate preferences or personal biases in assigning feedbacks (or ratings). In Section 6 we present the design of two randomized algorithms to approximate ramp up time and long term profit gains respectively. In Section 7 we present experimental results using datasets from eBay and Google Helpouts. Related work is given in Section 8 and we conclude in Section 9.

2 System Model

We present a general model to characterize online reputation systems, which are deployed in different types of Internet services, e.g., electronic commerce like eBay [7], crowdsourcing services like Google Helpouts [9], and hotel review websites like TripAdvisor [27]. In general, such Internet services consist of "*service providers*", "*customers*" and a "*reputation system*".

- *Service Providers:* we define "service providers" as users that supply items. A service provider can be a seller in eBay, a worker in Google Helpouts, or a hotel chain in TripAdvisor.
- *Customers:* we define "customers" as users that purchase items. A customer can be a buyer in eBay, a task requester in Google Helpouts, or a traveller in TripAdvisor.

TABLE 1
Main Notations

g	unit per product profit gain
$r(t)$	the average rating up to time slot t
$\mathcal{R}(\cdot)$	the rating mapping function
m	the total number of rating levels
$n_i(t)$	the number of level i rating up to time slot t
$Q, \widehat{Q}$	the intrinsic, perceived quality of a service provider
γ, N_h	threshold on average rating, total number of ratings
λ_1, λ_2	transaction rate before, after ramping up
$\lambda(t)$	transaction rate at time slot t
$T_t, E[T_r]$	ramp up time, expected ramp up time
G	expected long term profit gains
$\widehat{E}[T_r], \widehat{G}$	estimated ramp up time, long term profit gains
η_i	the probability of receiving a level i rating
δ	the discounting factor
ϵ	the relative estimation error
ξ	the fail probaibility

Customers conduct transactions with service providers. To encourage transactions among customers and service providers, Internet service companies use reputation systems to reflect the "*overall quality*" of service providers, and each service provider's reputation is accessible by all customers. Clearly, a service provider having a high reputation can attract more customers, which leads to larger revenue. Table 1 lists all the key notations.

2.1 Transaction Model

A customer pays a fee, which we call the $price$, to a service provider in order to complete a transaction. For example, a buyer pays a seller some money to buy a product, or a requester pays a worker some money to have a task solved. Without loss of generality, we focus on a normalized price, i.e., $price \in [0,1]$. A service provider incurs a $cost \in [0,1]$, in providing a service to a customer, i.e., a product has a manufacturing cost, or solving a task has a cost. The internet service company (e.g., eBay, Taobao, Google Helpouts), charges a $transaction\ fee \in [0,1]$ for each transaction. Our analysis also applies for a transaction fee, which is proportional to the price, because the focus of this paper is not on the pricing strategies. Thus, for brevity, we consider a fixed transaction fee. A service provider receives a unit profit gain of g for a completed transaction

$$g \triangleq price - cost - transaction\ fee. \tag{1}$$

To incentivize service providers to participate, we must have $g > 0$, which yields $transaction\ fee < price - cost$.

2.2 Model for Online Reputation Systems

Many Internet service companies deploy reputation systems to reflect the overall quality of service providers. For example, eBay maintains a reputation system to reflect the trustworthiness of sellers. In general, such reputation systems are composed of a "*feedback rating mechanism*" and "*a rating aggregating policy*".

When a transaction completes, a customer expresses a feedback rating to indicate the quality of a service. One most commonly adopted rating metric is the m-level cardinal rating metric $\{1,\ldots,m\}$, where $m \geq 2$. For example, eBay adopts a three-level cardinal rating metric: $\{1 =$

"negative", 2 = "neutral", 3 = "positive"}, while TripAdvisor adopts a five-level cardinal rating metric: {1="Terrible", 2 = "Poor", 3 ="Average", 4="Very good", 5="Excellent"}. Each rating level is associated with a numerical score, which is used to compute reputation scores for a service provider. Let $\mathcal{R}:\{1,\ldots,m\}\rightarrow\mathbb{R}$ denote a map which prescribes a score for each rating level, i.e., eBay adopts $\mathcal{R}(1)=-1, \mathcal{R}(2)=\ 0, \mathcal{R}(3)=1$, and TripAdvisor adopts $\mathcal{R}(i)=i, \forall i=1,\ldots,5$.

Each service provider is associated with a reputation score, which is an "*aggregation*" of all its feedback ratings. The reputation score quantifies the overall reputation of a service provider. One most widely adopted rating aggregating rule is the average score rule. Let r denote the reputation score of a service provider. Let n_i denote the number of ratings that are of rating level i, where $i=1,\ldots,m$, we have

$$r=\frac{\sum_{i=1}^{m} n_i \mathcal{R}(i)}{\sum_{i=1}^{m} n_i}. \tag{2}$$

The reputation score r is a public information accessible by all customers. For the ease of presentation, this paper focuses on the average score rule. We will see later, the results can be extended to the weighted average score rule.

We now describe the reputation updating process. Let $(r, n_1, \ldots, n_m)$ be the reputation profile for a service provider. In order to assist customers to assess the overall reputation of service providers, Internet service companies publish reputation profiles to the public. We use a discrete time system to characterize the reputation updating process. Let $(r(t), n_1(t), \ldots, n_m(t))$ be the reputation profile of a service provider at time slot $t\in\{0,1,\ldots,\infty\}$, where $r(t)$ is its reputation score up to time slot t, and $n_i(t)$ is the cumulative number of level i ratings up to time slot t. Each service provider is initialized with $(0,0,\ldots,0)$. Let $N_i(t)$ denotes the number of transactions completed in time slot t that lead to the level i rating. In real-world reputation systems, reputation updating has delays. We assume that the delay is one time slot. The reputation profile is updated as follows:

$$\begin{cases} r(t+1)=\frac{r(t)\sum_{i=1}^{m} n_i(t)+\sum_{i=1}^{m}\mathcal{R}(i)N_i(t)}{\sum_{i=1}^{m} n_i(t)+\sum_{i=1}^{m} N_i(t)}, \\ n_i(t+1)=n_i(t)+N_i(t), \text{for all } i=1,\ldots,m. \end{cases} \tag{3}$$

For brevity, we drop the time stamp t in our analysis when there is no confusion.

2.3 Model for Rating Behavior

Each service provider has an intrinsic quality, which indicates his true overall service quality. For example, a high quality seller in eBay sells high quality products and provides fast shipment. Let $Q\in[\mathcal{R}(1),\mathcal{R}(m)]$ denote the intrinsic quality of a service provider. After completing a transaction, a customer perceives the quality of a service provider, which is denoted by $\hat{Q}\in[\mathcal{R}(1),\mathcal{R}(m)]$. Customers pick the rating level, which is the most accurate one in reflecting the perceived quality $\hat{Q}$. Formally, we have

$$\text{feedback rating} = \arg\min_{i\in\{1,\ldots,m\}} \left|\mathcal{R}(i)-\hat{Q}\right|, \tag{4}$$

where the feedback rating denotes the individual rating assigned by a customer. For example, consider $m=5$ and $\mathcal{R}(i)=i$ for all $i\in\{1,\ldots,5\}$. When $\hat{Q}=4.7$, the feedback rating will be 5. When $\hat{Q}=4.4$, the feedback rating will be 4. Consider $\hat{Q}=4.5$, then according to Equation (4), both rating 4 and 5 are valid. In such ties, we pick the larger one, e.g., here we pick 5 to model that customers are lenient in assigning ratings. This choice will not influence the results for this paper.

For the purpose of illustrating intuitions and key ideas, we first assume that there are no errors in perceiving quality, i.e., $\hat{Q}=Q$. We will extend our model to accommodate quality perceiving errors (i.e., $\hat{Q}\neq Q$) in Section 5.

2.4 Model for Transaction's Arrival Rate

We now quantify the impact of a reputation system on service providers' revenue. The reputation system builds trust among customers and service providers. This trust is critical in attracting transactions. More precisely, customers aim to minimize the risk in service purchase and they prefer to interact with reputable service providers.

Based on the reputation profile, we categorize service providers into two types: "*reputable*", and "*average*". Note that each service provider is initialized with $(0,0,\ldots,0)$. To earn a reputable label, a service provider must improve his reputation to meet two requirements. The first one is that the reputation score r must be larger than or equal to a threshold $\gamma\in[\mathcal{R}(1),\mathcal{R}(m)]$. This requirement shows that a service provider can provide services with a high overall quality. The second requirement is that the number of feedback ratings must be larger than or equal to a threshold N_h. This requirement guarantees that the reputation score is statistically significant. Otherwise, a service provider is labeled as average.

Definition 1. *A reputable service provider must satisfy the following two conditions: $r\geq\gamma$ and $\sum_{i=1}^{m} n_i\geq N_h$. A service provider is labeled as an average service provider if and only if $r<\gamma$ or $\sum_{i=1}^{m} n_i<N_h$.*

Note that a new service provider is initialized with $(0,0,\ldots,0)$. Hence, a new service provider is always labeled as "*an average service provider*". We need both the rating scale and score scale in order to make our model practical, because in real-world applications such as eBay and Google helpouts both of these two scales are be displayed to users. These two scales are two important indicators of a service provider's reputation.

A service provider's label (i.e., reputable, or average label) is critical to its revenue. Customers are more willing (unwilling) to conduct transactions with reputable (average) service providers. Let λ_1 and λ_2 be the transaction's arrival rate when a service provider is labeled as average and reputable respectively. The transaction's arrival rate satisfies $\lambda_1<\lambda_2$, which signifies that a reputable service provider can attract more transactions.

Definition 2. *Denote $\lambda(t)$ the transaction's arrival rate at time slot t. Formally we can express it as*

$$\lambda(t)=\begin{cases}\lambda_1, & \text{if } r(t)<\gamma \text{ or } \sum_{i=1}^{m} n_i(t)<N_h, \\ \lambda_2, & \text{if } r(t)\geq\gamma \text{ and } \sum_{i=1}^{m} n_i(t)\geq N_h.\end{cases} \tag{5}$$

Let $N(t)$ denote the number of transactions that arrive to a service provider in time slot t. Then we have $E[N(t)] = \lambda(t)$. This paper focuses on that $N(t)$ follows a Poisson distribution with parameter $\lambda(t)$. This point is verified on real-world dataset in Section 7.4.

Remark. Our model focuses on two types of rates, i.e., λ_1, λ_2, in order to strike a good balance between simplicity and practicability. It is important to keep the simplicity of our model, since it enables us to present the key ideas and insights in a clear fashion. Note that our model is practical enough as well. To uncover the deficiencies a reputation system, it is reasonable to examine the average rate to service providers (either average or reputable provider). The transactions' rate λ_1 and λ_2 can be interpreted as the average rate to average service providers and reputable service providers respectively. These two types of rates are sufficient to capture the key impact of a reputation system on the profit of service providers.

3 Problem Statement

We formulate two novel measures to quantify the efficiency of online reputation systems: (1) *ramp up time T_r*, (2) *expected long term profit gains G* for a service provider. We pose a question of how to infer these two measures from real-world online reputation systems' datasets and how to uncover inefficiencies and limitations of real-world reputation systems.

3.1 Ramp Up Time

The number of time slots that a new service provider needs so to ramp up his reputation (or attain a "reputable" label) is critical to his revenue and it also affects the transaction gains of the Internet service company. Recall that each new service provider is initialized with an average label, while customers are more willing to conduct transactions with reputable service providers than average labeled service providers. Hence, it is critical for a service provider to earn a reputable label quickly so as to increase the transaction volume. Furthermore, the Internet service company will have a higher transaction gain when transaction volume increases. We next state the ramp up condition and the ramp up process.

Definition 3. *A new service provider's reputation profile is* $(0, 0, \ldots, 0)$. *To earn a reputable label, he must collect enough high feedback ratings. We define the process of earning a reputable label, i.e., increasing his reputation to* $r \geq \gamma$ *and* $\sum_{i=1}^{m} n_i \geq N_h$, *as the ramp up process. Furthermore, we say that a service provider satisfies the ramp up condition iff* $r \geq \gamma$ *and* $\sum_{i=1}^{m} n_i \geq N_h$.

Recall that a service provider's reputation profile at time slot t is $(r(t), n_1(t), \ldots, n_m(t))$. In the following, we formally define the ramp up time.

Definition 4. *The ramp up time is the minimum number of time slots that a service provider must spend to earn a reputable label. Let T_r denote the ramp up time*

$$T_r \triangleq \arg\min_{t} \left\{ r(t) \geq \gamma \text{ and } \sum_{i=1}^{m} n_i(t) \geq N_h \right\}. \tag{6}$$

The ramp up time quantifies the minimum time that a service provider must spend to earn a reputable label. It reflects how difficult it is for a service provider to start a business. If the ramp up time is large, a service provider may drop out or change to some other Internet service companies. We therefore consider the following problem.

Problem 1. *How to infer the ramp up time from real-world reputation system datasets and how it influences the efficiency of real-world reputation systems?*

3.2 Long Term Profit Gains

Profit gains are critical to service providers. They serve as one important incentive for service providers to maintain their business and they are one of the key motivations for service providers to join an Internet service company. If service providers have large profit gains, this also implies that the Internet service company (e.g., eBay or Alibaba) will have higher profit. On the other hand, service providers may quit if there is only a small profit gain, and this may lead to losses to an Internet service company.

We now formally quantify profit gains. Recall that a service provider earns a unit profit gain of g for completing one transaction (refer to Equation (1)). Note that $N(t)$ is a random variable which has a Poisson distribution with parameter $\lambda(t)$, where $\lambda(t)$ is expressed in Equation (5). Hence, on average, a service provider earns a profit gain of $gN(t)$ in the time slot t. Using micro-econometric analysis, we use a discounted long term profit gain to quantify service providers' total profit gains in time slot $0, 1, \ldots, \infty$. Let $\delta \in (0, 1]$ be the discounting factor.

Definition 5. *Denote G the expected long term profit gains for a service provider. We can express it as have*

$$G \triangleq E\left[\sum_{t=0}^{\infty} \delta^t g N(t)\right]. \tag{7}$$

We consider the following problem.

Problem 2. *How to infer the long term profit gains G from real-world reputation system datasets and reveal its impact on the real-world reputation systems?*

4 Baseline Data Analytical Framework

We first develop a data analytical framework to uncover deficiencies of online reputation systems. We then develop theoretical foundations for such framework, i.e., derive analytical expressions for the ramp up time T_r and the expected long term profit gains G. This gives us important insights to develop efficient algorithms to evaluate T_r and G from data.

4.1 Data Analytical Framework

Our baseline data analytical framework consists of three steps. In the first step, we infer model parameters, i.e., $m, \gamma, \mathcal{R}, N_h, \lambda_1$ and λ_2, from data. In the second step, we input them into our model, and apply our model to evaluate system efficiency measures, ramp up time T_r and long term profit gains G. In the third step we empirically analyze the ramp up time T_r and expected long term profit gains G so as to uncover deficiencies of online reputation systems. We outline this baseline data analytical framework in Algorithm 1.

Algorithm 1. Baseline Data Analytical Framework

1: **Parameter inference.** Infer model parameters $m, \gamma, \mathcal{R}, N_h, \lambda_1$ and λ_2 from data.
2: **Quantifying system efficiency.** Evaluating the ramp up time T_r and expected long term profit gains G based on these inferred parameters.
3: **Uncover system deficiencies.** Uncover deficiencies of online reputation systems via empirical studies on T_r and G.

In the remaining of this section, we focus on step two of the above framework, i.e., quantifying the system efficiency (we will present the details of step 1 and step 3 of the above framework in Section 7). More specifically, we derive analytical expressions for the ramp up time T_r and expected long term profit gains G, assuming that model parameters $m, \gamma, \mathcal{R}, N_h, \lambda_1$ and λ_2 are given. Then we apply them to develop efficient algorithms to evaluate T_r and G from data.

4.2 Algorithm to Evaluate Ramp Up Time

Recall that customers can perceive the intrinsic quality of service providers, i.e., $\hat{Q} = Q$. Applying Equation (4), we obtain that a service provide will receive $\arg\min_{i\in\{1,\ldots,m\}} |\mathcal{R}(i) - Q|$ level ratings. This means that the reputation score of a service provider will be of $r = \mathcal{R}(\arg\min_{i\in\{1,\ldots,m\}} |\mathcal{R}(i) - Q|)$. By Definition 4, a service provider can get ramped up if and only if $r \geq \gamma$, which yields $\mathcal{R}(\arg\min_{i\in\{1,\ldots,m\}} |\mathcal{R}(i) - Q|) \geq \gamma$. We can then introduce the notation of intrinsically reputable and average service providers respectively.

Definition 6. *We say a service provider is intrinsically reputable if and only if his intrinsic quality satisfies* $\mathcal{R}(\arg\min_{i\in\{1,\ldots,m\}} |\mathcal{R}(i) - Q|) \geq \gamma$, *otherwise we say a service provider is intrinsically average.*

We express the analytical expression for ramp up time T_r in the following theorem. This will give us important insights to develop algorithms to evaluate T_r from data.

Theorem 1. *Consider an intrinsically average service provider, the ramp up time can be expressed as*

$$T_r = \infty. \tag{8}$$

Consider an intrinsically reputable service provider, the expected ramp up time can be expressed as

$$E[T_r] = \sum_{t=1}^{\infty} \sum_{k=0}^{N_h - 1} e^{-(t-1)\lambda_1} \frac{((t-1)\lambda_1)^k}{k!}. \tag{9}$$

Remark. All proofs to lemmas and theorems are in the supplementary file, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TSC.2017.2730206. An intrinsically average service provider never gets ramped up. As long as we can infer λ_1 and N_h from data, we can evaluate the ramp up time $E[T_r]$ by applying Equations (8) and (9). However, one technical issue is that we have to perform a summation of infinity number of terms in order to evaluate Equation (9). In the following theorem we address this issue via truncation. Let $\hat{E}[T_r]$ denote an estimation of $E[T_r]$.

Theorem 2. *Let* $\hat{t} \in \{1, \ldots, \infty\}$ *and let* $\epsilon > 0$ *denote the relative error. Suppose*

$$\hat{E}[T_r] = \sum_{t=1}^{\hat{t}} \sum_{k=0}^{N_h - 1} e^{-(t-1)\lambda_1} \frac{((t-1)\lambda_1)^k}{k!}.$$

If $\hat{t} > 2\max\{(-\ln(1-e^{-0.5\lambda_1}) + \ln\frac{1}{\epsilon} - \ln\frac{N_h}{\lambda_1})/\lambda_1, 4\frac{N_h - 1}{\lambda_1} + \frac{1}{2}\}$, *then* $|\hat{E}[T_r] - E[T_r]| \leq \epsilon E[T_r]$.

Remark. The above theorem states an closed-form accurate estimation of the ramp up time. And the estimation error ϵ can be arbitrarily small by selecting a large enough $\hat{t}$. It is interesting to observe that $\hat{t}$ increases linearly in $\ln 1/\epsilon$.

Based on Theorem 2, we develop an efficient algorithm to evaluate the ramp up time in Algorithm 2. The computational complexity of Algorithm 2 is $\Theta(\lceil 2\max\{(-\ln(1-e^{-0.5\lambda_1}) + \ln\frac{1}{\epsilon} - \ln\frac{N_h}{\lambda_1})/\lambda_1, 4\frac{N_h-1}{\lambda_1} + \frac{1}{2}\}\rceil N_h) = \Theta(N_h \ln\frac{1}{\epsilon})$. This implies that Algorithm 2 is highly efficient. To apply Algorithm 2, we need to infer model parameters $\lambda_1, N_h, \mathcal{R}$ and γ from data (we will infer them in Section 7).

Algorithm 2. Evaluating Ramp Up Time

Input: Model parameters $\lambda_1, N_h, \mathcal{R}$ and γ. Accuracy factor ϵ. Intrinsic quality Q.
Output: $\hat{E}[T_r]$

```
 1: if R(arg min_{i∈{1,...,m}} |i − Q|) < γ then
 2:     Ê[T_r] = ∞
 3: else
 4:     Ê[T_r] ← 0.
 5:     t̂ ← ⌈2max{(−ln(1 − e^{−0.5λ_1}) + ln(1/ε) − ln(N_h/λ_1))/λ_1, 4(N_h−1)/λ_1 + 1/2}⌉.
 6:     for t = 1 to t̂ do
 7:         for k = 0 to N_h − 1 do
 8:             Ê[T_r] ← Ê[T_r] + e^{−(t−1)λ_1} ((t−1)λ_1)^k / k!
 9:         end for
10:     end for.
11: end if
```

4.3 Algorithm to Evaluate Long Term Profit Gains

To gain some insights in evaluating long term profit gains from data, we first close-form expression for them.

Theorem 3. *Consider an intrinsically average service provider, the long term profit gains can be expressed as*

$$G = \frac{g\lambda_1}{1-\delta}. \tag{10}$$

Consider an intrinsically reputable service provider, the expected long term profit gains can be expressed as

$$G = \frac{g\lambda_2}{1-\delta} + (\lambda_1 - \lambda_2) g \sum_{t=0}^{\infty} \sum_{k=0}^{N_h - 1} \delta^t e^{-\lambda_1 t} \frac{(\lambda_1 t)^k}{k!}. \tag{11}$$

Remark. The implication of this theorem is that as long as we can infer λ_1, λ_2 and N_h from data, we can characterize the long term profit gains G by applying Equations (10) and (11). However, to compute G for intrinsically reputable service providers (i.e., Equation (11)) we have to perform a summation of infinity number of terms. In the following theorem we address this issue via truncation. Let $\hat{G}$ denote an estimation on the long term profit gains.

Theorem 4. *Denote $\hat{t} \in \{1, \ldots, \infty\}$. Consider an intrinsically reputable service provider, i.e., $\mathcal{R}(\arg\min_{i\in\{1,\ldots,m\}}|i-Q|) \geq \gamma$. Suppose*

$$\hat{G} = \frac{g\lambda_2}{1-\delta} + (\lambda_1 - \lambda_2)g\sum_{t=0}^{\hat{t}}\sum_{k=0}^{N_h-1}\delta^t e^{-\lambda_1 t}\frac{(\lambda_1 t)^k}{k!},$$

If $\hat{t} > \max\{(\ln\frac{1-\delta e^{-0.5\lambda_1}}{1-\delta} + \ln\frac{\lambda_1}{\lambda_2-\lambda_1} + \ln\epsilon)/(\ln\delta - 0.5\lambda_1) - 1, 8\frac{N_h-1}{\lambda_1}\}$, *then* $|\hat{G} - G| \leq \epsilon G$.

Remark. The above theorem states a closed-form accurate estimation of the long term profit gains. And the estimation error ϵ can be arbitrarily small by selecting a large enough $\hat{t}$. It is interesting to observe that $\hat{t}$ increases linearly in $\ln 1/\epsilon$.

Based on Theorem 4, we outline an algorithm to evaluate the ramp up time in Algorithm 3. The computationally complexity of Algorithm 3 is $\Theta(\lceil\max\{(\ln\frac{1-\delta e^{-0.5\lambda_1}}{1-\delta} + \ln\frac{\lambda_1}{\lambda_2-\lambda_1} + \ln\epsilon)/(\ln\delta - 0.5\lambda_1) - 1, 8\frac{N_h-1}{\lambda_1}\}\rceil N_h) = \Theta(N_h \ln\frac{1}{\epsilon})$. This shows that Algorithm 3 is highly efficient. To apply Algorithm 3, we need to infer model parameters $\lambda_1, \lambda_2, N_h$ and Q from data (we will infer them in Section 7).

Algorithm 3. Evaluating Long Term Profit Gains

Input: Model parameters $\lambda_1, \lambda_2, N_h, \mathcal{R}, \gamma$ and δ. Accuracy factor ϵ. Intrinsic quality Q.
Output: $\hat{G}$

```
1: if R(argmin_{i∈{1,...,m}} |i − Q|) < γ then
2:    Ĝ = gλ1/(1−δ)
3: else
4:    Ĝ ← gλ2/(1−δ).
5:    t̂ = ⌈max{(ln((1−δe^{−0.5λ1})/(1−δ)) + ln(λ1/(λ2−λ1)) + ln ε)/(ln δ − 0.5λ1) − 1, 8(Nh−1)/λ1}⌉
6:    for t = 0 to t̂ do
7:       for k = 0 to Nh − 1 do
8:          Ĝ ← Ĝ + (λ1 − λ2) g δ^t e^{−λ1 t} (λ1 t)^k / k!
9:       end for
10:   end for.
11: end if
```

4.4 Summary

We developed a baseline data analytical framework to characterize the ramp up time T_r and the expected long term profit gains G from data. Note that our framework so far assumes a perfect scenario that customers never commit errors in perceiving service providers' intrinsic quality, i.e., $\hat{Q} = Q$. However, customers may commit errors due to human factors like biases, preferences, etc. We next extend our data analytical framework to incorporate such human factors.

5 Human Factors

We now present a probabilistic model to capture human factors in rating such as biases and preferences. To incorporate them into our data analytical framework (stated in Algorithm 1), and we show that it is computationally difficult to evaluate $E[T_r]$ and G from any data set due to human factors. This computational challenge motivates us to design efficient randomized algorithms which have theoretical performance guarantees to approximate $E[T_r]$ and G in Section 6.

5.1 Model for Human Factors

Customers may have personal preferences in expressing feedback ratings due to various human factors, e.g., inherent biases. More precisely, a critical customer may assign lower ratings while a lenient customer may assign higher ratings.

To illustrate, let us focus on just one service provider which we denote by S. S provides "high quality" ("low quality") services but may receive low (high) rating. We use the following probabilistic model to capture the collective rating behavior under such personal preferences

$$\Pr[S \text{ receives a level } i \text{ rating}] = \eta_i, \text{ for all } i = 1, \ldots, m,$$

where η_i denotes the probability of receiving a level i rating and $\sum_{i=1}^m \eta_i = 1$. One can vary the mean of the m-tuple $(\eta_1, \ldots, \eta_m)$ to reflect different level of personal preferences. The higher (lower) the mean implies that customers are more likely to be lenient (critical) ones. We point out that when all customers are unbiased, then $(\eta_1, \ldots, \eta_m)$ reflects the intrinsic quality of a service provider. The impact of inherent biases is to shift $(\eta_1, \ldots, \eta_m)$ towards a higher or lower mean. One of our objectives is to examine the impact of such human factors on the efficiency of online reputation system. We next extend Definition 6 to incorporate such human factors.

Definition 7. *In the presence of human factors, we say a service provider is intrinsically reputable if and only if* $\sum_{i=1}^m \eta_i \mathcal{R}(i) \geq \gamma$, *otherwise we say a service provider is intrinsically average.*

To incorporate them into the baseline data analytical framework (stated in Algorithm 1), we first need to infer some extra parameters, i.e., $\eta_1, \ldots, \eta_m$, in step 1 of Algorithm 1. Then in step 2 of Algorithm 1 we need to evaluate ramp up time and long term profit gains in the presence of human factors. We outline this extended data analytical framework in Algorithm 4.

Algorithm 4. Improved Data Analytical Framework

1: **Parameter inference.** (1) Infer rating distribution $\eta_1, \ldots, \eta_m$ from data. (2) Infer model parameters $m, \gamma, \mathcal{R}, N_h, \lambda_1$ and λ_2 from data.
2: **Quantifying system efficiency.** Evaluating the ramp up time T_r and expected long term profit gains G in the presence of human factors, i.e., $\eta_1, \ldots, \eta_m$.
3: **Uncover system deficiencies.** Uncover deficiencies of online reputation systems via studying the ramp up time T_r and expected long term profit gains G.

We will present the details of step 1 and step 3 of the above framework in Section 7. In this section we focus on addressing step 2 of Algorithm 4.

5.2 Two Rating Levels

We first consider a special case of two rating levels, i.e., $m = 2$. This will illustrate the key idea of our derivation as well as its underlying computational complexity.

- *Ramp up time:* Note that the ramp up time T_r is a random variable due to dynamics in a reputation updating process. The following lemma states the closed-form expected ramp up time $E[T_r]$.

Lemma 1. *Suppose the number of rating levels is two, i.e., $m = 2$. The expected ramp time can be expressed as*

$$E[T_r] = \sum_{t=0}^{\infty} t \sum_{(N_1(0),N_2(0),\ldots,N_1(t-1),N_2(t-1))\in\mathbb{N}^{2t-2}} \prod_{\ell=0}^{t-1} \phi(N_1(i), N_2(i), \lambda(i)) \mathbf{I}_{\{r(t)\geq\gamma, n_1(t)+n_2(t)\geq N_h\}} (1 - \mathbf{I}_{\{\exists j<t, r(t)\geq\gamma, n_1(j)+n_2(j)\geq N_h\}}), \tag{12}$$

where $\lambda(t)$ is derived in Equation (5), and function ϕ is $\phi(x, y, z) = \binom{x+y}{y}\eta_1^x \eta_2^y e^{-z} z^{x+y}/(x + y)!$.

Theorem 5. *The computational complexity of evaluating $E[T_r]$ derived in Equation (12) is $\Omega(\sum_{t=0}^{\infty} t)$.*

- *Long term profit gains:* Note that $G \triangleq E[\sum_{t=0}^{\infty} \delta^t gN(t)] = \sum_{t=0}^{\infty} \delta^t gE[N(t)]$. This implies that we have to compute $E[N(t)]$ for all $t = 0, 1, \ldots, \infty$. The following lemma states the analytical expression for G.

Lemma 2. *When the number of rating levels is two ($m = 2$), the long term profit gains G can be expressed as*

$$G = \sum_{t=0}^{\infty} \delta^t g \Big\{ \lambda_1 \Big\{ \sum_{n_1(t)=0}^{\infty} \sum_{n_2(t)=0}^{\lfloor n_1(t)\alpha \rfloor} \sum_{\sum_{i=0}^{t-1} N_1(i)=n_1(t)} \sum_{\sum_{i=0}^{t-1} N_2(i)=n_2(t)} + \sum_{n_1(t)+n_2(t)<N_h} \sum_{\sum_{i=0}^{t-1} N_1(i)=n_1(t)} \sum_{\sum_{i=0}^{t-1} N_2(i)=n_2(t)} - \sum_{n_1(t)=0}^{N_h} \sum_{n_2(t)=0}^{\min\{\lfloor n_1(t)\alpha \rfloor, N_h-n_1(t)\}} \sum_{\sum_{i=0}^{t-1} N_1(i)=n_1(t)} \sum_{\sum_{i=0}^{t-1} N_2(i)=n_2(t)} \Big\} + \lambda_2 \Big\{ \sum_{n_1(t)=0}^{\infty} \sum_{n_2(t)=\max\{N_h-n_1(t), \lceil n_1(t)\alpha \rceil\}}^{\infty} \sum_{\sum_{i=0}^{t-1} N_1(i)=n_1(t)} \sum_{\sum_{i=0}^{t-1} N_2(i)=n_2(t)} \Big\} \Big\} \prod_{i=0}^{t-1} \phi(N_1(i), N_2(i), \lambda(i)), \tag{13}$$

where $\lambda(t)$ is derived in Equation (5), function ϕ is $\phi(x, y, z) = \binom{x+y}{y}\eta_1^x \eta_2^y e^{-z} z^{x+y}/(x + y)!$, and $\alpha = \frac{\gamma-\mathcal{R}(1)}{\mathcal{R}(2)-\gamma}$.

Theorem 6. *The computational complexity of evaluating G derived in Equation (13) is $\Omega(\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \binom{i+j}{j})$.*

- *Summary of observations:* Let us summarize our observations thus far in analyzing the special case of two rating levels ($m = 2$): (1) We derived analytical expressions for $E[T_r]$ and G; (2) the analytical expressions indicate extremely large computational complexity (based on Theorems 5 and 6), and they are computationally infeasible. To overcome such problem, we propose efficient randomized algorithms (in Section 6) which have theoretical performance guarantees in computing $E[T_r]$ and G.

5.3 Extensions to More Than Two Rating Levels

One can extend Lemmas 1 and 2 to obtain closed-form expressions for $E[T_r]$ and G. Due to page limit, we will not present the derivation here but it is reasonable to expect that the underlying complexity is huge, so it makes naive computation of $E[T_r]$ and G impractical. Let us focus on developing a practical approach to tackle the challenges in evaluating $E[T_r]$ and G.

6 Randomized Algorithms

We showed in the last section that it is computationally infeasible to evaluate $E[T_r]$ and G in the presence of human factors. Here, we propose computationally efficient randomized algorithms which have theoretical performance guarantees to approximate $E[T_r]$ and G.

6.1 Approximating Ramp Up Time

One can compute $E[T_r]$ via stochastic monte carlo methods [19]. The basic idea of stochastic monte carlo methods is estimating a probabilistic metric via sample average. Specifically, we can simulate the reputation updating process for $K \in \mathbb{N}$ rounds. Each round produces one sample of the ramp up time (T_r). We use the average of these K samples, which we denote as $\hat{E}[T_r]$, to estimate $E[T_r]$. For each sample path of T_r, we simulate the reputation updating process until a service provider ramps up. The stochastic monte carlo method is depicted in Algorithm 5. More concretely, line 2 initializes the setting to consider a new seller. Line 3 checks whether the ramp up condition is satisfied and it stops the iteration until the condition is satisfied. Line 6 generates the number of transactions and updates the total number of ratings. Line 7 to 10 generate a feedback rating for each transaction. Line 11 updates the average score.

Algorithm 5. Randomized Algorithm for $E[T_r]$

Input: Model parameters $\eta_1, \ldots, \eta_m$, λ_1, N_h, m and $\mathcal{R}$.
Output: $\hat{E}[T_r]$

```
1:  for i = 1 to K do
2:      r ← 0, n_1 ← 0, ..., n_m ← 0, T_r^i ← 0
3:      while r < γ or Σ_{ℓ=1}^m n_ℓ < N_h do
4:          T_r^i ← T_r^i + 1
5:          λ ← λ_1
6:          N ~ Poisson(λ)
7:          for j = 1 to N do
8:              ℓ ~ Multinomial(η_1, ..., η_m)
9:              n_ℓ ← n_ℓ + 1
10:         end for
11:         r ← Σ_{k=1}^m n_k R(k) / Σ_{ℓ=1}^m n_ℓ
12:     end while
13: end for
14: Ê[T_r] ← Σ_{i=1}^K T_r^i / K
```

We next analyze the computational complexity (Theorem 7) of Algorithm 5 and derive the number of simulation rounds (K) needed to guarantee an accurate value of $E[T_r]$ (Theorem 8).

Theorem 7. *Suppose a service provider is intrinsically reputable, i.e., $\sum_{i=1}^{m} \eta_i \mathcal{R}(i) \geq \gamma$. The expected computational complexity for Algorithm 5 is $O(KN_h + K\frac{(\mathcal{R}(m)-\mathcal{R}(1))^4}{(\sum_i \eta_i \mathcal{R}(i)-\gamma)^4})$.*

Theorem 8. *Suppose a service provider is intrinsically reputable, i.e., $\sum_{\ell=1}^{m} \eta_\ell \mathcal{R}(\ell) > \gamma$. If the number of simulation rounds satisfies $K = O(\frac{1}{\xi\epsilon^2 N_h^2}(\frac{\mathcal{R}(m)-\mathcal{R}(1)}{\sum_{\ell=1}^{m} \eta_\ell \mathcal{R}(\ell)-\gamma})^6)$, then Algorithm 5 guarantees that $|\hat{E}[T_r] - E[T_r]| \leq \epsilon E[T_r]$ with probability of at least $1 - \xi$, where $\xi \geq 0$ denotes the fail probability.*

To illustrate the bound of K. Let us consider $\xi = 0.1$ and $\epsilon = 0.1$, i.e., to guarantee the approximation error is less than $0.1E[T_r]$ with probability of at least 0.9. Suppose $m = 5$, $\mathcal{R}(i) = i$, $\sum_{\ell=1} \eta_\ell \mathcal{R}(\ell) - \gamma = 0.2$ and $N_h = 100$. Then we have that $K = 2.44 \times 10^7$.

Remark. Algorithm 5 is computationally efficient and can determine $E[T_r]$ with arbitrarily small error.

6.2 Approximating Long Term Profit Gains

We compute G via stochastic monte carlo methods. We simulate our model for $K \in \mathbb{N}$ rounds. Each round produces one sample of the long term profit gains G. The average of these K samples, which we denote as $\hat{G}$, to estimate G. Note that to obtain one sample of G, one needs to simulate all transactions completed in time slot $0, 1, \ldots, \infty$. This is computationally expensive (and theoretically infeasible). To address this challenge, we show that one only needs to simulate M time slots, i.e., time slot 0 to M, and can tightly "bound" the error in estimating G. This stochastic monte carlo method is presented in Algorithm 6. More concretely, line 2 initializes the setting to consider a new seller. Line 3 stops the loop when the number of time slots hits M. Line 4 to 7 generate the transaction rate. Line 8 generates the number of transactions in a time slot. Line 9 updates the long term profit gain. Line 10 to 13 generates the feedback rating for each transaction and updates the total number of ratings. Line 14 updates the average score.

Algorithm 6. Randomized Algorithm for G

Input: Model parameters $\eta_1, \ldots, \eta_m$, λ_1, N_h, m and $\mathcal{R}$.
Output: $\hat{G}$
1: **for** $i = 1$ to K **do**
2: $r \leftarrow 0, n_1 \leftarrow 0, \ldots, n_m \leftarrow 0, G_i \leftarrow 0$
3: **for** $t = 0$ to M **do**
4: **switch** $(r, \sum_{\ell=1}^m n_\ell)$ **do**
5: **case** $r < \gamma$ or $\sum_{\ell=1}^m n_\ell < N_h$: $\lambda = \lambda_1$
6: **case** $r \geq \gamma$ and $\sum_{\ell=1}^m n_\ell \geq N_h$: $\lambda = \lambda_2$
7: **ends switch**
8: $N \sim Poisson(\lambda)$
9: $G_i \leftarrow G_i + Ng\delta^t$
10: **for** $j = 1$ to N **do**
11: $\ell \sim Multinomial(\eta_1, \ldots, \eta_m)$
12: $n_\ell \leftarrow n_\ell + 1$
13: **end for**
14: $r \leftarrow \sum_{k=1}^m n_k \mathcal{R}(k) / \sum_{\ell=1}^m n_\ell$
15: **end for**
16: **end for**
17: $\hat{G} \leftarrow \sum_{i=1}^K G_i / K$

We now analyze the computational complexity (Theorem 9) of Algorithm 6 and derive the appropriate K and M to guarantee an accurate approximation of G (Theorem 10).

Theorem 9. *The expected computational complexity for Algorithm 6 is* $O(KM\lambda_2)$.

Theorem 10. *If the number of simulation rounds satisfies* $K = O(\frac{1}{\epsilon^2} \frac{\lambda_2}{\lambda_1^2} \frac{1}{\xi})$, *and* M *satisfies* $M = O(\ln \frac{\epsilon \lambda_1}{\lambda_2} / \ln \delta)$, *then Algorithm 6 guarantees that* $|\hat{G} - G| \leq \epsilon G$ *holds with probability of at least* $1 - \xi$.

Remark. Algorithm 6 is computationally efficient and can compute G with arbitrarily small error.

TABLE 2
Statistics for Reputation Rating Datasets

	# of service providers	# of ratings
eBay	4,586	19,217,083
Helpouts	858	10,454
Amazon	32,888	5,066,070
TripAdvisor	11,543	3,114,876

7 Experiments on Real-World Data

We present experimental results on reputation rating for the datasets from eBay and Google Helpouts. We first infer model parameters from the data, and input these inferred values to our framework so to characterize the ramp up time and long term profit gains. We show that the existing ramp up time in eBay and Google Helpouts are long: around 791 days and 1327 days respectively. This shows the inefficiency of online reputation systems since the ramp up time can significantly influence the long term profit gains, i.e., a 80 percent reduction on ramp up time leads to 80 percent (eBay) and 100 percent (Google Helpouts) improvement in long term profit gains respectively. Lastly, we discover from the data that around 78.7 percent sellers have ramped up in eBay, but only 1.5 percent workers have ramped up in Google Helpouts.

7.1 Datasets

We crawled historical online reputation data (refer to Table 2). from four representative applications of reputation systems, i.e., electronic commerce (eBay), crowdsourcing (Google Helpouts), product review (Amazon), travel website (TripAdvisor).

eBay. Eay is a popular electronic commerce system, where buyers purchase products from online stores and when a transaction is completed, a buyer expresses a rating to indicate whether a seller is trustworthy or not. It deployes a three-level cardinal metric, i.e., $\{-1$ ("negative"), 0 ("neutral"), 1 ("positive")$\}$. Ratings are public and accessible to all buyers and sellers. We crawled the historical ratings of 4,586 sellers received from the first day that a seller joins the eBay till April 2013.

Google Helpouts. Google Helpouts provides online crowdsourcing services. In Google Helpouts, service providers (or workers) offer various types of services, e.g., teaching piano, teaching cooking, etc. Workers advertise the service that they can provide, e.g., a service provider provides piano teaching service. Requesters select workers to provide a service based on workers' reputation. When a transaction is completed, requesters express a rating to indicate the quality of the service using a five-level cardinal rating metric $\{1$ ("Terrible"), 2 ("Poor"), 3 ("Average"), 4 ("Very good"), 5 ("Excellent") $\}$. Ratings and overall rating statistics for each worker are public and accessible to all users. We crawled historical transactions of 858 workers received from the first day that a worker joins Google Helpouts till January 2015.

TripAdvisor. TripAdvisor is a popular travel website, where users express ratings to hotels, restaurants, etc., to reflect the quality (or reputation) of these items. It uses the

TABLE 3
Inferred γ, N_h, λ_1 and λ_2

	m	γ	N_h	λ_1	λ_2
eBay	3	0.5	100	0.1742	2.4883
Helpouts	5	4	100	0.0689	0.4076
Amazon	5	4	100	0.1067	0.8916
TripAdvisor	5	4	100	0.0723	0.3831

TABLE 4
Inferred m and $\eta_1, \ldots, \eta_m$

	$\{\eta_1, \ldots, \eta_m\}$
eBay	$\{0.0023, 0.0034, 0.9943\}$
Helpouts	$\{0.0150, 0.0039, 0.0211, 0.0950, 0.8650\}$
Amazon	$\{0.0452, 0.0335, 0.0674, 0.1967, 0.6572\}$
TripAdvisor	$\{0.0172, 0.0295, 0.0822, 0.3242, 0.5468\}$

same rating system as that of Google Helpouts. We crawled ratings of 11,543 hotels received from the first day that a hotel joins this web site till April 2013.

Amazon. Amazon is a typical product review system, where users express ratings (or reviews) on products to reflect the product quality (or reputation). It uses the same rating system as that of Google Helpouts. We crawled ratings of 32,888 products received from the first day that a product joins this web site till April 2013.

7.2 Inferring Model Parameters

We now infer the parameters of our model, which are summarized in Tables 4 and 3. A time slot is one day. From Table 2 we observe that eBay adopts a three-level rating metric and the other three adopt a five-level rating metric respectively. Hence we have $m = 3$ for eBay and $m = 5$ for Google Helpouts, Amazon and TripAdvisor. Furthermore, from Table 2, we obtain the rating map $\mathcal{R}$ as $\{\mathcal{R}(1) = -1, \mathcal{R}(2) = 0, \mathcal{R}(3) = 1\}$ for eBay and $\{\mathcal{R}(1) = 1, \ldots, \mathcal{R}(5) = 5\}$ for Google Helpouts, Amazon and TripAdvisor. Each rating level is associated with a physical meaning, e.g., in eBay, we have $\{-1$ ("negative"), 0 ("neutral"), 1 ("positive")$\}$. We therefore set the reputation threshold as $\gamma = \frac{0+1}{2} = 0.5$. Similarly, for Google Helpouts, Amazon and TripAdvisor we have $\{1$ ("Terrible"), 2 ("Poor"), 3 ("Average"), 4 ("Very good"), 5 ("Excellent")$\}$. We set the reputation threshold as $\gamma = 4$. Xie and Lui studied rating sufficiency conditions for online rating systems [31], and they revealed that around 100 ratings can reflect the true quality of a product. In other words, the aggregate rating is statistically significant if an service provider has around 100 ratings. We therefor set $N_h = 100$. We will justify that this selection on γ is reasonable in Section 7.3 via extensive experiments on our datasets. Apply the inferred γ and N_h on our datasets, we infer the transactions' rate λ_1 (λ_2) as average number of transactions' per day completed by "average" ("reputable") service providers

$$\lambda_1 = \frac{\#[\text{transations by average service providers}]}{\#[\text{days to accumulate these transactions}]}, \quad (14)$$

$$\lambda_2 = \frac{\#[\text{transations by reputable service providers}]}{\#[\text{days to accumulate these transactions}]}. \quad (15)$$

Applying these two rules on our data set, we obtain the transactions' rate before ramping up and after ramping up. We summarize them in Table 3. Note that η_i denotes the probability of an intrinsically reputable service provider receives a level i rating. We say a service provider is intrinsically reputable if it has at least $N_h = 100$ ratings and its average rating is above the inferred γ. We therefore infer η_i as the fraction of level i ratings across all intrinsically reputable service providers

$$\eta_i = \frac{\#[\text{level } i \text{ ratings across all intrinsically reputable SPs}]}{\#[\text{ratings across all intrinsically reputable SPs}]},$$

where *SPs* denote service providers for short. Performing this rule on our datasets, we have η_i presented in Table 4.

7.3 Justifications of the Inferred Parameters

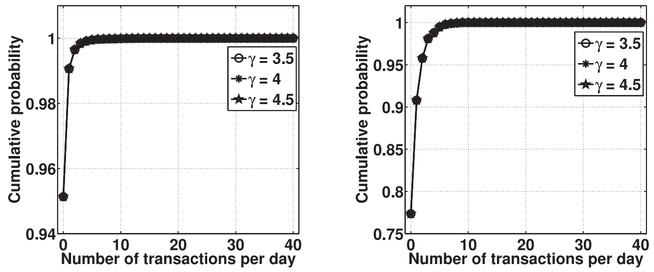
We conduct extensive experiments to justify that the inferred value of two building block parameters, i.e., γ and N_h are reasonable. These two parameters determine all other parameters like transactions' rate, the cumulative probability mass function of the number of transactions that arrive to a service provider per day. Our experimental results show that the inferred value of γ is an accurate estimation on its true value and the inferred threshold N_h is a typical value and selecting it does not loss any generality.

In particular, we study the impact of γ and N_h on the distribution of the the number of transactions per day to a service provider. This is because this distribution is the most fundamental one, since it determines the transactions rate and the per-day profit gains. With the above inferred parameters, we infer the cumulative probability mass function of the number of transactions that a service provider (average service provider and reputable service provider) receives in one day from data. Consider average labeled service providers, we infer the corresponding probability mass function via computing the fraction of days that they receive at most i transactions, where $i = 0, 1, \ldots, \infty$, i.e.,

$$\begin{aligned}&\Pr[\text{ASP receive at most } i \text{ transactions}]\\&= \frac{\#[\text{days that ASP receive at most } i \text{ transactions}]}{\sum_j \#[\text{days that ASP receive } j \text{ transactions}]},\end{aligned}$$

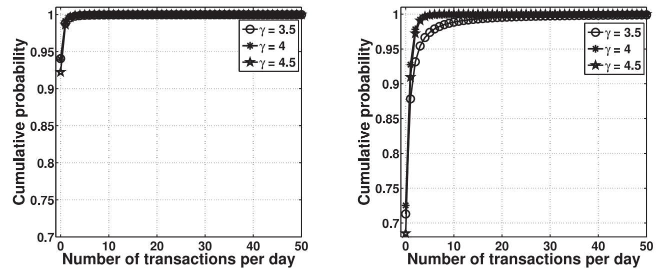
where ASP refers to average service providers. Similarly, we infer the cumulative probability function for reputable service providers.

We first study the impact of γ on the inferred the cumulative probability mass function of the number of transactions per day to a service provider almost remain unchanged. For Google Helpouts, Amazon and TripAdvisor we vary the value of γ from small to large, i.e., $\gamma = 3.5$, $\gamma = 4$ and $\gamma = 4.5$. Notice that in experience $\gamma = 3.5$ ($\gamma = 4.5$) is so small (large) for Google Helpouts that the true value of γ should be smaller (larger) than it. For eBay, we vary the value of γ from small to large, i.e., $\gamma = 0.1$, $\gamma = 0.5$ and $\gamma = 0.9$. Notice that in experience $\gamma = 0.1$ ($\gamma = 0.9$) is so small (large) for Google Helpouts that the true value of γ should be smaller (larger) than it. For each value of γ we infer the cumulative probability mass function, setting N_h



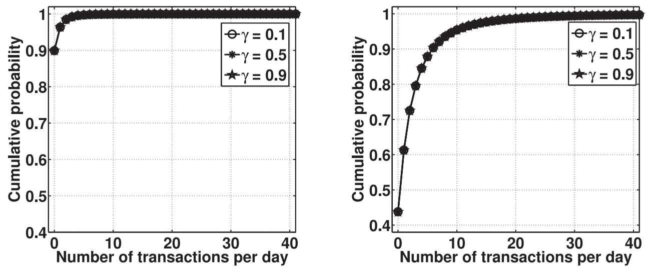
(a) Before Ramping Up (b) After Ramping Up

Fig. 1. Impact of γ on the transactions' distribution in Google Helpouts.



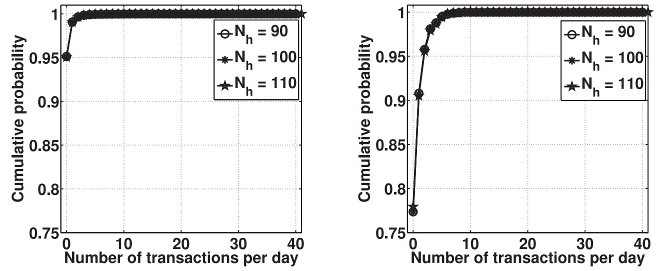
(a) Before Ramping Up (b) After Ramping Up

Fig. 4. Impact of γ on the transactions' distribution in TripAdvisor.



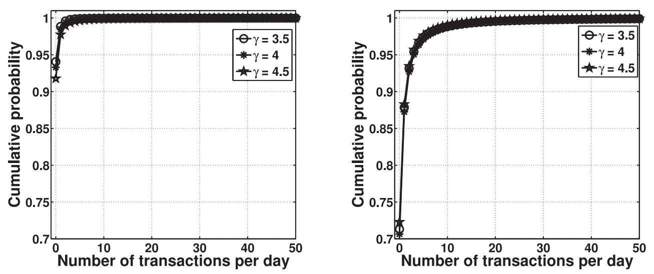
(a) Before Ramping Up (b) After Ramping Up

Fig. 2. Impact of γ on the transactions' distribution in eBay.



(a) Before Ramping Up (b) After Ramping Up

Fig. 5. Impact of N_h on the transactions' distribution in Google Helpouts.



(a) Before Ramping Up (b) After Ramping Up

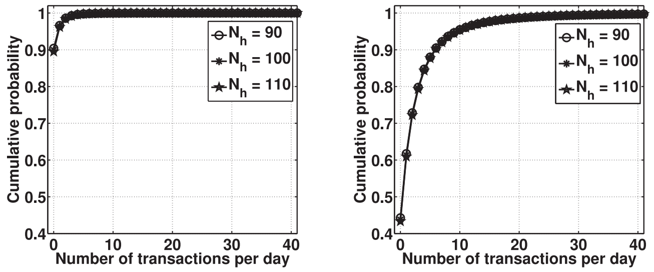
Fig. 3. Impact of γ on the transactions' distribution in Amazon.

as that inferred in Section 7.2. Fig. 1 depicts the cumulative probability functions for Google Helpouts, where the horizontal axis presents the number of transactions that a service provider receives in one day, and the vertical axis shows the corresponding cumulative mass probability. Consider average labelled service providers in Google helpouts, Fig. 1a presents the probability mass functions for average labeled service providers in Google helpouts. It contains three curves corresponding to $\gamma = 3.5, \gamma = 4$ and $\gamma = 4.5$ respectively. One can observe that these three curves overlap. This means that the cumulative probability mass function for average labeled service providers in Google Helpouts remains unchanged as we vary γ from small to large. This statement also holds for reputable service providers in Google helpouts as one can observe in Fig. 1b. Therefore, for Google helpuots the cumulative probability function for the number of transactions that a service provider receives in one day remains unchanged as γ varies from small to large. This shows that the inferred γ is an accurate estimation on its true value. This statement also holds for the eBay, Amazon and TripAdvisor dataset, as shown in Figs. 2, 3, and 4.

We now study the impact of N_h on the inferred the cumulative probability mass function of the number of transactions per day to a service provider. We will show that as we perturb N_h by one percent from its inferred value in Section 7.2, the inferred model parameters, i.e., transactions' rate λ_1, λ_2 and the cumulative probability mass function of the number of transactions per day, varies slightly. This means that the inferred N_h is a representative value and selecting it does not loss any generality. Again, it boils down to show that the cumulative probability mass function varies slightly. We perturb the value of N_h by one percent from its inferred value, i.e., set $N_h = 90, 100$ and 110. For each value of N_h we infer the cumulative probability mass function of the number of transactions received in one day setting γ as that inferred in Section 7.2. Fig. 5 depicts the cumulative probability functions inferred form the Google Helpouts dataset, where the horizontal axis presents the number of transactions that a service provider receives in one day, and the vertical axis shows the corresponding cumulative mass probability. Fig. 5a presents the corresponding probability mass functions for average labeled service providers. It contains three curves corresponding to $N_h = 90, N_h = 100$ and $N_h = 110$ respectively. One can observe that these three curves almost overlap. This means that the cumulative probability mass function for average labeled service providers in Google Helpouts varies slightly as we perturb N_h by one percent from its inferred value. This statement also holds for reputable service providers in Google helpouts as shown in Fig. 5b. Therefore, for Google helpuots the cumulative probability function for the number of transactions per day varies slightly as as we perturb N_h by one percent from its inferred value. The same observations can be obtained for the eBay, Amazon and TripAdvisor dataset as shown in Figs. 6, 7, and 8.

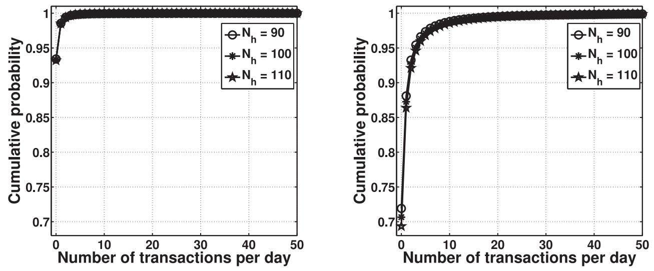
7.4 Model Validation

We validate that the number of transactions that a service provider receives in one day follows a Poisson distribution.



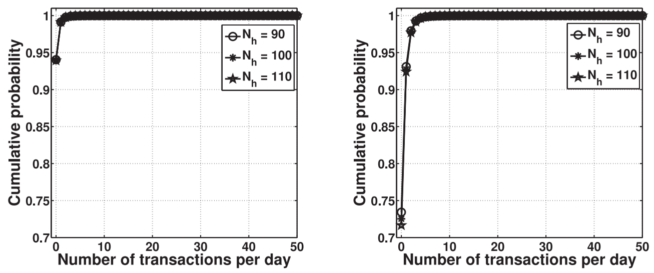
(a) Before Ramping Up (b) After Ramping Up

Fig. 6. Impact of N_h on the transactions' distribution in eBay.



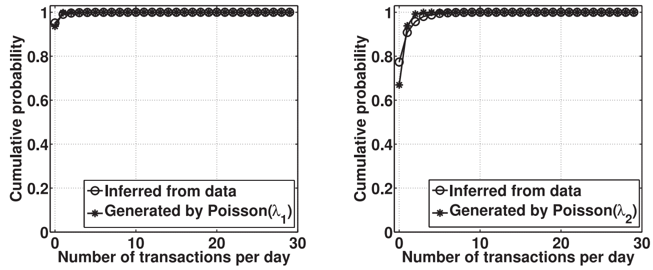
(a) Before Ramping Up (b) After Ramping Up

Fig. 7. Impact of N_h on the transactions' distribution in Amazon.



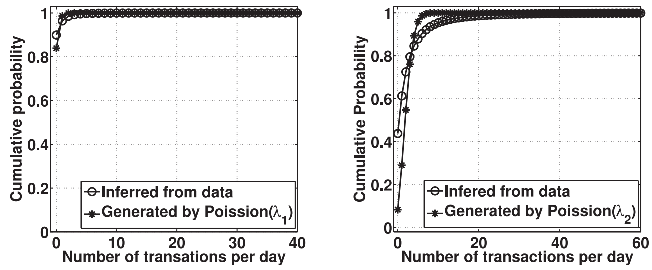
(a) Before Ramping Up (b) After Ramping Up

Fig. 8. Impact of N_h on the transactions' distribution in TripAdvisor.



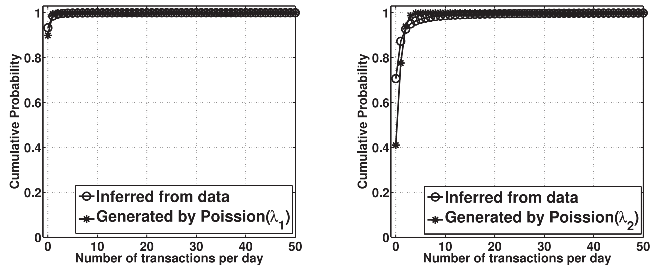
(a) Before Ramping Up (b) After Ramping Up

Fig. 9. Transaction distribution validation for Google Helpouts.



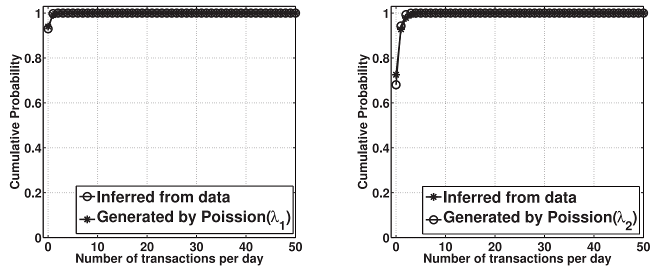
(a) Before Ramping Up (b) After Ramping Up

Fig. 10. Transaction distribution validation for eBay.



(a) Before Ramping Up (b) After Ramping Up

Fig. 11. Transaction distribution validation for Amazon.



(a) Before Ramping Up (b) After Ramping Up

Fig. 12. Transaction distribution validation for TripAdvisor.

In particular, we show that the inferred probability mass function are almost the same as the probability mass function of a Poisson distributions with the inferred transaction rates λ_1 and λ_2. Fig. 9 depicts the cumulative probability mass functions, where the horizon axis presents the number of transactions that a service provider receives in one day and the vertical axis shows the corresponding cumulative probability. It has two sub-figures corresponding to average labeled and reputable service provider respectively. Each sub-figure contains two curves corresponding cumulative probability mass functions inferred from day and generated by the Poisson distribution respectively. From Fig. 9a we observe that these two curves almost overlap. This shows that for average labeled service providers in Google helpouts, the distribution of the number of transactions per day follows a Poisson distribution. This statement also holds for reputable service providers as one can observe from Fig. 9b. Therefore for Google Helpouts, the distribution of the number of transactions per day follows a Poisson distribution. This statement also holds for the eBay, Amazon and TripAdvisor dataset as shown in Figs. 10, 11, and 12.

7.5 Characterizing Ramp Up Time

We apply Algorithm 5 to compute the ramp up time. Applying Theorems 7 and 8, we set $K = 10^8$ since it guarantees $|\hat{E}[T_r] - E[T_r]| \leq 0.01E[T_r]$ with probability at least 0.99. We input the above inferred parameters to Algorithm 5 and obtain the expected ramp up times for eBay, Google Helpouts, Amazon and TripAdvisor respectively and they are stated in Table 5. From Table 5 we observe that for Google Helpouts the expectation of the ramp up time is 1454 days. This means that on average, a worker needs to

TABLE 5
Expected Ramp Up Time $E[T_r]$

	Helpouts	eBay	Amazon	TripAdvisor
$E[T_r]$	1454 (days)	576 (days)	943 (days)	1383 (days)

spend 1454 days to get ramped up. This is quite a long time, which may lead to that some workers get dropped out and discourages new workers to join. The implication for Google Helpouts is that the reputation imposes a negative impact on increasing the user population and it needs extra strategies to recruit new workers effectively so as to increase user population in its early stage. Consider eBay, the expectation of the ramp up time is $E[T_r] = 576\,(\text{days})$. Namely, on average, it takes 576 days for a seller to get ramped up, which is a long duration. As a consequence, some sellers may get dropped out before getting ramped up or they shit to other online selling platforms and new sellers may get discouraged to join. The implication for the eBay is that it needs to deploy some mechanisms to help new sellers to attract buyers so as to get ramped up quickly. For Amazon and TripAdvisor, the expected ramp time are 943 days and 1383 days respectively. This implies that the Amazon (or TripAdvisor) website needs to deploy incentive mechanisms to attract users to assign ratings or reviews to new products (or hotels) so that high quality products can be identified in a shorter time. This will benefit users which in turn benefit the website by attracting more users. Lastly, from Table 5, we also observe that the ramp up time for eBay is the shortest and the ramp up time for Google Helpouts is the longest. This implies that the online reputation system in eBay is the most efficient one while the online reputation system in Google Helpouts is the most inefficient one.

We now apply our framework to investigate the fraction of service providers that have got ramped up. This fraction service providers is an important indicator for an Internet service platform and based on it we obtain deeper insights and implications to improve the Internet service platforms. Formally, let f_{ramp} denote fraction of service providers that have got ramped up

$$f_{ramp} \triangleq \frac{\#[\text{service providers having } r \geq \gamma, \sum_{i=1}^{m} n_i \geq N_h]}{\#[\text{service providers}]}.$$

Recall that the inferred condition for a service provider to get ramped up is $(\gamma, N_h) = (0.5, 100)$ for eBay, and $(\gamma, N_h) = (4, 90)$ for Google Helpouts, Amazon and TripAdvisor. Applying this condition to our dataset, we obtain the empirical values of f_{ramp}, which is presented in Table 6. From Table 6 we observe that for Google Helpouts we have $f_{ramp} = 1.5\%$. Namely, only 1.5 percent of workers have got ramped up. One possible reason is that Google Helpouts is only around one year old and its ecosystem is still at the infancy. Recall that the ramp up time is quite long, i.e., 1454 days. The long ramp up time and small f_{ramp} uncovers a deficiency of the reputation system of Google Helpouts in increasing the number of workers. This deficiency uncovers a key reason why Google Helpouts was eventually shut down in April 2015. Consider eBay, we observe that $f_{ramp} = 84.5\%$. This means that a large fraction of users have got ramped up. One reason is that eBay is over ten years old and sellers who remain in eBay have sufficient time to ramp up. There are still many workers have not got ramped up, i.e, around 16 percent. Recall that ramp up time is also long, i.e., 576 days. This uncovers a key reason why eBay is under a significant user loss [12], [23], [26]. For Amazon and TripAdvisor we have $f_{ramp} = 21.0\%$ and $f_{ramp} = 26.8\%$. Namely, most products (or hotels) have not got ramped up yet in Amazon (or TripAdvisor). Recall that the ramp up time for Amazon and TripAdvisor are long, i.e., 943 and 1383 days respectively. This implies that to identify more high quality products (or hotels) in a shorter time, the Amazon (or TripAdvisor) website needs to incentivize users to assign ratings to averaged labeled products (or hotels).

TABLE 6
Fraction of Ramped Up Service Providers

	Helpouts	eBay	Amazon	TripAdvisor
f_{ramp}	1.4 %	84.5 %	21.0%	26.8%

TABLE 7
Long Term Profit Gains ($g=1, \delta=0.999$)

	G	$G_{\max}$	$G/G_{\max}$
Helpouts	147.5	402.8	36.61%
eBay	1477.3	2472.6	59.74%
Amazon	413.8	884.0	46.81%
TripAdvisor	151.1	377.6	40.01%

7.6 Characterizing Long Term Profit Gains

Now we study the impact of ramp up time on the long term profit gains. In particular, we would like to know to what extend reducing ramp up time can improve the long term profit gains. Through this we reveal whether reducing ramp up time is meaningful for service providers. We apply Algorithm 6 to compute long term profit gains. We set the discounting factor as $\delta = 0.999$, the unit profit gain to be $g = 1$. Applying Theorems 9 and 10 we set $K = 10^8$ and $M = 50000$ since they guarantee $|\hat{G} - G| \leq 0.01G$ with probability at least 0.99. We input the inferred parameter into Algorithm 6 to compute G. To show the potential improvement of long term profit gains via reducing the ramp up time, we also compute the theoretical maximum long term profit gains denoted by $G_{\max}$, which is attained when N_h is equal to zero. Table 7 presents numerical results on $G, G_{\max}$ and $G/G_{\max}$. One can observe that for the long term profit gains, Google Helpouts only achieves 36.61 percent, eBay only achieves 59.74 percent, Amazon achieves 46.81 percent and TripAdvisor achieves 40.01 percent of its maximum possible value $G_{\max}$. This shows that there is a great opportunity to improve the long term profit gains via reducing the ramp up time. Namely, it meaningful to reduce the ramp up time for service providers.

Let us now study how the ramp up time influences the long term profit gains, since this will give us important insights on why some service providers drop out, and why some new service providers participate. We examine the impact of T_r on G by varying T_r. We consider the scenario that the Internet service company can control N_h, and we want to find our how G can be improved if we reduce T_r (T_r can be reduced by reducing N_h). We define reduction ratio

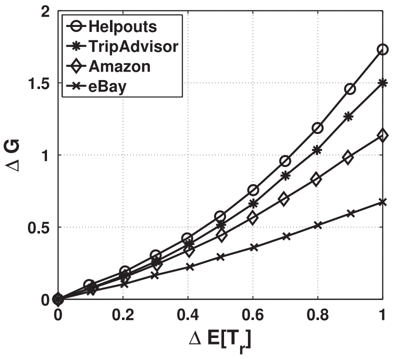


Fig. 13. Impact of ramp up time T_r on G.

of ramp up time

$$\Delta E[T_r] = (E[T_r] - \widetilde{E}[T_r])/E[T_r],$$

where $E[T_r]$ is the ramp up time without reduction on N_h, $\widetilde{E}[T_r]$ is the new ramp up time with some reduction on N_h. We define improvement ratio on the long term profit gains

$$\Delta G = (\widetilde{G} - G)/G,$$

where G is the long term profit gains without reduction on N_h, $\widetilde{G}$ is the new long term profit gains with some reduction on N_h. Fig. 13 shows the impact of $\Delta E[T_r]$ on ΔG, where the horizontal axis represent $\Delta E[T_r]$ and the vertical axis shows the corresponding ΔG. One can observe that as we reduce ramp up time, we increase the long term profit gains. When we reduce $E[T_r]$ by 80 percent, the improvement of G is around 100 percent for Google Helpouts, 50 percent for eBay, and around 80 percent for Amazon and TripAdvisor. It shows that reducing $E[T_r]$, we can significantly improve the long term profit gains.

7.7 Implications

Our findings can benefit both the practice and theory of reputation systems. For eBay like e-commerce systems they need to deploy some extra mechanisms or refine the design of their reputation system to reduce the ramp up time so that they can reduce the probability that sellers drop out and attract more sellers. For newly start up systems like Google helpouts, they need to to deploy some reputation systems having short ramp up time, because in the early stage ramp up users is critical. For practical reputation system design, they should be aware of the ramp up time. Lastly, our frameworks can serve as important building blocks to identify deficiencies of reputation systems.

8 Related Works

Reputation systems [21] is an important research topic in network economics. Research on reputation system can be categorized into three typical aspects: (1) reputation formulation and calculation, e.g., [11], [16], (2) attacks and defense techniques design e.g., [3], [10], and (3) effectiveness and efficiency of reputation systems [5]. A survey is given in [13].

Many theoretical works explored reputation metric formulation and calculation. There are two typical reputation formulating models, i.e., the rating-based model [11], [24] and the transitive trust model [3], [4]. The rating-based reputation formulating model aims to solicit explicit human feedbacks (or ratings) [2], [11], [24], [33]. It computes a reputation score for each user by summarizing his feedback ratings. The transitive trust reputation model [3], [4], [16], [22], [32] captures the propagation of trust among users. Graph model is applied to quantify users' reputation, i.e., each user is abstracted as a node, and each weighted directed link, e.g., from B to A, measures the trust that B expresses to A. Several algorithms were proposed to compute the reputation score for users [3], [4], [16], [22], [32]. The key difference between our work and theirs is that we conduct a data-oriented study to uncover the inefficiency of real-world online reputation systems, while theirs are theoretical in nature. Our work enriches theoretical studies by uncovering the importance of ramp up time.

A number of defense techniques have been developed for reputation systems. One typical potential attack is dishonest feedbacks. Peer-prediction mechanisms were proposed to address this attack [14], [15], [18]. Another potential attack is reputation inflation, and number of techniques have been proposed to address this attack [3], [10], [28], [32]. A nice survey on the state-of-the-art attack and defense techniques is [10]. Our work propose a general framework to investigate the efficiency of defense techniques. We propose an important factor, i.e., ramp up time, that various defense techniques need to be aware of.

Several works explored the effectiveness of reputation systems [1], [5], [6], [17]. In [5], authors tried to improve the efficiency of eBay reputation computation by proposing an algorithm which relies on buyer friendship to filter out unfair ratings. The work [5] explored how buyers' rating biases (i.e., leniency or criticality) may influence sellers' product advertising behavior in eBay. Authors in [17] conducted a measurement study on the impact of negative feedbacks on eBay reputation system. Our work is different form theirs in that their works only applies to eBay reputation system, while our work applies to general rating-based reputation systems. We examine ramp up time and propose randomized algorithms to carry out large scale data analytics to uncover the deficiencies of online reputation systems.

9 Conclusions

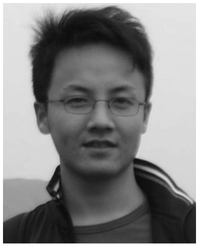
This is the first paper which presents a data driven approach to uncover the deficiencies of real-world online reputation systems. We proposed two measures to quantify the efficiency of online reputation systems: (1) ramp up time of a new service provider, (2) long term profit gains for service providers. We present a novel data analytical framework to evaluate these two measures from data. We showed that it is computationally infeasible to evaluate these two measures due to inherent preference or personal biases in expressing feedbacks (or ratings). We developed computationally efficient randomized algorithms with theoretical performance guarantees to address this computational challenge. We apply our methodology to real-life datasets from eBay, Google Helpouts, Amazon and TripAdvisor. We extensively validate our model. We discover the deficiencies of online reputation systems: (1) the ramp up time is more than 500 days; (2) reducing ramp up time can improve the long term profit gains significantly, e.g., an 80 percent reduction on ramp up time leads to at least 50 percent (as high as 100 percent) improvement in long term profit gains. Our experimental results also uncover insights on why Google Helpouts was eventually shut down and why eBay is losing sellers heavily.

ACKNOWLEDGMENTS

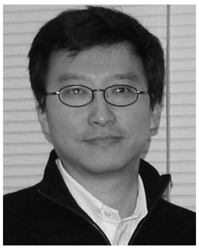
This work is supported in part by the GRF 14205114. An earlier conference version of the paper appeared in the IEEE International Conference on Data Mining (ICDM 2015) [30]. In this journal version, we add a novel measure, i.e., long term profit gains to quantify the efficiency of a reputation system. We propose a computationally efficient randomized algorithms with theoretical performance guarantees to approximate the long term profit gains. We conduct more experiments to validate our model and uncover the deficiencies of online reputation systems.

REFERENCES

[1] R. Bhattacharjee and A. Goel, "Avoiding ballot stuffing in eBay-like reputation systems," in *Proc. ACM SIGCOMM Workshop Econ. Peer-to-Peer Syst.*, 2005, pp. 133–137.
[2] S. Buchegger and J.-Y. Le Boudec, "A robust reputation system for peer-to-peer and mobile ad-hoc networks," in *Proc. 2nd Workshop Econ. Peer-to-Peer Syst.*, 2004, pp. 120–125.
[3] A. Cheng and E. Friedman, "Sybilproof reputation mechanisms," in *Proc. ACM SIGCOMM Workshop Econ. Peer-to-Peer Syst.*, 2005, pp. 128–132.
[4] R. Delaviz, N. Andrade, J. Pouwelse, and D. Epema, "SybilRes: A sybil-resilient flow-based decentralized reputation mechanism," in *Proc. IEEE 32nd Int. Conf. Distrib. Comput. Syst.*, 2012, pp. 203–213.
[5] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," in *Proc. 2nd ACM Conf. Electron. Commerce*, 2000, pp. 150–157.
[6] C. Dellarocas, "Analyzing the economic efficiency of eBay-like online reputation reporting mechanisms," in *Proc. 3rd ACM Conf. Electron. Commerce*, 2001, pp. 171–179.
[7] eBay, 1995. [Online]. Available: http://www.ebay.com/
[8] Forbes, "Google helpouts is shutting down on april 20th," 2015. [Online]. Available: http://www.forbes.com/sites/amitchowdhry/2015/02/16/google-helpouts-is-shutting-down-on-april-20th/
[9] Google Helpouts, 2013. [Online]. Available: https://helpouts.google.com/
[10] K. Hoffman, D. Zage, and C. Nita-Rotaru, "A survey of attack and defense techniques for reputation systems," *ACM Comput. Surveys*, vol. 42, no. 1, pp. 1:1–1:31, 2009.
[11] D. Houser and J. Wooders, "Reputation in auctions: Theory, and evidence from eBay," *J. Econ. Manage. Strategy*, vol. 15, no. 2, pp. 353–369, 2006.
[12] IWantCollectibles, "Why most eBay sellers fail?" 2011. [Online]. Available: http://www.news.iwantcollectibles.com/ebay-sellers-fail.shtml
[13] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Syst.*, vol. 43, no. 2, pp. 618–644, 2007.
[14] R. Jurca and B. Faltings, "Minimum payments that reward honest reputation feedback," in *Proc. 7th ACM Conf. Electron. Commerce*, 2006, pp. 190–199.
[15] R. Jurca and B. Faltings, "Collusion-resistant, incentive-compatible feedback payments," in *Proc. 8th ACM Conf. Electron. Commerce*, 2007, pp. 200–209.
[16] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The Eigentrust algorithm for reputation management in P2P networks," in *Proc. 12th Int. Conf. World Wide Web*, 2003, pp. 640–651.
[17] T. Khopkar, X. Li, and P. Resnick, "Self-selection, slipping, salvaging, slacking, and stoning: The impacts of negative feedback at eBay," in *Proc. 6th ACM Conf. Electron. Commerce*, 2005, pp. 223–231.
[18] N. Miller, P. Resnick, and R. Zeckhauser, "Eliciting informative feedback: The peer-prediction method," *Manage. Sci.*, vol. 51, no. 9, pp. 1359–1373, 2005.
[19] M. Mitzenmacher and E. Upfal, *Probability and Computing*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[20] oDesk, 2009. [Online]. Available: https://www.odesk.com/o/home
[21] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Commun. ACM*, vol. 43, no. 12, pp. 45–48, 2000.
[22] P. Resnick and R. Sami, "Sybilproof transitive trust protocols," in *Proc. 10th ACM Conf. Electron. Commerce*, 2009, pp. 345–354.
[23] Retailwire, "Ebay's loss is amazon marketplace gain," 2015. [Online]. Available: http://www.retailwire.com/discussion/18203/ebays-loss-is-amazon-marketplaces-gain
[24] A. Singh and L. Liu, "TrustMe: Anonymous management of trust relationships in decentralized P2P systems," in *Proc. 3rd Int. Conf. Peer-to-Peer Comput.*, 2003, pp. 142–149.
[25] Taobao, 2003. [Online]. Available: http://www.taobao.com/
[26] Topix, "Ebay exodus: Why are thousands of sellers leaving eBay?" 2013. [Online]. Available: http://www.topix.com/forum/indy/TFE181M8MUGHKR73C
[27] TripAdvisor, 2000. [Online]. Available: http://www.tripadvisor.com/
[28] B. Viswanath, M. Mondal, K. P. Gummadi, A. Mislove, and A. Post, "Canal: Scaling social network-based Sybil tolerance schemes," in *Proc. 7th ACM Eur. Conf. Comput. Syst.*, 2012, pp. 309–322.
[29] WebProNews, "Top 10 frustrations for eBay sellers," 2009. [Online]. Available: http://www.webpronews.com/top-10-frustrations-for-ebay-sellers-2009–01
[30] H. Xie and J. C. S. Lui, "Data driven approach to uncover deficiencies in online reputation systems," in *Proc. IEEE Int. Conf. Data Mining*, 2015, pp. 1045–1050.
[31] H. Xie and J. C. S. Lui, "Mathematical modeling and analysis of product rating with partial information," *ACM Trans. Knowl. Discovery Data*, vol. 9, no. 4, pp. 26:1–26:33, 2015.
[32] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, "SybilGuard: Defending against Sybil attacks via social networks," in *Proc. Conf. Appl. Technol. Archit. Protocols Comput. Commun.*, 2006, pp. 267–278.
[33] R. Zhou and K. Hwang, "PowerTrust: A robust and scalable reputation system for trusted peer-to-peer computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 4, pp. 460–473, Apr. 2007.



Hong Xie received the BEng degree from the School of Computer Science and Technology, The University of Science and Technology of China, in 2010 and the PhD degree from the Department of Computer Science & Engineering, The Chinese University of Hong Kong, in 2015, proudly under the supervision of Professor John C. S. Lui. He is currently a postdoctoral research fellow in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. His research interests include data science, approximation algorithms, network economics, etc. He is a member of the IEEE.



John C. S. Lui received the PhD degree in computer science from UCLA. He is currently the Choh-Ming Li chair professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. His current research interests include communication networks, system security (e.g., cloud security, mobile security, etc.), network economics, network sciences, large-scale distributed systems, and performance evaluation theory. He serves in the editorial board of the *IEEE/ACM Transactions on Networking*, the *IEEE Transactions on Computers*, the *IEEE Transactions on Parallel and Distributed Systems*, the *Journal of Performance Evaluation*, and the *International Journal of Network Security*. He was the chairman of the CSE Department from 2005-2011. He received various departmental teaching awards and the CUHK Vice-Chancellors Exemplary Teaching Award. He is also a co-recipient of the IFIP WG 7.3 Performance 2005 and IEEE/IFIP NOMS 2006 Best Student Paper Awards. He is an elected member of the IFIP WG 7.3, fellow of the ACM, the IEEE, and Croucher senior research fellow. His personal interests include films and general reading.