Cheating in Ranking Systems

Lihi Dery1 · Dror Hermel2 · Artyom Jelnov1

Published online: 23 March 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract
Consider a software application that pays a commission fee to be sold on an on-line platform (e.g., Google Play). The sales depend on the applications’ customer rankings. Therefore, developers have an incentive to dishonestly promote their application’s ranking, e.g., by faking positive customer reviews. The platform detects dishonest behavior (cheating) with some probability, and then decides whether to ban the application. We provide an analysis and find the equilibrium behaviors of both the application (cheat or not) and the platform (setting the commission fee). We provide insights into how the platform’s detection accuracy affects the incentives of the application’s developers.

Keywords Manipulation · Ranking fraud · Ranking systems · Ratings

1 Introduction

Various systems allow users to rate items. Using these ratings, the systems are then able to present a ranked list of items. Strategic agents may attempt to manipulate these ranked recommendations in order to increase their personal utility. However, these manipulations are costly. Furthermore, such manipulation attempts may be identified by inspection, which is also costly.

Consider, for example, an application—an “app”—and an on-line platform (e.g., the Apple App store). The app may buy fake ratings, which translate into a higher ranking on the App Store; this ranking is a measure that many users look for when they search for a new app to download. The negative impact that is generated in

Artyom Jelnov
artyomj@ariel.ac.il

Lihi Dery
lihid@ariel.ac.il

Dror Hermel
dror.hermel@rotman.utoronto.ca

1 Ariel University, 40700 Ariel, Israel
2 University of Toronto, 105 St. George Street, Toronto, ON M5S 3E6, Canada
this scenario is multifold. From the end-user side, the purchase of an app that does not generate much positive utility yields a net loss. Collectively, from the developer side, those who cheat gain profits in the short run, while for the others there is a short-term monetary loss: The apps that they develop are ranked lower.

For the platform there is reputation loss (which can be associated with lost revenues), as users may be overly cautious before downloading new apps. Therefore, it is customary for the platform to use some mechanism to detect and remove cheating apps.

In this paper, we develop a model that studies the interaction between a platform and an application. The platform collects a fee from applications that want to use it. We study two cases: when the fee is exogenously set; and when the platform sets the fee. We show that from the application’s side, for an imperfect cheating detection technology, cheating will take place. We analyze how the quality of the detection algorithm and app rankings affect the incentive to cheat. Furthermore, we analyze the platform’s decision as to what commission fee to impose.

To the best of our knowledge, this is the first attempt to apply game theory into ranking systems in practice.

We begin with some background (Sect. 2). We then present our theoretical model (Sect. 3) and conclude with a discussion of our main findings and future research directions (Sect. 4).

2 Background

We begin by surveying related work (Sect. 2.1), proceed to provide some intuition as to why rankings are significant enough so that people are willing to manipulate ratings in order to receive a high ranking (Sect. 2.2), and then survey methods for manipulation detection (Sect. 2.3).

2.1 Related Works

We suggest a new focal point to address cheating in ranking systems; this is an approach that is related to the well-established inspection games literature (cf. Avenhaus et al. 2002 for a survey). The substantial difference that we implement is that while in inspection games one of the players decides whether to commit some violation, and another player—the inspector—decides whether to perform a costly test to detect this violation, in our model a noisy alert of the violation is sent automatically.

The notion of an automatically sent signal that is based on the action of one of the players appears in the literature in different contexts: e.g., industrial espionage (Barrachina et al. 2014); international conflicts (Jelnov et al. 2017); and sports (Berentsen 2002; Kirstein 2014). In our case, we identify the favorable and adverse effects that result when the platform attempts to deter cheaters using an imperfect detection mechanism.

Our paper is related to the economics of law enforcement literature, which goes back to Becker (1968) and is surveyed in Polinsky and Shavell (2007). In our setting,
we have an enforcer (the platform) and a potential violator (the application). We study a specific kind of violation: cheating in reviewer ratings. In our setting, this violation depends on the initial rating. Moreover, the enforcer and the potential violator may have a common interest, because the application pays a commission fee to the platform.

The work of Darby and Karni (1973) resembles our topic in the sense that in their paper a violation is wrong information that is given by a service supplier to a consumer. They study the existence of this kind of violation in a free market, and discuss how government intervention can reduce it. Darby and Karni (1973) do not model strategic behavior by the government. In our paper, a platform, not a government, enforces honest behavior by an application, and we incorporate strategic considerations for the platform.

The literature on the economics of tort law, which can be traced to Landes and Posner (1984), relates to our paper as the application may cause damage to the platform. However, the tort law literature discusses how to cause one party to take care and prevent accidents that damage another party. Compensation for damage is the most common tool in tort law. In our case, no compensation is paid to the platform.

### 2.2 Why Rankings Matter

It has been shown that a website’s rank—not just its relevance—strongly and significantly affects the likelihood of a click (Glick et al. 2014).

As of March 2017, Google had 2.8 million apps that were available through its Android platform, and Apple had 2.2 million apps that were available on its iTunes App Store.¹ With such massive numbers, users who are interested in discovering apps rely on rated listings, which are known as “top charts” such as “top free games”, “top free apps”, etc. Furthermore, a study by Carare (2012) indicates that users are willing to pay $4.50 or more for an app that is top ranked as compared to the same app that is unranked, as people in general tend to disproportionately select products that are ranked at the top (Smith and Brynjolfsson 2001; Cabral and Natividad 2016). With respect to the monetary effects of higher ranking of apps, Lee and Raghu (2014) claim that one of the keys to a successful app is top rank status.

Reviews play a critical role in online commerce (Mauri and Minazzi 2013). For example, hotel reviews on websites that customers perceive as credible influence purchase behaviors (Casalo et al. 2015). Mayzlin et al. (2014) show that competing products can self-promote by faking positive reviews for themselves, or by posting negative reviews for competitors. For analysis of the impact of reputation in e-commerce, see also Resnick and Zeckhauser (2002), who show that positive reviews of previous online transactions can predict good transactions in the future. Thus, the monetary gain from a highly ranked app is an incentive for app developers to boost their app rankings on the charts. In competing over reputation and higher ranking,

¹ Statista: The Statistics Portal [https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/](https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/)
product managers might be tempted to engage in manipulative behavior (Gössling et al. 2016).

In the app development context, some app developers may choose to do so in a deceptive manner by paying fraudulent ranking services (and thus engaging in moral hazard behavior). If not addressed, these deceptions are harmful to the app platform eco-system: other developers; users; and the platform’s owners. For the other developers, fraudulent ranking leads to unfair competition, which might discourage honest developers (which is a manifestation of adverse selection). The users might be led erroneously to install misbehaving or malicious apps, or they might be dissatisfied with platform app ratings and stop trusting the app charts (which is another manifestation of adverse selection, as well as moral hazard). Finally, the platform’s reputation might be compromised. As noted above, this can be seen as a case of both moral hazard and adverse selection.

2.3 Manipulation Detection

Detection of deceptions is a top-priority for platforms, and is performed by detecting suspicious app patterns, user patterns, or both. A closely related line of work focuses on malware detection (Burguera et al. 2011; Narudin et al. 2016; Seneviratne et al. 2017). However, our focus is on ranking fraud, where the manipulating app is not necessarily malware.

Manipulating apps exhibit a different review pattern when compared with honest apps. Some algorithms for fake review detection focus on textual analysis of fake reviews; some algorithms find certain language constructs that are often used in fake reviews (e.g., Ott et al. 2011; Hu et al. 2012; Banerjee et al. 2017), while others (Schuckert et al. 2016) highlight the contradictions between the overall rating (e.g., for a hotel) and the detailed rating of the same product (e.g., specifics such as cleaning or location), which can expose fraudulent ratings.

Manipulated app rankings are likely to generate drastic ranking increases or decreases in a short time, or show strong rating deviations (see Zhu et al. 2013). Heydari et al. (2016) show that the time interval within which the rating is given is also a measure of review trustworthiness. The detection can be based solely on ratings, by analyzing rating shifts, under the assumption that most ratings are honest (Akoglu et al. 2013; Savage et al. 2015). In addition, manipulative users have a different review pattern. As setting up a valid account is time consuming (e.g., to rate an app on Google Play requires a Google account, a mobile phone registered with the account, and the installed app), manipulators reuse their account and rate many apps over a short time frame. Manipulators may rate up to 500 apps a day, and rate them all with five stars.

Manipulative users are usually part of a well-organized crowdsourcing system that performs malicious tasks. These are nicknamed “crowdturfing” systems, and their unique features have been mapped (see Wang et al. 2012). A recent study by Chen et al. (2017) focused on identifying app clusters that are co-promoted by collusive attackers. The identification is based on unusual changes in rating patterns; the identification measures feature similarity in apps and applies machine-learning
techniques. Ye and Akoglu (2015) show that network information can also be employed. The baseline assumption is that an honest set of reviews for a product (or app) is formed by independent reviewer actions with various levels of activities and reviews. Therefore, a non-manipulative app should have reviews with various levels of network centrality. Furthermore, correlated review activities can be combined with linguistic and behavioral review signals (Rahman et al. 2017).

A supermarket that offers a product or a newspaper that advertises a product may face a problem when that product promises to deliver a high-quality outcome but instead fails the customer. However, we see several distinctions; for example:

1. A platform displays a list of ranked applications, whereas the supermarket or newspaper usually does not rank the products. In a supermarket, the customer has the opportunity to choose among a few parallel products on display on the shelves. In a newspaper there may be a few similar advertised products on display. In both cases (supermarket and newspaper), the selection is limited to a few unranked products. In the case of apps, there can be millions of ranked products.
2. A supermarket (or even a newspaper) can usually inspect the product to confirm whether the product is of high quality or not. For online platforms confirmation is difficult, and typically there is reliance on an imperfect algorithm that might err.

In the next section, we present a formal model for the interaction of two agents, an application and a platform. Our model assumes that one (or all) of the above manipulation detection capabilities are available to the platform.

3 Model

We first consider two models: one with an exogenous fee; and one with an endogenous fee: a fee that is set by the platform. In both these models, we assume that the final rating is a continuous random variable. Last, we consider a model where the app obtains (either truthfully or by cheating) the topmost rating. A list of the notations that are used throughout the models is provided in Table 1.

3.1 Preliminaries

We consider two risk-neutral agents: an application (A); and a platform (P). We focus only on cases where the application decides to enter the platform.

A rating for each application is calculated periodically. The rating represents the opinion of the users and is observed both by the application and the platform. Upon entrance, at stage $t_0$ each application receives an initial rating of $r_0 = 0$. At stage $t_1$ the application obtains a rating of $r \in [0, 1]$.

For the application, a higher rating results in higher visibility on the platform, which translates into higher profits. We assume that the profit is positively related to the application’s current rating $r$, minus a commission fee $f$ that is a proportion of
the revenue \((0 \leq f \leq 1)\) that is paid to the platform. Thus the application is left with a revenue of \(\gamma r(1 - f)\), where \(\gamma > 0\) is the rating-related profit coefficient.

In order to increase the expected rating \(r\), the application may decide to cheat \((c)\): by adding fake ratings. Cheating is a costly action with the cost \(e \geq 0\). If the application does not cheat \((\tilde{c})\), then \(r\) is picked from a uniform distribution over the interval \([0, 1]\). If it cheats, \(r\) is distributed uniformly on \([d(e), 1]\), \(0 < d(e) < 1\). The expected value of \(r\) and, consequently, the application’s expected profit are higher if the app cheats. For simplicity of analysis we assume \(e = 0\) and \(d(e) \equiv d\).

The platform has some an imperfect algorithm that enables the platform to detect applications that might be cheating (see Sect. 2.3 for more details on such algorithms). Indeed, no algorithm or technology is 100% error free, and the used algorithm might overlook some cheating applications as well as incorrectly identify honest applications as cheaters. We assume that once the platform implements a specific cheating detection algorithm, there is no cost to running the algorithms and scanning all of the applications.

| Table 1 | List of notations in the model |
|---------|--------------------------------|
| Notation | Description |
| \(A\)  | Application |
| \(P\)  | Platform |
| \(r\)  | \(A\)’s rating |
| \(f\)  | Proportional fee that is paid by the application to the platform |
| \(\gamma\) | Coefficient that relates \(A\)’s rating to \(A\)’s profit |
| \(c\)  | Action of cheating |
| \(\tilde{c}\) | Action of not cheating |
| \(e\)  | Cost of cheating |
| \(d\)  | Lower limit of the rating, following cheating |
| \(s\)  | Suspecting alert |
| \(\tilde{s}\) | Non-suspecting alert |
| \(b\)  | Banning action |
| \(\tilde{b}\) | Non-banning action |
| \(a\)  | Probability of a type-I error in detection |
| \(\beta\) | Probability of type-II error in detection |
| \(v\)  | Revenue from not making false accusations |
| \(w\)  | Cost of non-detection |
| \(P_c\) | Probability of cheating |
| \(\tilde{r}\) | Threshold of suspicious rating, determined by the platform |
| \(\rho\) | Initial rating in the topmost model |
| \(a(\rho)\) | Probability of a type-I error in detection in the topmost model |
| \(\beta(\rho)\) | Probability of type-II error in detection in the topmost model |
| \(l(\rho)\) | The probability to obtain honestly the highest rating (in the topmost model) |
| \(P_b\) | Probability that \(P\) bans suspected \(A\) (in the topmost model) |
At stage $t_1$, an automatic noisy alert is sent to the platform. The alert $s$ means that the application is suspected of cheating ($\hat{s}$ means the opposite alert). When the platform receives $s$, it chooses whether to ban ($b$) or to ignore the alert and thus not ban ($\hat{b}$) the application. The decision to ban the app sets the application’s rating to $r = 0$.

Let $\alpha$ be a type-I probability error: the probability that $s$ is sent when $A$ does not cheat; and let $\beta$ be the probability of a type-II error: the probability that $s$ is not sent when $A$ cheats.

We consider $\alpha$ and $\beta$ as commonly known. When a platform chooses a cheating detection algorithm, as part of the acceptance testing that is performed when integrating the algorithm, it is tested on different scenarios (where the results are known), and $\alpha$ and $\beta$ can then be estimated. In a similar manner, application developers can estimate these parameters.

The platform’s utility consists of three factors: the revenue from the commission fee that the application pays ($\gamma rf$); the cost of non-detection ($w$); and the revenue from not making false accusations ($v$). The two latter parameters can be interpreted as the impact on the platform’s reputation, which translates into loss (or gain) of user confidence in the platform,\(^2\) which leads to a decrease (or increase) in purchases and thus in the platform’s revenues.

Consequently, if the application cheats and is not banned, $P$’s utility is $\gamma rf - w$, $w > 0$. However, the revenue from not making false accusations $v$ is positive. This is due to the fact that an application that does not cheat and is not banned has a positive effect on the platform’s reputation, which increases the platform’s profit in the long run. Thus, if there is no false accusation, $P$ obtains $\gamma rf + v$, $v > 0$. If the application is banned, the platforms revenue is 0.

The total revenue of the application is $\gamma r(1 - f)$ if it does not cheat and $\gamma r(1 - f) - e$ if it cheats. Recall that we assume that $e = 0$ and that the revenue in the case that an app cheats and is caught is $r = 0$.

The game and player utilities are defined in Fig. 1.

### 3.2 Exogenous Fee

We now proceed and describe our results for an exogenous fee. The platform receives a signal $s$. If $\alpha = 0$—if there is no possibility for a mistake with regard to the signal—the platform bans the application. We assume hereafter $\alpha > 0$. We consider only an equilibrium of the threshold form: The threshold $\tilde{r} - d \leq r \leq 1$—is predetermined by the platform. Following the alert $s$, the platform bans the application if and only if $r$ is above the threshold $\tilde{r}$ ($\tilde{r} < r$). The intuition is that the platform is interested in considering only top-rated apps, and is not concerned with minor marginal improvements in the apps rating. For example, for $\tilde{r} = 0.3$ means that the

---

\(^2\) An application is an experience good. The users assume that high-rated apps are ones with which other users have had a positive experience. If an app receives a high rank by cheating, the user might be disappointed from his experience with the app.
platform is not concerned with apps that have a rating of $r = 0.29$. Note that $\bar{r} = 1$ means that $P$ never bans $A$.

**Proposition 1** Let $\alpha > 0$.

1. If $w < \gamma fd$, then the application cheats with certainty; and with certainty the platform does not ban the application.

2. If $w > \gamma f$ and $\beta > \frac{d+(1-a)(1+d)}{1+d}$, then the application cheats with certainty; and following an alert $s$, the platform bans with certainty.

3. If $w$ is sufficiently high and $\beta < \frac{(1-a)(1-d^2)+d^2}{1+d}$, then there is an equilibrium where the application cheats with a probability $P_c$, $0 < P_c < 1$; and following an alert $s$ the platform bans iff $r > \bar{r}$, $d < \bar{r} < 1$.

All proofs appear in the “Appendix”. Proposition 1 can be explained as follows:

1. If the platform’s revenue from the commission fee is higher than the cost of non-detection—the reputation loss—then the platform will not ban the application even if the platform suspects cheating.
2. If $\beta$ is high, then the application is encouraged to cheat, since it is likely that cheating will not be detected.
3. If none of the former occurs, then the application will cheat with some positive probability, but not with certainty.

We now focus on part (3), where cheating occurs with some probability $0 < P_c < 1$. In this case, an increase in $v$ and $f$ and a decrease in $w$ results in more cheating: The first-order effect of an increase in $f$ (the commission fee) is a decrease in profits for the application. In this case the application may attempt to cheat in order to gain a higher rank and thus compensate for the loss in profit that is due to $f$. An increase in the revenue from not making false accusations ($v$) means that the platform is more reluctant to ban an application, which also leads to more cheating. An increase in the non-detection cost ($w$) has an opposite effect: less cheating.

As a direct result of part 3 in Proposition 1:

**Corollary 1** Let $w$ be sufficiently high and $\beta < \frac{(1-\alpha)(1-d^2)+d^2}{1+d}$. Then in the equilibrium:

1. The probability that the application cheats increases in: $f$ and $v$, and decreases in $w$.
2. The threshold $\bar{r}$ decreases in $u$ and $\bar{u}$.

Corollary 1 can be explained as follows: An increase in the probability of false accusation $\alpha$ means that even an honest application is more likely to be (mistakenly) banned for cheating; thus the application has less incentive to act honestly. Consequently, the region of ratings where the platform will ban applications increases in $\alpha$. An increase in the probability of non-detection of cheating $\beta$ may encourage cheating, therefore, the region of ratings where the platform will ban applications increases in $\beta$.

### 3.2.1 Efficiency of Cheating

The parameter $d$ may be interpreted as the efficiency of cheating. If the platform does not ban the application, the application’s expected payoff is higher as $d$ increases. However, for higher $d$, the platform is more suspicious that the application is cheating, and hence may ban it with higher probability. Thus, the effect of increase in the efficiency of cheating on application’s utility is ambiguous. Figure 2 illustrates this.

As shown in Fig. 2, it may be the case that the utility of the application decreases in the efficiency of cheating. This effect may be even stronger if we assume a positive cost of cheating ($e > 0$).
Fig. 2 Result for $\gamma = 1$, $\alpha = 0.1$, $\beta = 0.1$, $v = 4$, $w = 3$ and $f = 0.5$.

Fig. 3 Result for $d = 0.5$, $\gamma = 1$, $\alpha = 0.1$, $\beta = 0.1$, $v = 4$, $w = 3$. 
In Sect. 3.2 the fee $f$ was exogenous. However, in reality the fee is set by the platform. Consider a model where at an initial stage $t_0$ the platform chooses a fee $0 \leq f \leq 1$, and then the game proceeds as defined in Fig. 1.\(^3\)

An increase in the commission fee affects the platform’s utility in two ways: (1) positive—an increase in revenue; and (2) negative—an increase in cheating, which reduces the platform’s utility (which is a direct consequence of Corollary 1).

As Fig. 3 illustrates, the platform maximizes its expected utility at a fee that is lower than 1.

However, this result is not general. As can be seen in Fig. 4, when the cost of non-detection is set to $w = 4$ instead of $w = 3$, the platform maximizes its expected utility for a fee of $f = 1$.

Note that if for some reason there is no development cost for the application, then the application is indifferent to entering the platform even when $f = 1$. This could happen for example if an engineering or computer science student develops

\(^3\) The platform maximizes its expected utility. The technicalities are characterized in Proposition 3 in the “Appendix.”
an app as part of his/her portfolio, or if someone develops an app as part of his/her hobby. In these cases, $f = 1$ is a maximizing outcome for the platform.

### 3.4 Topmost Rating

In the previous model the initial rating of the application is 0, while the final rating is a continuous random variable. We consider next a model where the application obtains a rating denoted by $\rho$, $0 \leq \rho \leq 1$, and then can be promoted, in an honest way or by cheating, to the top rating, which is normalized to be 1. Different from the previous model, the accuracy of detection of cheating is not constant, but depends on $\rho$. Below is a description of the model and results:

Upon entrance, at stage $t_0$ each application receives an initial rating of $r_0 = 0$. At stages $t_1,t_2$, the application obtains a rating of $r_1,r_2 \in [0,1]$, respectively. For simplicity, we denote $r_1$ as $\rho$. The rating $r_1$ is obtained by standard application promotion activities. We hereby study the last stage ($t_2$).

As in the previous model, we assume that the profit of the application is $\gamma r_2(1 - f)$, where $\gamma > 0$ and $f$ is the commission fee ($f \geq 0$). The profit of the platform is $\gamma r_2 f$.

In order to increase the rating $r_2$, the application may decide to cheat ($\hat{c}$). If the application does not cheat ($\hat{\hat{c}}$) it still has a probability $l(\rho)$ of obtaining the highest rating $r_2 = 1$. However, with probability $1 - l(\rho)$ the rating is $r_2 = r$. The probability $l(\rho)$ ($0 < l(\rho) < 1$) increases in $\rho$. The higher is the rating $\rho$, the more probable is the outcome that the application will reach $r_2 = 1$.

At stage $t_2$, a rating of $r_2 = 1$ triggers an automatic noisy alert that is sent to the platform. As before, the alert $s$ means that the application is suspected of cheating; and $\hat{s}$ means that it is not suspected. When the platform receives $s$, it chooses whether to ban ($b$) or not ban ($\hat{b}$) the application. The ban decision is equivalent to setting the application’s rating to $r_2 = 0$. Note that this implies a different penalty cost for different applications; an application with a higher rating at stage $t_1$ has more to lose from a ban than does an application with a lower rating.

Let $\alpha(\rho)$ be a commonly known type-I probability error, and let $\beta(\rho)$ be the commonly known probability of a type-II error.

We assume that $\beta(\rho)$ weakly increases in $\rho$: A high rating $\rho$ that increases to $r_2 = 1$, is less detectable than a low rating that increases to $r_2 = 1$.

Consequently, if the application cheats and is not banned, $P$’s utility is $\gamma r_2 f - w$, $w > 0$. However, if the application does not cheat and is not banned, $P$ obtains $\gamma r_2 f + v$, $v > 0$. If the application is banned, the platforms revenue is 0. Parameters $v$ and $w$ are defined as in the previous model.

If $\rho = 1$, the highest rating is guaranteed to the application. Trivially, in this case there is an equilibrium where the application does not cheat, and the platform does not ban it. We assume hereafter $\rho < 1$.

The next result is similar to Proposition 1, and we refer the reader to the intuition that follows Proposition 1.
Proposition 2 Let $\alpha(\rho) > 0$.

1. If $w < \gamma f$, then in the unique equilibrium the application cheats with certainty; and with certainty, the platform does not ban the application.

2. If $l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) < \beta(\rho)$, then in the unique equilibrium the application cheats with certainty; and following an alert $s$, the platform bans with certainty.

3. If $w > \gamma f$ and $l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) > \beta(\rho)$, then in the unique equilibrium the application cheats with certainty; and following an alert $s$, the platform bans with certainty.

Note, that for $w = \gamma f$ or $l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) = \beta(\rho)$ the equilibrium may not be unique.

We now focus on part (3), where cheating occurs with some probability $0 < P_c < 1$. In this case, an increase in $\alpha(\rho)$, $\beta(\rho)$, $v$, and $f$ and a decrease in $w$ results in more cheating. These results are similar to the results of Corollary 1; however, the dependence of the probability of cheating on $\alpha(\rho)$ and $\beta(\rho)$ is new. Formally:

Corollary 2 Let $w > \gamma f$ and $l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) > \beta(\rho)$. Then in the equilibrium of $G$:

1. The probability that the application cheats increases in: $f$, $v$, $\alpha(\rho)$, and $\beta(\rho)$, and decreases in $w$.
2. The probability that $P$ bans $A$, following alert $s$, increases in $\beta(\rho)$ and in $\alpha(\rho)$.

These new results are intuitive. If the detection algorithm is less reliable (high probability of error), the application has more incentive to cheat. Therefore, the platform is more aggressive, and bans more frequently.

Corollary 3 Suppose $\alpha(\rho)$ is constant, $\alpha(\rho) \equiv \alpha$. Let $w > f a$ and $l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) > \beta(\rho)$. Then in the equilibrium, the probability that $A$ cheats increases in $\rho$.

Corollary 3 states that when $\alpha(\rho)$ is independent of $\rho$, and cheating occurs with some probability $0 < P_c < 1$, then as the application approaches a rating of 1, it is more likely to cheat. The intuition behind this is that since applications with a rating close to 1 are less suspected of cheating, and hence less likely to be detected, they can thus cheat more freely.

The statement in Corollary 3—that cheating increases in $\rho$—is not surprising for a strictly increasing $\beta(\rho)$ since this means that fewer alerts $s$ are sent when $\rho$ is close to 1. However, what is somewhat surprising is the fact that this claim holds even for those cases where $\beta(\rho)$ is constant: Even when the probability of the alert is constant, cheating increases in $\rho$. 
The analysis in this section is for an exogenous $f$. Note that similarly to Sect. 3.3, it can be shown numerically that if the platform chooses $f$, it may be better off by choosing $f < 1$.

4 Conclusions

In this paper, we provide a novel stylized framework to study the interaction between an app sales platform (e.g., Apple’s app store or Google Play) and an app developer who may be tempted to cheat in order to increase its app ranking. Our framework captures some of this interaction, and the consequential equilibrium analysis gives rise to some important implications. and an app developer that may be tempted to cheat in order to increase its app ranking. Our framework captures some of this interaction, and the consequential equilibrium analysis gives rise to some important implications.

Our most significant finding is that a higher fee leads to more cheating. Consequently, even a monopolistic platform may choose not to impose a high fee.

Furthermore, we found that precise alert signals decrease cheating; when the cheating detection algorithm is a good one—when $\alpha$ and $\beta$ are low—less cheating occurs. Thus, we conclude that if the platform has a good manipulation detection algorithm then it should make this known (i.e. publicize its $\alpha$ and $\beta$), since the application developers will refrain from cheating if they know that there is a high chance that they will be caught.

We focused on the commission fee that the product has to pay the platform and considered other costs—such as promotion costs and the cost of creating fake reviews—as negligible. We assumed that the cost of a fake review is sufficiently low, and the reward from an undetected cheat is high, so there is an incentive to cheat. It can be shown that if the cost of cheating is sufficiently high, the application will not cheat. Note that platforms have various tools to protect against manipulations and to make it more difficult to create a fake review. For example, they can require reviewers to use the verification system CAPTCHA, or to verify that a reviewer really consumes the product. Still, sophisticated manipulators can bypass these barriers.

Utilities in our model are exogenous. We assume that the platform is interested in its reputation, and that cheating harms the platform’s reputation. In a future extended model, platform competition can be considered, where more than one platform competes for customers and customers abandon a platform if they are dissatisfied with its cheating prevention level (for platform competition in a non-cheating environment see, for example, Hałaburda and Yehezkel 2016).

Most importantly, we provide initial insights into how the platform’s detection accuracy affects the incentives of the app developers. Understanding these interactions, and the resulting equilibria, provide an ample foundation to address future points of interest. For example, it can be used to understand better how to put in place mechanisms that align incentives and to provide a benchmark framework for future empirical work.

Our findings and conclusion are relevant to other types of e-commerce as well, and can be of interest in any scenario that involves a product that faces a fee to
be rated and ranked on an online platform. Other examples of such systems may include online vendor sites such as Amazon, eBay, and hotel bookings sites.

**Acknowledgements** We wish to thank Christopher Thomas Ryan, Yair Tauman, Richard Zeckhauser, the journal editor—Lawrence J. White—and the anonymous reviewers for their helpful suggestions.

**Appendix**

**Proof of Proposition 1** Consider first an equilibrium where $P$ chooses $\bar{r} = 1$. Then $c$ is a superior action of $A$. If $P$ chooses $\hat{b}$, its payoff is therefore $\gamma f \bar{r} - w$. If $P$ chooses $b$, its payoff is $0$; and the platform prefers $\hat{b}$ to $b$ for each $d \leq r \leq 1$ if $w < \gamma f d$.

Next, consider $\bar{r} = d$. Observe, that for $\alpha > 0$, pure $\hat{c}$ is not an equilibrium in this case. By contrary, if $A$ chooses pure $\hat{c}$, with positive probability it obtains a rating above $d$, and a false signal $s$ is sent to $P$; therefore, $b$ is not the best reply of the platform.

If $A$ chooses $c$ and $P$ (following $s$) chooses pure $b$, $A$’s expected utility is \[ \frac{\gamma(1-f)\beta(1+d)}{1-d} \alpha \bar{r} \] . If $A$ does not cheat and if $P$ (following $s$ and $r > d$) chooses pure $b$, $A$’s expected utility is \[ \frac{\gamma(1-f)}{2}(d + (1-\alpha)(1+d)). \] Thus, $A$ prefers $c$ to $\hat{c}$ for $\beta > \frac{d+(1-\alpha)(1+d)}{2}$.

Consider next $d < \bar{r} < 1$. $A$ is indifferent between $c$ and $\hat{c}$ for

\[ \frac{\gamma(1-f)}{1-d} \int_{d}^{\bar{r}} r \, dr + \beta \int_{\bar{r}}^{1} r \, dr = \gamma(1-f)\int_{0}^{\bar{r}} r \, dr + (1-\alpha)\int_{\bar{r}}^{1} r \, dr, \]

which simplifies to

\[ \bar{r} = \sqrt{\frac{(1-\alpha)(1-d) - \beta + d^2}{d - \beta + (1-\alpha)(1-d)}}. \] (1)

By (1), for $\beta < \frac{(1-\alpha)(1-d^2)+d^2}{1+d}$, $d < \bar{r} < 1$.

Let $A$ choose $c$ with probability $P_c$. Given alert $s$ and rating $r > d$, let $P(\text{cls})$ be belief of $P$ that $A$ cheats:

\[ P(c|s) = \frac{P_c(1-\beta)^{r-d}}{P_c(1-\beta)^{r-d} + (1-P_c)\alpha r}. \] (2)

Following alert $s$, $P$ is indifferent between $b$ and $\hat{b}$ for the threshold rating $r$ of $A$ iff

\[ \gamma f \bar{r} - w P(c|s) + v(1 - P(c|s)) = 0, \] (3)

and by (2), this is equivalent to

\[ P_c = \frac{\alpha \bar{r}(\gamma f \bar{r} + v)(1-d)}{\alpha \bar{r}(\gamma f \bar{r} + v)(1-d) + (1-\beta)(w - \gamma f \bar{r})(\bar{r} - d)}. \] (4)
By (4), \(0 < P_c < 1\) for \(w > \gamma f \tilde{r}\).

By substitution of (2) into (3) one can verify that for sufficient high \(w\) the left hand side of (3) decreases in \(\tilde{r}\). For \(r < \tilde{r}\) the platform does not ban; and for \(r > \tilde{r}\), it bans, as is required by a threshold strategy.

**Proof of Corollary 1** Since conditions of part 3 of proposition 1 hold, probabilities of cheating and the threshold of banning are given by (4) and (1), respectively. Results follow directly by (4) and (1).

---

**Proposition 3**

1. If \(\frac{2w}{\gamma(1+d)} < f\), then in equilibrium the expected utility of \(P\) is \(\frac{\gamma f (1+d)}{2} - w\).

2. If \(\frac{2w}{\gamma(1+d)} > f\) and \(\beta > \frac{d + (1-\alpha)(1+d)}{1+d}\), then in equilibrium the expected utility of \(P\) is \(\beta \frac{\gamma f (1+d)}{2} - w\).

3. If \(w\) are sufficiently high and \(\beta < \frac{\gamma f (1+d)}{d(1-d^2) + d^2}{1+d}\), then in equilibrium the expected utility of \(P\) is

\[
(1 - P_c) \left[ \frac{\gamma f}{2} \tilde{r}^2 + (1 - \alpha)(1 - \tilde{r}^2) \right] + v[\tilde{r} + (1 - \alpha)(1 - \tilde{r})]
\]

\[
+ \frac{P_c}{1 - d} \left[ \frac{\gamma f}{2} \tilde{r}^2 - d^2 + \beta(1 - \tilde{r}^2) \right] - w[\tilde{r} - d + \beta(1 - \tilde{r})],
\]

where \(P_c\) and \(\tilde{r}\) are given by (4) and (1).

**Proof** Directly from Proposition 1 and Fig. 1.

**Proof of Proposition 2** Consider first an equilibrium where \(P\) chooses pure \(\hat{b}\). Then \(c\) is a superior action of \(A\). The expected utility of \(P\) in this case is \(\gamma f - w\). If \(P\) chooses \(b\), its payoff is 0, and the platform prefers \(\hat{b}\) to \(b\) for \(w < \gamma f\).

Next, consider that \(P\), following \(s\), chooses pure \(b\). Observe, that for \(\alpha(\rho) > 0\), pure \(\hat{c}\) is not an equilibrium in this case. By contrary, if \(A\) chooses pure \(\hat{c}\), with positive probability it obtains the rating 1 and a false signal \(s\) is sent to \(P\); therefore, \(b\) is not the best reply of the platform.

If \(A\) chooses \(c\) and if \(P\) (following \(s\)) chooses pure \(b\), \(A\)'s expected utility is \(\gamma(1 - f)\beta(\rho)\). If \(A\) does not cheat and if \(P\) (following \(s\)) chooses pure \(b\), \(A\)'s expected utility is \(\gamma(1 - f)(\rho(1 - l(\rho)) + l(\rho)(1 - \alpha(\rho)))\). Thus, \(A\) prefers \(c\) to \(\hat{c}\) for \(\beta(\rho) > r - rl(\rho) + l(\rho) - \alpha(\rho)l(\rho)\).

Let \(A\) choose \(c\) with probability \(P_c\). Similar to the proof of Proposition 1,

\[
P_c = \frac{\alpha(\rho)l(\rho)(\gamma f + v)}{\alpha(\rho)l(\rho)(\gamma f + v) + (1 - \beta(\rho))(w - \gamma f)}.
\]

By (5), \(0 < P_c < 1\) for \(w > \gamma f\).

Let \(P_b\) be a probability with which \(P\) bans the application, following alert \(s\). \(A\) is indifferent between \(c\) and \(\hat{c}\) for
\[
\gamma (1-f)[1-(1-\beta(\rho))P_b] = \gamma (1-f)[(1-l(\rho))\rho + l(\rho)(1-\alpha(\rho)P_b)],
\]
which simplifies to
\[
P_b = \frac{(1-\rho)(1-l(\rho))}{1-\beta(\rho)-\alpha(\rho)l(\rho)}.
\tag{6}
\]
By (6), \(0 < P_b < 1\) for \(l(\rho) - \alpha(\rho)l(\rho) + \rho - \rho l(\rho) > \beta(\rho)\).

**Proof of Corollary 2** Since the conditions of part 3 of Proposition 2 hold, probabilities of cheating and of banning are given by (5) and (6), respectively. Results follow directly by (5) and (6).

**Proof of Corollary 3** Since the conditions of part 3 of Proposition 2 hold, probabilities of cheating is given by (5). The result follows directly by (5) and, by \(\frac{d\beta(\rho)}{d\rho} > 0\) and \(\frac{d\beta(\rho)}{d\rho} \geq 0\).

**References**

Akoglu, L., Chandy, R., & Faloutsos, C. (2013). Opinion fraud detection in online reviews by network effects. In *Proceedings of the seventh international AAAI conference on Weblogs and social media*.

Avenhaus, R., Von Stengel, B., & Zamir, S. (2002). Inspection games. In R. J. Aumann & S. Hart (Eds.), *Handbook of game theory with economic applications* (Vol. 3, pp. 1947—1987). North–Holland, Amsterdam.

Banerjee, S., Chua, A. Y., & Kim, J.-J. (2017). Don’t be deceived: Using linguistic analysis to learn how to discern online review authenticity. *Journal of the Association for Information Science and Technology, 68*(6), 1525–1538.

Barrachina, A., Tauman, Y., & Urbano, A. (2014). Entry and espionage with noisy signals. *Games and Economic Behavior, 83*, 127–146.

Becker, G. S. (1968). Crime and punishment: An economic approach. In *The economic dimensions of crime* (pp. 13–68). London: Palgrave Macmillan.

Berentsen, A. (2002). The economics of doping. *European Journal of Political Economy, 18*(1), 109–127.

Burguera, I., Zurutuza, U., & Nadjm-Tehrani, S. (2011). Crowdroid: behavior-based malware detection system for android. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices* (pp. 15–26). ACM.

Cabral, L., & Natividad, G. (2016). Box-office demand: The importance of being# 1. *The Journal of Industrial Economics, 64*(2), 277–294.

Carare, O. (2012). The impact of bestseller rank on demand: Evidence from the app market. *International Economic Review, 53*(3), 717–742.

Casalo, L. V., Flavian, C., Guinaliu, M., & Ekinci, Y. (2015). Do online hotel rating schemes influence booking behaviors? *International Journal of Hospitality Management, 49*, 28–36.

Chen, H., He, D., Zhu, S., & Yang, J. (2017). Toward detecting collusive ranking manipulation attackers in mobile app markets. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security* (pp. 58–70), New York, NY, USA. ACM.

Darby, M. R., & Karni, E. (1973). Free competition and the optimal amount of fraud. *The Journal of Law and Economics, 16*(1), 67–88.

Glick, M., Richards, G., Sapozhnikov, M., & Seabright, P. (2014). How does ranking affect user choice in online search? *Review of Industrial Organization, 45*(2), 99–119.
Gössling, S., Hall, C. M., & Andersson, A. C. (2016). The manager’s dilemma: a conceptualization of online review manipulation strategies. Current Issues in Tourism, 21(5), 1–20.

Halaburda, H., & Yehezkel, Y. (2016). The role of coordination bias in platform competition. Journal of Economics & Management Strategy, 25(2), 274–312.

Heydari, A., Tavakoli, M., & Salim, N. (2016). Detection of fake opinions using time series. Expert Systems with Applications, 58, 83–92.

Hu, N., Bose, I., Koh, N. S., & Liu, L. (2012). Manipulation of online reviews: An analysis of ratings, readability, and sentiments. Decision Support Systems, 52(3), 674–684.

Jelnov, A., Tauman, Y., & Zeckhauser, R. (2017). Attacking the unknown weapons of a potential bomb builder: The impact of intelligence on the strategic interaction. Games and Economic Behavior, 104, 177–189.

Kirstein, R. (2014). Doping, the inspection game, and Bayesian enforcement. Journal of Sports Economics, 15(4), 385–409.

Landes, W. M., & Posner, R. A. (1984). Tort law as a regulatory regime for catastrophic personal injuries. The Journal of Legal Studies, 13(3), 417–434.

Lee, G., & Raghu, T. S. (2014). Determinants of mobile apps’ success: Evidence from the App Store market. Journal of Management Information Systems, 31(2), 133–170.

Mauri, A. G., & Minazzi, R. (2013). Web reviews influence on expectations and purchasing intentions of hotel potential customers. International Journal of Hospitality Management, 34, 99–107.

Mayzlin, D., Dover, Y., & Chevalier, J. (2014). Promotional reviews: An empirical investigation of online review manipulation. The American Economic Review, 104(8), 2421–2455.

Narudin, F. A., Feizollah, A., Anuar, N. B., & Gani, A. (2016). Evaluation of machine learning classifiers for mobile malware detection. Soft Computing, 20(1), 343–357.

Ott, M., Choi, Y., Cardie, C., & Hancock, J. T. (2011). Finding deceptive opinion spam by any stretch of the imagination. In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies (Vol. 1, pp. 309–319). Association for Computational Linguistics.

Polinsky, A. M., & Shavell, S. (2007). The theory of public enforcement of law. Handbook of Law and Economics, 1, 403–454.

Rahman, M., Rahman, M., Carbunar, B., & Chau, D. H. (2017). Search rank fraud and malware detection in Google Play. IEEE Transactions on Knowledge and Data Engineering, 29(6), 1329–1342.

Resnick, P., & Zeckhauser, R. (2002). Trust among strangers in internet transactions: Empirical analysis of ebay’s reputation system. In The economics of the internet and e-commerce (pp. 127–157). Emerald Group Publishing Limited.

Savage, D., Zhang, X., Yu, X., Chou, P., & Wang, Q. (2015). Detection of opinion spam based on anomalous rating deviation. Expert Systems with Applications, 42(22), 8650–8657.

Schuckert, M., Liu, X., & Law, R. (2016). Insights into suspicious online ratings: direct evidence from tripadvisor. Asia Pacific Journal of Tourism Research, 21(3), 259–272.

Seneviratne, S., Seneviratne, A., Kaafar, M. A., Mahanti, A., & Mohapatra, P. (2017). Spam mobile apps: Characteristics, detection, and in the wild analysis. ACM Transactions on the Web (TWEB), 11(1), 4.

Smith, M. D., & Brynjolfsson, E. (2001). Consumer decision-making at an internet shopbot: Brand still matters. The Journal of Industrial Economics, 49(4), 541–558.

Wang, G., Wilson, C., Zhao, X., Zhu, Y., Mohanlal, M., Zheng, H., & Zhao, B. Y. (2012). Serf and turf: crowdturfling for fun and profit. In Proceedings of the 21st international conference on World Wide Web (pp. 679–688). ACM.

Ye, J., & Akoglu, L. (2015). Discovering opinion spammer groups by network footprints. ECML/PKDD, 1, 267–282.

Zhu, H., Xiong, H., Ge, Y., & Chen, E. (2013). Ranking fraud detection for mobile apps: A holistic view. In Proceedings of the 22nd ACM international conference on information & knowledge management (pp. 619–628). ACM.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.