HKUST BUSINESS SCHOOL 香港科大商學院 WORLD CLASS IN ASIA

Center for Economic Policy 經濟政策研究中心

Merger Analysis in the App Economy: An Empirical Model of Ad-Sponsored Media

Kohei Kawaguchi, Toshifumi Kuroda and Susumu Sato

HKUST CEP Working Paper No. 2021-03

February 26, 2021

HKUST Center for Economic Policy
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
T: 3469 3007 E:cep@ust.hk W: cep.hkust.edu.hk

Merger Analysis in the App Economy: An Empirical Model of Ad-Sponsored Media

Kohei Kawaguchi, Toshifumi Kuroda and Susumu Sato

HKUST CEP Working Paper No. 2021-03
February 26, 2021

Abstract

This paper proposes a new model of imperfect competition of ad-sponsored media, which can sell “free” products, for a merger analysis applicable to the mobile app industry. To analyze developers' monetizing with both price and advertising in an app, we consider a consumer who faces both budget and time constraints. Moreover, to catch up with newly created and quickly redefined markets, we automate the conversion from in-text product descriptions to numerical product attributes by combining word embedding and dimension reduction techniques. The model defines an equilibrium over consumers' downloads, usage, and in-app purchase decisions and app developers' price and non-price competition. We estimate the model using mobile app data from Japan from 2015 to 2017. The estimate revealed that the marginal disutility of watching advertisements is 12.4% of the ad price for games and 3.1% for other apps. The relevant markets defined by a Small, Non-transitory but Significant Increase in Cost (SSNIC) test are larger than the product categories. Merger simulations show that the app market is, at the static level, competitive and even a merger among the top 10 apps has negligible effects on surplus. The proportional transaction fee imposed by the platform, whose welfare implication is ambiguous because it increases the advertisements and decreases the download prices, is more influential. For game apps, the total surplus is maximized at 12%-15% rather than the actual 30%, increasing welfare by 2.4% and app developers' profits by 44%. For other apps, the total surplus curve is almost flat around 30%.

Authors' contact information

Kohei Kawaguchi

Department of Economics, Hong Kong University of Science and Technology
Email: kkawaguchi@ust.hk

Toshifumi Kuroda

Department of Economics, Tokyo Keizai University
Email: kuroda@tku.ac.jp

Susumu Sato

Institute of Economic Research, Hitotsubashi University
Email: susumusato@live.jp

Merger Analysis in the App Economy: An Empirical Model of Ad-Sponsored Media *

Kohei Kawaguchi† Toshifumi Kuroda‡ Susumu Sato§

February 26, 2021

Abstract

This paper proposes a new model of imperfect competition of ad-sponsored media, which can sell "free" products, for a merger analysis applicable to the mobile app industry. To analyze developers' monetizing with both price and advertising in an app, we consider a consumer who faces both budget and time constraints. Moreover, to catch up with newly created and quickly redefined markets, we automate the conversion from in-text product descriptions to numerical product attributes by combining word embedding and dimension reduction techniques. The model defines an equilibrium over consumers' downloads, usage, and in-app purchase decisions and app developers' price and non-price competition. We estimate the model using mobile app data from Japan from 2015 to 2017. The estimate revealed that the marginal disutility of watching advertisements is 12.4% of the ad price for games and 3.1% for other apps. The relevant markets defined by a Small, Non-transitory but Significant Increase in Cost (SSNIC) test are larger than the product categories. Merger simulations show that the app market is, at the static level, competitive and even a merger among the top 10 apps has negligible effects on surplus. The proportional transaction fee imposed by the platform, whose welfare implication is ambiguous because it increases the advertisements and decreases the download prices, is more influential. For game apps, the total surplus is maximized at 12%–15% rather than the actual 30%, increasing welfare by 2.4% and app developers' profits by 44%. For other apps, the total surplus curve is almost flat around 30%.

Keywords: Merger simulation, market definition, SSNIP, antitrust policy, ad-sponsored media, platform transaction fee, app economy, distributed word representation.

JEL Codes: L11, L13, L41, L86, M13, M21.

*This study was financially supported by JSPS grant 18K12768 and 20K22117 and NHK. At the time of the research, Kuroda was a member of NHK's Internet business committee and Japanese Fair Trade Commission (JFTC)'s Study Group on Improvement of Trading Environment surrounding Digital Platforms. The views and opinions in this paper are those of the authors and do not represent those of NHK and JFTC. We thank Akira Mizukami, Pedro Pereira, Masayuki Asahara and Taku Masuda for their helpful comments. We are also grateful for helpful comments and discussions at seminars at Kansai University, Kyoto University, International Christian University, ITS European Conference, SWET, JEES, NBER SI, and Happy Hour Seminar! We thank Akira Matsushita for the research assistance. All errors are our own.

†kkawaguchi@ust.hk, Department of Economics, Hong Kong University of Science and Technology

‡kuroda@tku.ac.jp, Department of Economics, Tokyo Keizai University

§susumusato@live.jp, Institute of Economic Research, Hitotsubashi University

1 Introduction

Defining a relevant market and conducting a merger simulation are cornerstone activities of an antitrust policy. Nonetheless, doing so is not straightforward in the app economy, which is playing an increasingly vital role in shaping the ecosystem of software platforms such as smartphones, tablets, and laptops. This difficulty occurs because the co-existence of multiple monetizing policies such as *freemiums* prevents us from identifying substitution patterns based on a traditional method that primarily uses price variations. Therefore, the authority faces greater challenges when imposing conditions on deals. Thus, the uncertainty in the definition of a relevant market and the assessment using a merger simulation gives interest groups the possibility of manipulating antitrust policies. For example, when *Facebook* attempted to merge with *WhatsApp*, European telecommunication companies encouraged the European Commission to challenge the case because the merged entity would hold a dominant position in the "instant messaging" market.[1] Although the merger was once approved, the U.S. Federal Trade Commissions (FTC) and states later requested a breakup.[2]

The policymakers are also facing a challenge with regulating the vertical relation between app developers and app marketplaces such as *App Store* and *Google Play*. *Spotify* has long complained about 30% transaction fee charged by App Store on the download revenue and in-app purchases.[3] In 2020, *Epic Games* also filed lawsuits against *Apple* and *Google* for the transaction fee.[4] However, it is theoretically unclear whether the current 30% transaction fee is too high or too low because most mobile apps can monetize either through prices or advertising. When they are charged on the non-advertising revenues, they may increase advertisements and reduce prices (c.f., Belleflamme and Toulemonde (2018)). Thus, the welfare implication depends on the market environment, such as consumers' disutility of watching advertisements.

Ways in which the relevant market is defined and the changes in the market power due to a merger and the welfare implications of vertical restraints are evaluated are factors that influence the decisions of the antitrust authorities and their consequences. Any other ad-sponsored media, such as newspaper and cable TV, poses the same problem of antitrust policies.

In this paper, we propose a new framework of ad-sponsored media and apply it to the mobile app industry. To address the co-existence of multiple monetizing policies of ad-sponsored media, we consider a consumer who faces both budget and time constraints and explicitly model the time cost for a consumer to use a free service when watching advertisements. In this setting, in addition to the traditional pecuniary prices to download and use the apps, mobile app developers can effectively set the "price" by increasing mobile advertising intensity. Such an increase will raise the time cost

[1] Case No COMP/M.7217 - Facebook/ WhatsApp `https://ec.europa.eu/competition/elojade/isef/case_details.cfm?proc_code=2_M_7217`, accessed on February 12, 2021.

[2] FTC v. Facebook. `https://www.ftc.gov/enforcement/cases-proceedings/191-0134/facebook-inc-ftc-v`, accessed on February 12, 2021.

[3] Spotify's claim `https://www.timetoplayfair.com/timeline/`, accessed on February 21, 2021 and European Commision press release `https://ec.europa.eu/commission/presscorner/detail/en/IP_20_1073`, accessed on February 21, 2021.

[4] Epic Game's claim `https://www.epicgames.com/site/en-US/free-fortnite-faq`, accessed on February 21, 2021.

for consumers and the revenues that a developer receives from advertisers. We explicitly model this non-price competition of mobile app developers regarding advertising intensity. Furthermore, following Armstrong and Vickers (2001), we transform the original model of price and advertising competition into an equivalent *competition-in-utility* model wherein app developers set the mean utilities served to consumers and then choose the optimal level of prices and advertising intensities to achieve those utilities. This transformation allows us to define the "price" or the "cost" of an app in a unified manner, regardless of the monetizing policies, as the difference in the maximal attainable utility from the app and the achieved utility due to either download price or advertisements. This is particularly useful in the assessment of market power of ad-sponsored media and firms facing a two-sided market.

We use word embeddings to a semantic space (Deerwester et al., 1990) to convert a product description into a numerical vector that is used as a product characteristics vector in the consumer choice model. This approach allows us to catch up with a fast-growing market by automating the translation from product descriptions to numerical product characteristics. By referring to detailed information on product characteristics, we can also avoid colloquialisms when defining a relevant market. We integrate the resulting numeric representation of product descriptions into a consumer choice model and let the choice data reveal the substitution pattern across products. We apply a rigorous post-LASSO method (Belloni and Chernozhukov, 2013) by assuming sparsity in the manner in which the semantic vectors affect consumer choice to identify key dimensions in the product characteristics space. This entire procedure allows us to define a market supervised by choice data rather than based on an unsupervised classification solely using product descriptions.

One of the empirical problems is that we do not observe an app's advertising intensity. At best, we can only observe whether advertisements are shown in an app. Our approach is to elicit the advertising intensity set by developers by exploiting a unique feature of the mobile app industry: the direct marginal cost of acquiring sponsored advertisements is negligible because of the so-called ad-network service. In other words, we use advertising optimality conditions to elicit unobserved advertising intensity instead of identifying the marginal cost parameter. We note that the marginal cost of increasing usage is identified from the pricing optimality condition as usual.

We estimate the model using detailed data about mobile download, revenue, and usage. The estimates show that advertising decreases consumers' utility. The parameter estimates indicate the disutility of watching one unit of advertisements is on average JPY 14 for applications and JPY 56.4 for games. Because the average advertising price is JPY 425.8, these amount to 3.1% and 12.4% of the app's advertising revenue. These numbers are smaller than the estimate of Facebook presented in the study of Benzell and Collis (2020). Based on a survey, they estimated that the disutility of advertising is approximately 20% of Facebook's advertising revenue.

We conduct several policy experiments by relying on the estimated model for mobile apps. First, applying a competition-in-utility framework to the estimated model, we conduct *Small, Non-transitory but Significant Increase in Cost (SSNIC)* test to define relevant markets in the mobile apps market, which extends the *Small, Non-transitory but Significant Increase in Price (SSNIP)* based on the competition-in-utility model. Although it is impossible to conduct standard SSNIP test because of the prevalence of free apps, summarizing advertisements and prices as costs allows us to conduct the SSNIC test. We observed that the relevant market defined by the SSNIC test is larger than the market defined by the product category specified in Google Play. This result indicates that an arbitrary definition of the relevant market can mislead an antitrust authority.

Our model enables us to conduct a full-equilibrium merger simulation. Unfortunately, we cannot name apps because of the confidentiality contract. Thus, we consider a hypothetical merger in which the top app in each product category acquires the 2nd-10th closest apps in terms of cross elasticity. The magnitude of the effects differs across apps but are small in most categories. For example, by the merger among 10 social networking apps, the total surplus drops by less than 0.001%. Even when top 30 apps are merged, the total surplus dropped by only 4.7% for game apps and by 1.9% for other apps. Therefore, at the static level, the app market is competitive.

Finally, we conducted a counterfactual analysis by gradually reducing the platform transaction fee from 30% to 0. In contrast to the standard double marginalization argument, we show that the download prices decrease with the level of platform transaction fees because when effective marginal costs are negative, an increase in proportional fee rate reduces the prices set by developers, and when app developers are ad-sponsored, the effective marginal costs of attractive users are often negative. Therefore, a reduction in transaction fee increases the download price and decreases the advertisements substantially because of the change in the relative profitability of prices and advertising. Consequently, the welfare implication of platform fee is ambiguous. For games, 12%-15% transaction fee maximizes total surplus, resulting in a 2.4% increase in the total surplus and 44% higher app developers' surplus compared to that under 30% transaction fee. For other apps, the total surplus is flat around 30%.

The remainder of this paper is organized in the following manner. In the rest of this introduction, we clarify the novelty and contributions of our paper in combination with an overview of the relevant literature. Section 2 provides an overview of the mobile app market and its institution, and Section 3 explains the data we use for the analysis. Section 4 describes how to numerically represent the in-text app description. Section 5 lays out the model and proposes an algorithm to solve the model. Section 6 derives an estimator for the key structural parameters in the model and 7 describes the estimation results. Section 8 defines the relevant market for several apps based on the estimated model. Section 9 conducts hypothetical split and merger simulations and evaluate the competitiveness of the mobile app market. Section 10 analyzes the effect of platform fee reduction, and Section 11 concludes the paper by restating the contributions and clarifying the limitations of the analysis.

1.1 Novelty and Contributions

Our structural model of competition among mobile app developers can be classified as a model of competition among ad-sponsored media (Anderson and Gabszewicz, 2006). This body of literature analyzes mergers among ad-sponsored media in an environment in which consumers single-home, and advertisers multi-home (Anderson and Peitz, 2020), or both consumers and advertisers multi-home (Anderson et al., 2019). Our model belongs to the former framework. Our model differs from existing theoretical models of mergers among ad-sponsored media in one way: business models (paid media or free media) can change after a merger, whereas existing studies assume that the business model is exogenously given.

Some studies also analyzed in-app purchases as versioning strategies of a monopolistic two-sided platform (Jeon et al., 2016; Lin, 2020). Compared with these studies, by developing a simple model of consumers' in-app purchasing decisions, our model incorporates in-app purchases in an oligopoly framework. Some studies also analyzed the endogenous choice of business models as a device for strategic differentiation (Calvano and Polo, 2019) or as a form of second-degree price discrimination (Sato, 2019), among others, in a different environment. Our model uses non-negativity constraints for prices and advertising intensities to derive the endogenous choice of a business model: when the non-negativity constraint for download price binds, the app is provided for free, and when the non-negativity constraint for advertising intensity binds, the app is provided without advertisements. Given the heterogeneity in app and developer features, this characterization enables an analysis of the co-existence of multiple business models in a single framework.

Some studies used text data for an economic analysis. Each such study numerically represented different information in text data in various ways. Gentzkow et al. (2019) reviewed the exploding body of literature of various fields of economics research using text as data. In the mobile app industry, Liu (2017) and Ershov (2020) used app descriptions to categorize apps. Deng et al. (2018) used app's descriptions to study differences in functions between their paid and free versions. Leyden (2018) used the descriptions of app's release notes to define product categories and distinguish bug fixes and feature updates. Pervin et al. (2019) evaluated user reviews as positive, negative, and neutral. Barlow et al. (2019) and Angus (2019) used product descriptions to measure the similarity of apps. Existing studies manually processed text data, counted word frequency, or used sentiment analysis. Our study differs from their method by using product characteristics represented by a semantic vector obtained through word embedding (Deerwester et al., 1990; Mikolov et al., 2013b).

The following papers used information elicited from text data as part of the product characteristics in demand estimation. Gentzkow and Shapiro (2010) used a slant measure based on text data to estimate the demand for newspapers. Ghose and Han (2014) and Kesler et al. (2017) used several pieces of information in product descriptions such as file size, version, and number of characters as product characteristics. Kwark and Pavlou (2019) judged whether a good is a substitute or a complement for other goods based on product descriptions and then studied the effect of a product's consumer review on its substitutes and complements. Leyden (2018) used the aforementioned information to estimate demand. Following the approach in Ackerberg and Rysman (2005) and Ershov (2020) used the number of products in the categories to control for unobserved product characteristics approach. Ours is the first paper that uses high-dimensional embedding for words in product descriptions as product characteristics to estimate consumer demand.

The body of literature on mobile app demand estimation is growing. Carare (2012) and Ifrach and Johari (2014) estimated the effect of mobile app store rankings on demand. Ghose and Han (2014) estimated the discrete choice random coefficients demand for mobile apps that considers various product characteristics, including in-app purchases, in-app advertising, and the number of updates as fixed characters. Han et al. (2016) estimated a consumer choice model on both mobile app downloads and usage through a discrete-continuous choice framework. Ershov (2020) examined consumer product discovery costs for game apps on the Google Play platform. Leyden (2018) estimated the dynamic discrete choice of a consumer over mobile apps to investigate the effect of product updates. Our paper differs from these studies in multiple dimensions. First, we consider both download and usage decisions over mobile apps. The only exception is Han et al. (2016). However, their data and analysis are at the product category level, whereas ours is at the product level. Second, we explicitly model the interaction between advertising intensity and consumer download and usage choice. Ghose and Han (2014) and Leyden (2018) included an advertisement dummy to estimate demand, but did not consider advertising intensity. Third, we include high-dimensional product characteristics elicited from in-text product information, allowing us to avoid making an assumption about the product category to which each app belongs. Thus, we do not restrict the substitution pattern based on a pre-specified product category. Finally, our data cover a wider variety of mobile apps.

Several papers have studied the strategy of mobile app developers. Ghose and Han (2014) considered the price competition faced by mobile app firms. Ershov (2020) investigated entry as a firm strategy. Leyden (2018) investigated the pricing and update strategy of mobile apps. Liu (2017) investigated app developers' choice of platform. Our paper differs from these studies by jointly considering the pricing and advertising strategies. Our paper is the first to explicitly model and empirically analyze the imperfect competition of mobile app developers over consumer app choice and time usage.

Some studies included the opportunity cost of time usage in a consumer decision problem. Jara-Díaz and Rosales-Salas (2017) reviewed time use studies in transportation research that ranges from purely descriptive studies to econometric modeling analyses. Regarding time usage in the digital economy, Goolsbee and Klenow (2006), Brynjolfsson and Oh (2012) and Pantea and Martens (2016) used the opportunity cost of usage time to evaluate the value of free digital services on the Internet. Han et al. (2016) estimated the utility and satiation of mobile app usage with a multiple discrete-continuous choice model at the app category level. Regarding competition among ad-sponsored media, Crawford et al. (2018) studied households' time allocation problem over TV channels to investigate vertical integration in the TV market. The novelty of our paper is that it integrates into the analysis the supply side's response in advertising. In our model, mobile app developers compete over the time spent by consumers, which affects how consumers allocate time across activities by strategically setting in-app advertising intensity.

Competition authorities in developed countries are concerned with potential anti-competitive practices in the digital economy. However, differences exist in the status of merger regulations. The Japan Fair Trade Commission (JFTC) addressed non-price competition by revising in December

17, 2019, its merger guidelines (Japan Fair Trade Commission, 2019) to evaluate the competitive impact of a merger on the characteristics of content, qualities, and user-friendliness when defining product and geographic ranges in digital services. Nevertheless, Crémer et al. (2019) pointed out the practical difficulty in obtaining a precise measure of digital service quality. The U.S. Department of Justice (DOJ) set up a task force to monitor the information technology industry that addressed these issues. The literature has provided several approaches to defining the relevant market for a product offered through ad-sponsored media. Emch and Thompson (2006) proposed using the sum of the prices of both sides to conduct a version of a hypothetical monopolist test in payment card networks. Evans and Noel (2008) proposed using a relevant market definition based on a critical loss analysis of multi-sided platform. They applied the concept to Google's acquisition of DoubleClick. Filistrucchi et al. (2012) investigated mergers of newspapers using a two-sided market model. Affeldt et al. (2013) extended the concept of the Upward Pricing Presser to two-sided markets and applied it to a hypothetical merger in the Dutch daily newspaper market. Our paper is the first to provide a framework for relevant market definitions when a product's retail price can be free in an equilibrium. Our model differs from the literature on two-sided markets in that either the retail price or advertising can be at the zero boundary or in the interior. In other words, app developers can endogenously select different monetization modes, including zero prices with advertisements, positive prices without advertisements, and both.

Regarding the relevant market definition of mobile apps, previous papers regarded the product category as a relevant market. Ghose and Han (2014) and Ershov (2020) used product categories to set up a nested-logit model. Liu (2017) and Leyden (2018) focused on a few categories of apps, namely, game and productivity apps. We elicited the relevant market for mobile apps from the top apps in Google Play by developing a new framework for estimating the demand for mobile apps.

Certain papers conducted merger simulations among ad-sponsored media. Some of them studied the newspaper industry (Filistrucchi et al., 2012; Fan, 2013; Gentzkow et al., 2014; Van Cayseele and Vanormelingen, 2019). Others studied the radio (Jeziorski, 2014) and magazine (Song, 2011) industries. Our paper is the first to simulate a horizontal merger, in which suppliers can choose monetization mode over retail prices and advertising, and different monetization modes co-exist in the market. Previous papers identified the marginal costs of printing, producing, and acquiring new advertisements from advertising optimality conditions. These costs do not exist in the app economy. App developers can use a Software Development Kit (SDK) for an ad network to automate advertising. We exploit this unique feature of the app economy to elicit advertising intensity through the condition of advertising optimality.

2 Industry Background

2.1 Mobile App Industry

Although complete information on the global app economy is unavailable, several reports provide a fragmented view of this rapidly growing app economy. We sketch the landscape of the app economy during the data period from 2015 to 2017 and provide recent competition policy related issues.

Mobile app Mobile app is an application software designed for mobile devices, such as smartphones and tablets. Smartphones and tablets are multi-purpose mobile computing devices that typically have a touchscreen, Internet access, camera, microphone, speaker, and a specific operating system (OS) that manages the hardware and software. The distinction between smartphones and tablets is unclear. However, smartphones usually provide mobile data access through a cellular network and are smaller than seven inches.

As of 2017, *Android* and *iOS* are the two mainstream OSs. Android is developed by Google and iOS by Apple. In 2017, the OS market share in smartphones was 73.5% for Android and 19.9% for iOS.[5] The market share in tablets was 29.0% for Android and 70.7% for iOS.[6]

Mobile apps take up a significant amount of Internet usage time. App Annie (2017) reported a breakdown of the time spent using the mobile Internet in selected countries. The report indicated that consumers in both developed and developing countries spent more time on mobile apps than on mobile web browsers. For example, in the United States, the ratio of app usage time is 88%. Moreover, consumers are increasingly using the Internet through the mobile Internet. comScore (2017) showed that the 2017 mobile share in the United States was 65%. Thus, understanding consumer behavior in the mobile app industry is essential for understanding consumer behavior on the Internet.

Mobile app stores Consumers can download and install mobile apps from online stores for both OSs. Some apps are free to download, and others have a price attached to them. Mobile apps for iOS can be downloaded only from the App Store but can be downloaded from several stores for the Android OS. Google operates the Google Play as mobile apps store, and other firms operate mobile app stores for Android devices including *Galaxy Store* for *Samsung* devices and *Epic Games* for some games such as *Fortnite*. Nevertheless, in 2017, the majority of the downloaded apps were still from Google Play and App Store. To distribute a mobile app through a mobile app stores, the developer has to pass a review process, and these processes have review policies that differ across mobile app stores.[7]

A mobile app has a page on each mobile store that provides information about the app. The information in the store is described in Section 3.2. Mobile app stores classify apps into several categories. Google play first classifies apps into *Application* and *Games*. Then, it classifies them into categories such as "Social", "Music and Audio" and "News and Magazines" for application apps, and "Action", "Puzzle" and "Sports" for game apps to enable consumers to easily find a

[5] Mobile operating system market share worldwide in 2017, statcounter, `http://gs.statcounter.com/os-market-share/mobile/worldwide/2017`, accessed on February 13, 2021.

[6] Tablet operating system market share worldwide in 2017, statcounter, `http://gs.statcounter.com/os-market-share/tablet/worldwide/2017`, accessed on February 13, 2021.

[7] The review guideline for App Store is available at `https://developer.apple.com/app-store/review/guidelines/` and the review guideline for Google Play is available at `https://play.google.com/intl/ja/about/developer-content-policy-print/`, accessed on February 13, 2021.

desired app. The details of the categories are also described in Section 3.2. Figure 1 provides an example of a page on Japanese Google Play in 2018.

Mobile app developers Mobile app developers face a two-sided market of consumers and advertisers. Both sides of the market have grown rapidly during the data period. According to App Annie (2017), the number of mobile app downloads increased by 60% between 2015 and 2017 and amounted to more than USD 175 billion in 2017. App Annie (2019) also reported that mobile ad sales increased by 30% during 2017 and mobile ads were expected to account for 62% of global digital ad spend in 2018, representing USD 155 billion, an increase from 50% in 2017.

Revenues from consumers consist of priced downloads and in-app purchases. The download price is usually charged only when a consumer downloads the app for the first time. A consumer who purchased an app is allowed to download the app multiple times without paying extra and can use the app on multiple devices. Of course, some apps restrict the number of devices on which a consumer can use them or issue licenses that restrict this number. An app developer can also collect in-app purchases through mobile app stores. A consumer pays within an app to remove restrictions on the app's functionality or to upgrade the service. For example, a consumer may pay to suppress mobile ads or purchase an item in a game.

Mobile app developers also have a vertical relationship with the app stores. The app stores charge a transaction fee on the revenues from download and in-app purchases. Both Google Play and App Store charge a transaction fee of 30%, and developers earn only 70% of the download and in-app purchase revenues. Some app developers attempted to collect revenues outside the app stores. However, the review guideline of Google Play and App Store prohibit this practice.

Mobile ad networks Another source of revenue for a mobile app developer is advertising fees that advertisers pay to display their advertisements on the app. Most advertisers and mobile apps use a service that connects advertisers and websites or apps, an *ad network*, to distribute and host advertisements. Some mobile apps choose not to use an ad network and sell advertising space directly to advertisers. They do so to reduce the transaction fees paid to ad networks and to target specific advertisers by taking advantage of their app's unique customer base.

In 2018, more than 250 mobile ad networks were in operation.[8] An ad network distributes software development kits (SDK) to integrate ads into mobile apps. An ad network then allows advertisers to specify parameters, such as region, device, OS, interests, and gender, to determine the target audience. Advertising space is usually transacted through an auction. For example, in Google's *AdMob* ad network, advertisers can bid on a per click or impression basis. AdMob ranks between click bids and impression bids in order of expected revenue to predict the likelihood that a click bid ad will be clicked. For the developer side, mobile app developers set a price floor. Then, AdMob distributes ads only to websites and apps that have expected revenue higher than the price floor. AdMob also provides advertisers with an optimizer that dynamically sets price

[8] Available at `https://www.appsflyer.com/2018indexpage/`, accessed on February 13, 2021.

floors depending on a geographic location, traffic, and other pieces of historical data.[9] Other than AdMob, other services assist mobile app developers with hosting mobile ads through multiple ad networks. *InMobi* provides an ad mediation platform that assists mobile apps with hosting mobile ads from the highest bidder across multiple ad networks.[10] Because of the high number of ad networks and apps that accept ads in the market, the cost of ad-network is far lower than direct selling.

Recent antitrust and merger cases During the past two decades, high-tech titans (Gilbert, 2020) including Facebook and Google acquired many start-up firms. Google acquired *YouTube* for USD 1.65 billion in 2006. App Annie (2017) reported that YouTube was the most used video streaming app in the United States in 2017. This acquisition completed this case in early termination. In contrast, Facebook acquired Instagram in 2012 for USD 1 billion and WhatsApp in 2014 for USD 19 billion. The antitrust authority in the United States and the European Union approved these mergers after a detailed merger review. App Annie (2017) reported that Facebook, Messenger, and Instagram are the top three apps by monthly active users in the United States. In addition, WhatsApp is the most used social app in Germany, Indonesia, India, Russia, Spain, and the United Kingdom, and its merger of Facebook appears to have relaxed the market competition in the social app market. In addition, the European Union fined Facebook EUR 110 million (USD 122 million) for providing misleading information on its merger with WhatsApp. In 2021, the FTC and states in the U.S. requested a breakup and the debate is still ongoing.

Another recent merger case that involves a high-tech titan is the Google/Fitbit case. In 2019, Google announced to acquire *Fitbit* that sells health and fitness smartwatch. The European Commission and JFTC approved it with a number of conditions, such as the prohibition of the use of Fitbit health data for ad targeting.[11] In January 2021, Google announced that it had closed the deal for USD 2.1 billion, although it was under review by the U.S. DOJ and the Australia's Competition & Consumer Commission.[12]

There are also ongoing antitrust cases regarding the vertical relation between app developers and app stores. Spotify claimed that the transaction fees are used to protect *Apple Music* (Recode, 2016). In 2020, the European Commission launched a formal investigation into Apple to address Spotify's claim.[13] In 2020, Epic Games also took a legal action on Apple and Google's restrictions on app stores in the U.S., Australia and the European Union.[14]

Apple and Google reduced transaction fees to 15% for consumers whose subscription terms

[9]See `https://support.google.com/admob/answer/3418058?hl=en`, accessed on February 13, 2021.

[10]Available at `https://japan.inmobi.com/advertising-cloud/mediation`, accessed on February 13, 2021.

[11]European Commission press release is available at `https://ec.europa.eu/commission/presscorner/detail/en/ip_20_2484`, and JFTC press release is available at `https://www.jftc.go.jp/en/pressreleases/yearly-2021/January/210114.html`, accessed on February 13, 2021.

[12]See `https://www.theverge.com/2021/1/14/22188428/google-fitbit-acquisition-completed-approved`

[13]European Commission press release is available at `https://ec.europa.eu/commission/presscorner/detail/en/IP_20_1073`, accessed on February 21, 2021.

[14]Epic Game's claim `https://www.epicgames.com/site/en-US/free-fortnite-faq`, accessed on February 21, 2021.

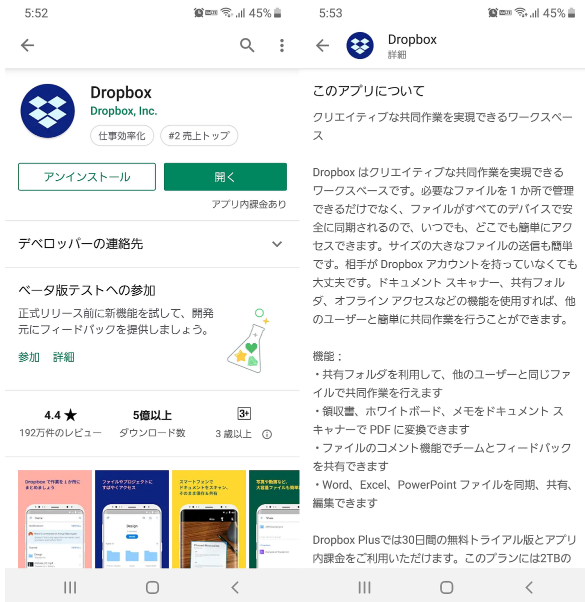


Figure 1: Product description in Google Play

went beyond 1 year in 2016 and 2018, respectively.[15][16] Furthermore, Apple introduced "App Store Small Business Program," which reduces transaction fee to 15% for small businesses earning up to $1 million per year in 2021.[17]

3 Data

The data we use to estimate the model come from several sources. First, we use the data provided by the consulting company *App Annie* to construct app download, usage, in-app purchase, and market size data. Second, we collect information on Google Play using the web scraping method and combine them with similar data provided by App Annie to complete the product description and characteristics data. Third, we use data provided by the mobile ad platform *Adtapsy* to construct unit advertisement price data. Because the App Annie database contains complete information on iOS only after June 2018, when we lack information on advertisement price data, the subsequent analysis focuses on Android apps.

[15] Available at `https://support.google.com/googleplay/android-developer/answer/112622?hl=en`, accessed on February 13, 2021.

[16] The news report Apple's reduction of transaction fee. See `https://techcrunch.com/2016/06/08/apple-to-introduce-search-ads-on-app-store-along-with-changes-to-app-review-discovery-and-splits/`, accessed on February 13, 2021.

[17] See `https://www.apple.com/newsroom/2020/11/apple-announces-app-store-small-business-program/`, accessed on February 13, 2021.

3.1 Download, Usage, Download Price and In-app Purchase

Source App Annie is a consulting company that surveys, collects, assembles, processes, and sells a mobile app database. The App Annie API allows us to extract data on a wide variety of apps in more than 150 countries worldwide that are distributed through the App Store or Google Play. The company combines statistical models and procedures to estimate download, usage, revenue, and several other variables of each mobile app using data from key mobile app stores, key ad networks, proprietary consumer panel surveys, in-app tracking information, and publicly available data. App usage is defined as the number of minutes that it runs in the foreground. Apps in the background are not recorded as in use.

Coverage, period, and selection Because the same app can be sold with different names across different platforms, the company assigns a unique identifier to each app. We use the list of unique app identifiers as the list of products. The company classifies apps first into "Game" and "Application" and then into finer categories, such as news, music, and education apps. For every unit of the observation period daily, weekly, and monthly the company calculates for each app the number of downloads, the revenue, and its rank within each category. The API only allows us to access data on the top 1,000 apps in each sub-category and during a period for each variable. We use daily data as the baseline, if available, and aggregate them depending on the type of analyses. For variables that are only available weekly, we use weekly data. Because the day $\times$ category is a fine enough segment, apps below the top 1,000 have almost zero downloads and revenues.

The data are available since March 2010 for iOS apps and since January 2012 for Android apps. However, because the unit advertisement price data are only available from March 2015, we use data between March 2015 and January 2017 in the estimation. We select the set of apps to be analyzed in the following manner. First, we use information on price and whether each app appears in-app advertisements to classify apps into three business models: free advertising apps, paid advertising apps, and paid non-advertising apps. Next, we compute the fraction of each business model relative to free advertising apps. Using this fraction, for each week and business model, we select the apps ranked higher than the threshold rank, defined by 100 multiplied by the fraction of the business model relative to free advertising apps. The ranking is in either usage time or number of downloads. Finally, we select the apps ranked higher than the threshold rank of 10 times or more in usage or download. The selected apps are the set of apps to be included in the sample.

For each app selected using these criteria, for some weeks, download, usage, or revenue information is missing because the app was not ranked higher than 1,000 but may have operated during those weeks. We fill in these missing values by substituting the minimum value of the observed data in the same categories.

Variables For each app, the data contain product name, developer name, parent company name, product category, devices available, release date, and download price if it is not free. For each app and period (daily, weekly, and monthly), the data contain the number of active users, which is defined by the number of unique users who opened the app during each period (only weekly), the usage penetration rate, which is defined as the number of active users divided by the number of active devices (only weekly), the number of downloads, the average time spent by active users (only weekly), revenues during the period, and several other variables that are not used in our analysis. The one drawback to the data is that the price information does not reflect sales discounts.

Because an app's revenues include the revenues from both downloads and in-app purchases, we subtract the price times the number of downloads from the revenues of each app to calculate the revenues from in-app purchases. The in-app purchase per user is calculated as the revenues from in-app purchases divided by the number of downloads.

We multiply the number of active users and the penetration rate of an app to calculate the estimated number of active devices. We use this value as a proxy for the size of the consumer base for mobile apps. We assume that a consumer has a unit download demand per day and define the market size as the number of active devices multiplied by the number of days in a period. To ensure that the sum of the market shares in each period does not exceed one and evolves steadily, we conduct the following calculation to modify the market sizes. We first multiply the estimated number of active devices in all periods by a constant number. We then calculate the total download share in each market as the sum of downloads of all apps in the sample divided by the number of active devices. Based on the calculation of total download shares, we compute the trend of the total download shares. Finally, we replace the market size of each market with the one that fits the trend in the total download shares.

Summary statistics Table 1 provides the summary statistics for usage time, in-app purchase per download, number of downloads, and download price. The average usage time and in-app purchases are higher among games (3.6 h and JPY 1,790 per week) than among applications (1.2 h and JPY 161 per week). However, number of downloads is higher among applications (17,037 per week) than games (6,732 per week).

Figure 2 demonstrates the market share and Herfindahl-Hirschman Index (HHI) based on the Google Store categories. The upper panel is for applications and the lower panel is for games. Figure 3 summarizes the time series for advertising price, hourly wage, and market size. It indicates that market share based on downloads and usage can be substantially different. In terms of the download, the market structure of each category is between competitive and mildly concentrated.

3.2 Product Description

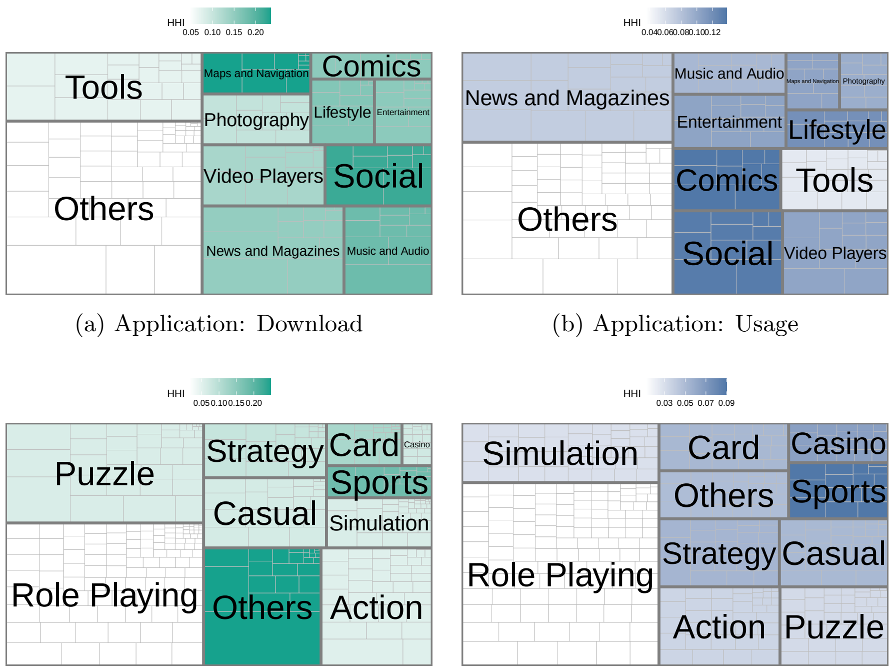
Source We use product descriptions displayed on Google Play to construct the advertising dummy, product class, and semantic vectors. We use App Annie's record of app descriptions and html files of apps in Google Play ,which was obtained by scraping the websites of Google Play as of December 2018. App Annie records a history of product descriptions in one language from among English, Japanese and other languages. It records descriptions of apps that are deleted from Google Play. The html files scraped from Google Play contains descriptions of apps written

Table 1: Summary statistics at the week/app-level

	N	Mean	SD	Median	Min	Max
Application						
Usage time (Hour/User)	17800	1.2	1.2	0.8	0.0	13.7
In-app charge per download (JPY)	17800	160.8	769.9	0.0	0.0	8458.9
Download	17800	17037.2	25876.6	9498.0	0.0	369601.0
Download price (JPY)	17800	9.5	62.9	0.0	0.0	604.3
Game						
Usage time (Hour/User)	33286	3.6	3.2	3.3	0.0	31.3
In-app charge per download (JPY)	33286	1790.2	2795.5	52.9	0.0	8458.9
Download	33286	6732.0	52847.1	1300.0	0.0	8670831.0
Download price (JPY)	33286	31.9	227.3	0.0	0.0	2871.3

Table 2: Share of monetization types at the app-level for each category

	With ad.	With price
Application		
Comics	0.700	0.000
Entertainment	0.769	0.077
Lifestyle	1.000	0.000
Maps and Navigation	0.636	0.000
Music and Audio	0.692	0.077
News and Magazines	1.000	0.000
Photography	0.909	0.000
Social	0.727	0.000
Tools	0.917	0.042
Video Players	0.786	0.071
Game		
Action	0.615	0.000
Card	0.739	0.000
Casino	0.895	0.158
Casual	0.655	0.034
Puzzle	0.805	0.000
Role Playing	0.543	0.034
Simulation	0.732	0.000
Sports	0.786	0.000
Strategy	0.250	0.000



(a) Application: Download

(b) Application: Usage

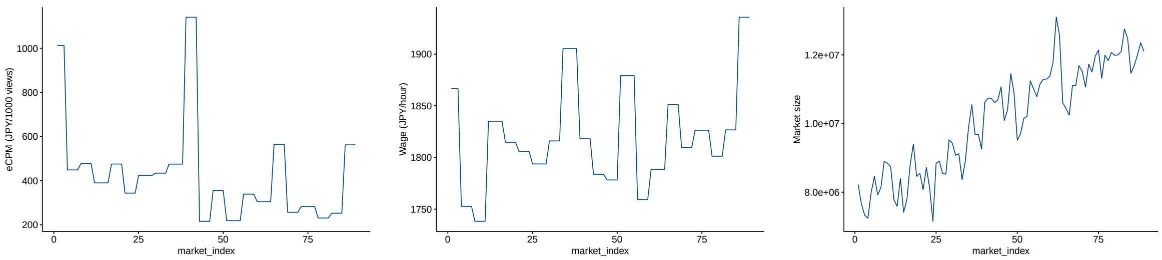
(c) Game: Download

(d) Game: Usage

Figure 2: Market share and Herfindahl–Hirschman Index (HHI) defined by Google Store category

Table 3: Share of apps with advertising and download price across categories

	With ad.	With price
Application		
Comics	0.700	0.000
Entertainment	0.769	0.077
Lifestyle	1.000	0.000
Maps and Navigation	0.636	0.000
Music and Audio	0.692	0.077
News and Magazines	1.000	0.000
Photography	0.909	0.000
Social	0.727	0.000
Tools	0.917	0.042
Video Players	0.786	0.071
Game		
Action	0.615	0.000
Card	0.739	0.000
Casino	0.895	0.158
Casual	0.655	0.034
Puzzle	0.805	0.000
Role Playing	0.543	0.034
Simulation	0.732	0.000
Sports	0.786	0.000
Strategy	0.250	0.000



(a) Advertising price (b) Hourly wage (c) Market size

Figure 3: Summary statistics at market-level

by app developers in several languages. It also contains an automated translation made by Google. To obtain as much information as possible from the description of apps, when it is available, we use Japanese descriptions written by app developers in html files. When it is not available, we use machine translation of descriptions written by app developers in other languages. For apps that have been removed from Google Play, we use App Annie's latest records. For the apps that App Annie recorded in Japanese, we use them. For the apps that App Annie recorded in other languages than Japanese, we use the machine translation of them. We employ *Microsoft Azure Translator* to machine translate app descriptions that are written in other languages.

App Category Google Play classifies apps into 49 categories: 32 are application apps and 17 are game apps. However, as a result of the selection of apps, only 10 categories in application apps and 9 in game apps are used. The other categories are aggregated into "Others" and used as the base category. We use those categories as the product category dummy.

However, as app categories in Google Play are set by app developers[18], the rules for categorizing apps are not uniform. Similar apps can be classified into different categories and apps with completely different features can be put in a category. Therefore, metrics that quantify app characteristics in more detail than categories are required to capture the substitution pattern of apps.

Advertising Dummy Our original data scraped from Google Play contain information that indicates whether or not the app shows ads. We define the advertising dummy as having a value of 1 when the app's store page contains "Contains Ads" strings in a predetermined place. Table 2 represents the share of apps showing advertisements and download prices charged in each category. Most of the apps show some advertisements but the ratio is relatively lower in games. Few apps charge the download price.

[18]See `https://support.google.com/googleplay/android-developer/answer/9859673`, accessed on February 13, 2021.

Morphological analysis To use product descriptions data in demand estimation, we first convert the sentences in the product description into a bag of words by parts of speech. In Japanese, because words are not separated in a sentence, we employ a morphological analysis engine to split Japanese sentences into a bag of words and classify words into parts of speech. Specifically, we use an open-source program called *MeCab* (version 0.996) (Kudo, 2005). MeCab is widely used in the Japanese natural language processing literature to decompose sentences into a bag of words. We use a neologism dictionary for MeCab as a word dictionary, which can be downloaded from the developer's website (Sato et al., 2017). MeCab also distinguishes the parts of speech. We made a bag of words for each parts of speech that is converted into multi-dimensional vectors using the methods described in Section 4.

3.3 Advertisement Price

Source *Adtapsy* is a mobile app advertising platform that matches advertisers with app developers and distributes advertisements through matched apps. Adtapsy is connected to several global ad networks that operate in Japan, such as AdMob, *AdColoy*, *InMobi* and *AppLovin*.

CPM and eCPM The advertisement price that an advertiser pays to show a unit of an advertisement on an app is determined through auctions and can differ across ad networks, advertisers, apps, and devices. A popular buying method is based on CPM, or cost per mille (Latin word for thousands), and represents a fixed price to buy 1,000 ad impressions. If an advertiser buys an ad only in CPM units, the actual price to buy 1,000 ad impressions coincides with the CPM. However, in reality, ad impressions are transacted in various units and formats. Therefore, *eCPM*, or the effective CPM, is the actual costs per 1,000 ad impressions, and is often used as a measure of the market price of ad impressions. According to Adtapsy's estimates, the average eCPM was USD 5.4 and USD 6.3 for Android and iOS in March 2015 and USD 2.2 and USD 2.9 in January 2017, respectively.

Market average eCPM Adtapsy has published a monthly time series of market average eCPM since April 2015. Because these data are the only ones available for mobile advertisement prices, to the best of our knowledge, we use them as the market price of an ad impression in the mobile app industry. The price is unbiased if the ad impression is a homogeneous product but can be biased to the degree that the mobile app is differentiated in the mobile ad market. Thus, our analysis should be extended with reservations for large mobile platforms with impressions when such platforms may have different values and may have market power in the mobile ad market.

3.4 Auxiliary Data

We use wage data as a proxy for the opportunity cost of mobile app usage. We obtain wage data for each age and gender class from the *Basic Survey on Wage Structure*, a survey by the *Labour*

Standards Inspection Offices.[19]

4 Numerical Representation of Product Description

We use *nwjc2vec*, developed by Asahara (2018), to numerically represent in-text product descriptions. The nwjc2vec is publicly available for Japanese language embedding and employs *fastText* (Bojanowski et al., 2017) as the model and the *National Language Web Corpus* (Asahara et al., 2014) as data. Several implementations of *word embedding* methods exist, including *word2vec* (Mikolov et al., 2013b) and *GloVe* (Pennington et al., 2014). fastText is a popular implementation that incorporates *distributional statistics* and word-internal structures into word embeddings. We use nwjc2vec to transform app descriptions into 300-dimension semantic vectors.

4.1 Distributional Hypothesis and Word Embedding

The algorithm is based on the so-called *distributional hypothesis* (Firth, 1957). For example, consider a situation in which the weather news reports today's weather. Both "sunny" and "raining" fit into a context such as "It's ___ today". A word such as "birthday" could also fit into the context. However, we consider a sentence such as "It has been ___ lately," to which "sunny" and "raining" fit but "birthday" does not. In this way, we can construct a matrix that records the fit of words into different contexts. This matrix is called the distributional statistics of words, and a column in this matrix is called a corpus. The distributional hypothesis assumes that the similarity in the distributional statistics implies similarity in the semantics.

We can estimate multi-dimensional numerical vectors of real numbers that well approximate the meanings of words represented in distributional statistics. The resulting multi-dimensional semantic vector space is called *word embeddings*. Word embeddings are obtained by maximizing the likelihood of distributional statistics under a model that predicts a corpus. The model differs in how it defines contexts and relates contexts with words. The model in nwjc2vec uses local word neighborhoods in a sentence as contexts as well, as in Mikolov et al. (2013a) and Bojanowski et al. (2017). The model uses both *continuous bag-of-words (CBOW)* and *skipgrams* to relate contexts to a corpus. We use a model based on skipgrams, because they are often reported to outperform CBOW (Mikolov et al., 2013b; Eisenstein, 2019).

In addition to distributed statistics, fastText also exploits word-internal structures to estimate word embeddings. fastText assumes that a word vector should be consistent with the sum of the vectors of n-grams in the word. For example, the word "phone" is divided into n-grams such as "pho," "phon," "phone," "hone," and "on." The model assumes that two n-grams sharing either similar former or latter n-grams also share a similar meaning. By doing so, the model predictions are robust to differences in tenses such as "go," "goes," and "gone"; forms such as "decide," "decision," and "decisive"; and synonyms such as "economy," "economic," and "economist."

[19]https://www.mhlw.go.jp/english/database/db-l/wage-structure.html

4.2 NWJC2VEC

The National Language Web Corpus used by nwjc2vec is a Japanese language corpus constructed by the *National Institute for Japanese Language* that targets the 10 billion words used on web sites. The corpus first used a program called *Heritrix* to crawl approximately 100 million URLs every three months starting in October 2012. The version of nwjc2vec that we use to build the product characteristics uses data crawled from October 2014 to December 2014.

Hyperparameters exist to train the fastText model. nwjc2vec chooses 300 as the dimension of word embeddings, a local neighborhood size of h_{max} is 8, the number of negative samples is 25, and the range of character lengths of n-grams is $3 - 6$.

4.3 Converting word vectors into a product characteristics

Stopwords Stopwords are the words that are used too often to differentiate sentences, such as "a" and "the." We use the percentage of apps using a word as criteria to define stopwords. We tried 100%, 95%, 75%, 50%, and 25%. For each part of speech, such as noun and verb, we chose criteria that made the p-value of the joint significance of the high-dimensional features the lowest.

Semantic vectors of product As word vectors are numerical vector of words, we have to convert numerical vectors of words in a product description into the semantic vectors of the product. We use the weighted average of numerical vectors of words in an app as semantic vectors of the product. Inverse frequency of words in each app category are used as weight.[20]

4.4 Conversion Procedure

We use nwjc2vec to convert app descriptions described in Section 3.2. We use an open-source *Python* library *gensim* (version 3.7.2) (Řehůřek and Sojka, 2010) to construct numerical representations of product descriptions as follows.

1. Build data that record each app's identifiers and the app's Japanese descriptions.
2. Split Japanese descriptions into a bag of words by parts of speech using Mecab.[21]
3. Load nwjc2vec using gensim.models.fasttext.
4. Convert each words into multi-dimensional vectors using gensim.models.fasttext.
5. Estimate low-dimensional specification.
6. Define stopwords by parts of speech by using p-value of rigorous lasso.
7. Take a weighted average of the multi-dimensional vectors by parts of speech.
8. Take an average of the multi-dimensional vectors across parts of speech.

[20]We finds weighed average of numerical vectors of words has higher correlation with ξ_d and ξ_u of low-dimensional estimation described Table 4 and 5 than non-weighted average.

[21]The number of words in our data is 22,118.

5 Model

In this section, we present a model of consumer's choice for mobile apps and app developer's pricing and non-pricing competition. The term *market* in this section means the sets of all apps at a time, and differs from a *relevant market* that is constructed for making antitrust policy decisions. In this section, we suppress the index of a market.

5.1 Setting

Population and covariates Consider a market with a set of apps $\mathcal{J} \coloneqq \{1, \ldots, J\}$ provided by a group of app developers $\mathcal{D} \coloneqq \{1, \ldots, D\}$. For each app, the developer can set the download price $F_j \in \mathbb{R}_+$ and in-app advertising intensity $a_j \in \mathbb{R}_+$. The market has a unit mass of consumers in the market. Each consumer has a unit download demand and decides on the app to download, how much to use the app, $q_j \in \mathbb{R}_+$, and how much to spend on the in-app purchases when using the app, $e_j \in \mathbb{R}_+$. Let w be the opportunity cost of a unit time for a consumer, that is, the wage.

When analyzing mobile apps, distinguishing utilities from an app's foreground and background processes of an app is important because the former requires consumers to spend their time, whereas the latter does not. For example, playing a game usually requires consumers to open and manually control the app. In contrast, an anti-virus software runs in the background and consumers only have to spend some time setting up the app after downloading it. In the following, we refer to the utility from a foreground process as *usage-related utility* and the utility from a background process as *download-related utility*, and model them separately. A usage-related utility should be a function of usage time whereas the download-related utility should be independent of usage time.

Let $X_{uj} \in \mathbb{R}^{K_u}$ and $X_{dj} \in \mathbb{R}^{K_d}$ be the observed characteristics of the app that affect a consumer's usage- and download-related utilities of a consumer. Let $\xi_{uj} \in \mathbb{R}$ and $\xi_{dj} \in \mathbb{R}$ be the characteristics of the app that affect a consumer's usage- and download-related utilities of a consumer but that are not observed to an econometrician. We assume that ξ_{uj}, ξ_{dj}, X_{uj}, and X_{dj} are mutually independent and $\mathbb{E}\{\xi_{uj}\} = \mathbb{E}\{\xi_{dj}\} = 0$.

Consumer preference The indirect utility from downloading and using app j for consumer i, u_{ij}, consists of usage-related and download-related components as follows:

$$u_{ij} \coloneqq S_j + \beta_{di}' X_{dj} - \alpha_y F_j + \xi_{dj} + \varepsilon_{ij} \tag{1}$$

where

$$S_j \coloneqq \max_{q_j, e_j} \{v_j(q_j, e_j, a_j, w, X_{uj}, \xi_{uj})\} \tag{2}$$

is the benefit from the optimal usage choice. The benefit from usage is assumed to have the following functional form:

$$v_j(q_j, e_j, a_j, X_{uj}, \xi_{uj}) \coloneqq [\beta_u' X_{uj} - \alpha_{aj} a_j - \alpha_y w + g_j(e_j, q_j) + \xi_{uj}] q_j - \psi_j(q_j) - \alpha_y e_j, \tag{3}$$

where ε_{ij} is an idiosyncratic taste shock distributed according to an i.i.d. type-I extreme-value distribution. We allow for β_{di} to have random coefficients as:

$$\beta_{di} \coloneqq \beta_d + \Sigma\nu_i, \tag{4}$$

with a K_d-dimensional random variable ν_i each of whose elements is drawn from an i.i.d. standard normal distribution. $\beta_u \in \mathbb{R}^{K_u}$ and $\beta_{di} \in \mathbb{R}^{K_d}$ represent the consumer's tastes for the characteristics, $\alpha_y \in \mathbb{R}_+$ is the utility from money, and α_{aj} is the disutility from being revealed to a unit advertisement in app j. We allow α_{aj} to vary across apps depending on observable characteristics and specify the form of α_{aj} as follows:

$$\alpha_{aj} = \alpha_{a1} \frac{\exp\left(\alpha_{a2}' X_{\alpha_a j}\right)}{1 + \exp\left(\alpha_{a2}' X_{\alpha_a j}\right)}, \tag{5}$$

where $X_{\alpha_a j} \in \mathbb{R}^{K_{\alpha_a}}$ is observed characteristics of the app that affect the disutilitiy of advertisements, and $\alpha_a \coloneqq (\alpha_{a1}, \alpha_{a2})$ is the parameter that determines the value of α_{aj}. We do not allow for the other parameters to have consumer-specific random coefficients because of a computational issue that we explain in detail in the relevant section. We expect that the inclusion of random coefficients in β_{di} should already allow for a flexible substitution pattern across apps.

Additional functional-form assumptions To obtain an analytical solution and facilitate computations while maintaining flexibility, we specify the functional forms of g_j and ψ as follows:

$$g_j(e_j, q_j) \coloneqq \sqrt{\xi_{ej} e_j / q_j}, \tag{6}$$

$$\psi_j(q_j) \coloneqq \frac{\eta_j}{2} q_j^2, \tag{7}$$

where $\xi_{ej} \in \mathbb{R}_+$ represents the characteristics of the app that affect the usage utility of a consumer through in-app purchases but is not observed to an econometrician. $\eta_j \in \mathbb{R}_+$ is the degree of satiation from usage, which is specified as follows:

$$\eta_j = \eta_1 \frac{\exp\left(\eta_2' X_{\eta j}\right)}{1 + \exp\left(\eta_2' X_{\eta j}\right)} + 0.05, \tag{8}$$

where $X_{\eta j} \in \mathbb{R}^{K_\eta}$ is observed characteristics of the app that affect the degree of satiation, and $\eta \coloneqq (\eta_1, \eta_2)$ is the parameter that determines the value of η_j. Because the model becomes numerically unstable as η_j approaches 0, we put a lower bound on it by adding 0.05. The estimate shows that this lower-bound is not binding.

As a result, the benefit from the optimal usage choice takes the following form:

$$S_j = \max_{q_j, e_j} \left\{ \left(\beta_u' X_{uj} - \alpha_{aj} a_j - \alpha_y w + \sqrt{\frac{\xi_{ej} e_j}{q_j}} + \xi_{uj} \right) q_j - \frac{\eta_j}{2} q_j^2 - \alpha_y e_j \right\}, \tag{9}$$

and the indirect utility takes the following form:

$$\begin{aligned} u_{ij} &= S_j - \alpha_y F_j + \beta_{di}' X_{dj} + \xi_{dj} + \varepsilon_{ij} \\ &\coloneqq \delta_j + \nu_i' \Sigma X_{dj} + \varepsilon_{ij}, \end{aligned} \tag{10}$$

where δ_j is the common mean indirect utility of consumers from app j.

5.2 Consumer's Problem

Usage and in-app purchase decisions Next, we solve the consumer problem. Let $X_j = (X_{uj}', X_{dj}')'$, $\xi_j = (\xi_{uj}, \xi_{dj}, \xi_{ej})$, $\theta_i = (\alpha_{aj}, \beta_u', \alpha_y, \eta_j, \beta_{di}')'$, and $\theta = (\alpha_{aj}, \beta_u', \alpha_y, \eta_j, \beta_d', \mathrm{vec}(\Sigma)')'$. By solving the first-order condition for in-app purchase e_j, we obtain the following relationship between in-app purchase and usage:

$$e_j = \frac{1}{4\alpha_y^2} \xi_{ej} q_j. \tag{11}$$

By solving the first-order condition for usage q_j and inserting equation (11), we obtain:

$$q_j = \tilde{q}_j(a_j, w, X_j, \xi_j; \theta) \coloneqq \max\left\{ \frac{1}{\eta_j} \left(\beta_u' X_{uj} - \alpha_{aj} a_j - \alpha_y w + \frac{\xi_{ej}}{4\alpha_y} + \xi_{uj} \right), 0 \right\}. \tag{12}$$

By inserting this back into equation (11), we obtain:

$$e_j = \tilde{e}_j(a_j, w, X_j, \xi_j; \theta) \coloneqq \frac{\xi_{ej} \tilde{q}_j(a_j, w, X_j, \xi_j; \theta)}{4\alpha_y^2}. \tag{13}$$

By substituting equation (12) into equation (9), we obtain the following usage surplus function:

$$S_j = S(a_j, w, X_j, \xi_j; \theta) \coloneqq \frac{\eta_j}{2} \tilde{q}_j^2(a_j, w, X_j, \xi_j; \theta), \tag{14}$$

which leads to the mean indirect utility δ_j of consumers from app j:

$$\begin{aligned} \delta_j =& \delta_j(a_j, F_j, w, X_j, \xi_j; \theta) \\ \coloneqq& S(a_j, w, X_j, \xi_j; \theta) + \beta_d' X_{dj} - \alpha_y F_j + \xi_{dj}. \end{aligned} \tag{15}$$

Download decision Next, we derive the probability that a consumer downloads an app. Let $a = (a_j)_{j \in \mathcal{J}}$, $F = (F_j)_{j \in \mathcal{J}}$, $X = (X_j)_{j \in \mathcal{J}}$, w, and $\xi = (\xi_j)_{j \in \mathcal{J}}$. Under the assumption that ε_{ij} follows an i.i.d. type-I extreme-value distribution, the probability that a consumer downloads app j is:

$$s_j = \tilde{s}_j(a, F, w, X, \xi; \theta) \coloneqq \int_{\mathbb{R}^{K_d}} \frac{\exp\left[\delta_j(a_j, F_j, w, X_j, \xi_j; \theta) + \nu_i' \Sigma X_{dj}\right]}{1 + \sum_k \exp\left[\delta_k(a_k, F_k, w, X_k, \xi_k; \theta) + \nu_i' \Sigma X_{dk}\right]} dG_{\nu_i}(\nu_i). \tag{16}$$

5.3 Developer's Problem

Developer's profit Now consider an app developer's decisions related to its apps' download prices and advertising intensity. Let $\mathcal{J}_d \subset \mathcal{J}$ be the set of apps that developer d sells. A developer's profit is the sum of profits from each app, as follows:

$$\Pi_d(a, F, X, w, \xi; \theta) \coloneqq \sum_{j \in \mathcal{J}_d} \pi_j(a, F, X, w, \xi; \theta), \tag{17}$$

and the profit from each app consists of the revenues from downloads, in-app purchases, and advertisements:

$$\begin{aligned} &\pi_j(a, F, X, w, \xi; \theta) \\ \coloneqq &s_j(a, F, X, w, \xi; \theta) \left\{(1-\rho)[F_j + e_j(a, F, X, w, \xi; \theta)] + q_j(a, F, X, w, \xi; \theta)(a_j r - \lambda_j) - \epsilon_j\right\}, \end{aligned} \tag{18}$$

where r is the advertising revenue per unit of advertisements shown to the consumer, ρ is the transaction fee rate that app developers pay to the app platform (i.e., Apple App Store and Google Play Store) for each download of the apps and the in-app purchases, and λ_j is other marginal cost for the usage and ϵ_j is other marginal cost on download. We allow λ_j to vary according to the observable characteristics, and it is specified as follows:

$$\lambda_j = \lambda_1 \frac{\exp(\lambda_2' X_{\lambda j})}{1 + \exp(\lambda_2' X_{\lambda j})}, \tag{19}$$

where $X_{\lambda j} \in \mathbb{R}^{K_\lambda}$ represents the observed characteristics of the app that affect the marginal cost, and $\lambda \coloneqq (\lambda_1, \lambda_2)$ is the parameter that determines the value of λ_j. We assume that ϵ_j is an i.i.d. positive random variable drawn from distribution G_ϵ with parameters ι.

Key identification assumption We imposed the following key identification assumption for our model: *no direct marginal cost of showing an advertisement on their app exists other than the loss from a decrease in demand attributable to the inconvenience caused to consumers by the advertisement.* We note that serving more consumers and greater usage incurs a marginal cost. Revenue is lost from decreasing consumer demand attributable to an increase in advertising intensity. What is assumed to be zero here is the *direct* marginal cost regarding increasing advertising intensity a_j.

In a standard merger analysis, we estimate the marginal cost from a firm's pricing decisions. However, in this paper, we estimate advertising intensity, the effective price, from the optimality condition assuming that no marginal cost exists that is specific to the decision. This assumption seems to be valid because connecting to ad networks and showing advertisement distributed through networks is almost automatic. The existing literature of ad-sponsored media focused on the media such as newspapers and cable TV. In their models, the costs of printing and producing advertisements, and acquiring new sponsors are included as direct marginal cost parameters of advertisements. They are not relevant in the context of mobile app advertisements.

Han et al. (2016) used a dummy to show advertisements as one of the product characteristics of a mobile app. We use the same information, but in a different way. We use the advertising dummy as a partial observation of advertising intensity and match the dummy with elicited advertising intensity, as discussed in further detail in the estimation section. Remark that the identification comes from the assumption of no direct marginal cost of advertising and the advertisement dummy is used only to further discipline the estimates.

Download price and advertising intensity decisions The decision problem for app developer d is written as:

$$\max_{\{(a_j, F_j)\}_{j \in \mathcal{J}_d}} \Pi_d(a, F, X, \xi; \theta) \tag{20}$$

$$\text{s. t.} \quad a_j \geq 0, \quad j \in \mathcal{J}_d, \tag{21}$$

$$F_j \geq 0, \quad j \in \mathcal{J}_d. \tag{22}$$

The first-order conditions for this problem are:

$$\frac{\partial \Pi_d}{\partial F_j} := (1-\rho)s_j + \sum_{k \in \mathcal{J}_d} \frac{\partial s_k}{\partial F_j}[(1-\rho)(F_k + e_k) + q_k(a_k r - \lambda_k)] \leq 0, \tag{23}$$

with equality if $F_j > 0$ for each $j \in \mathcal{J}_d$, and:

$$\begin{aligned} \frac{\partial \Pi_d}{\partial a_j} &:= s_j q_j r + s_j \frac{\partial q_j}{\partial a_j}(a_j r - \lambda_j) + s_j(1-\rho)\frac{\partial e_j}{\partial a_j} \\ &+ \sum_{k \in \mathcal{J}_d} \frac{\partial s_k}{\partial a_j}\{(1-\rho)(F_k + e_k) + q_k(a_k r - \lambda_k)\} \\ &\leq 0, \end{aligned} \tag{24}$$

with equality if $a_j > 0$ for each $j \in \mathcal{J}_d$, where the component derivatives are:

$$\begin{aligned} \frac{\partial s_k}{\partial y} &:= \frac{\partial \tilde{s}_k(a, F, w, X, \xi; \theta)}{\partial y} \text{ for } y \in \{F_j, a_j\}, \\ \frac{\partial q_k}{\partial a_j} &:= \frac{\partial \tilde{q}(a_j, w, X_j, \xi_j; \theta)}{\partial a_j}, \\ \frac{\partial e_k}{\partial a_j} &:= \frac{\partial \tilde{e}(a_j, w, X_j, \xi_j; \theta)}{\partial a_j}. \end{aligned}$$

A Bertrand-Nash equilibrium of the pricing game is a profile of pairs of advertising intensity and download prices $(a_j, F_j)_{j \in \mathcal{J}}$ that satisfies the system of equations (23) and (24).

5.4 Computing Equilibrium

The equilibrium conditions derived above is used to estimate the model. However, computing the equilibria in the original model is not easy because there are potentially many corner solutions in prices and advertising intensities. To avoid dealing with many corner solutions directly, we transform the original model of competition in advertising intensities and prices into the model of *competition in utility* (Armstrong and Vickers, 2001). We show the overview of the analysis below, and the detail can be found in Appendix A.

Suppose that each developer is constrained to achieve the mean utility δ_j for app j. Then, the developer chooses the optimal pair (a_j, F_j) of advertising intensity and download price to achieve δ_j. With this pair, the developer earns per-consumer profit $\bar{\pi}_j(\delta_j)$ by offering mean utility δ_j. Note that $\delta_j \in (-\infty, \delta_j^0]$, where δ_j^0 is the maximum mean utility that app j can provide, which is achieved when app j has zero download prices and advertising intensities. Then, given a profile of mean utilities $\delta = (\delta_j)_{j\in\mathcal{J}}$, and the profit earned by app j is given by $s_j(\delta) \times \pi_j(\delta_j)$. Therefore, each developer d's profit can be written in the following manner:

$$\bar{\Pi}_d = \sum_{j\in\mathcal{J}_d} s_j(\delta)\bar{\pi}_j(\delta_j),$$

which has the partial derivative

$$\frac{\partial\bar{\Pi}_d}{\partial\delta_k} = s_k(\delta)\bar{\pi}'_k(\delta_k) + \sum_{j\in\mathcal{J}_d} \frac{\partial s_j(\delta)}{\partial\delta_k}\bar{\pi}_j(\delta_j).$$

Thus, solving the first-order condition

$$\frac{\partial\bar{\Pi}_d}{\partial\delta_k} \le 0$$

with equality if $\delta_j < \delta_j^0$ for $j \in \mathcal{J}_d$ and $d \in \mathcal{D}$ gives the equilibrium mean utility δ_j^* for $j \in \mathcal{J}$. Finally, computing the optimal pair (a_j, F_j) of advertising intensity and download price of app j that achieves mean utility δ_j, we can separate the equilibrium advertising intensities and download prices.

Tables 13-15 of Appendix C illustrate how the equilibrium prices and advertising intensities depend on the underlying model parameters.

6 Estimation

6.1 Moment Conditions for Consumer Choice with Advertising Elicitation

We fix the parameters and data and first solve the equilibrium conditions for unobserved fixed effects ξ_{ej}, ξ_{uj}, and ξ_{dj}. To solve for ξ_{uj}, we elicit the advertising intensity a_j that is implied from the parameters and the data. Then, we define a generalized method-of-moments estimator that exploits the moments regarding these unobserved fixed effects. Let $\theta := (\theta'_d, \theta'_u, \lambda', \iota')'$, where

$\theta_d := (\alpha_y, \eta, \beta_d', \mathrm{vec}(\Sigma)')'$ is a set of parameters related to the download-related moment condition and $\theta_u := (\alpha_a', \beta_u')$ is the set of parameters. Let $\theta_0 := (\theta_{d0}', \theta_{u0}', \lambda_0', \iota_0')'$ denote true parameters.

Solving for ξ_e By arranging the first-order condition for a consumer regarding her in-app purchase decision (11), we can have:

$$\xi_{ej} = 4\alpha_y^2 \frac{e_j}{q_j}. \tag{25}$$

Let $\xi_{ej}(\theta_d)$ be the implied value of ξ_{ej} for $j \in \mathcal{J}$.

Solving for ξ_d We solve for the value of ξ_d from the following equation:

$$s_j = \int_{\mathbb{R}^{K_d}} \frac{\exp\left[\delta_j(a_j, F_j, w, X_j, \xi_j; \theta) + \nu_i' \Sigma X_{dj}\right]}{1 + \sum_k \exp\left[\delta_k(a_k, F_k, w, X_k, \xi_k; \theta) + \nu_i' \Sigma X_{dk}\right]} dG_{\nu_i}(\nu_i). \tag{26}$$

Although this equation involves ξ_{ej} and ξ_{uj} in general, we can solve for the values of ξ_{dj} as a function of observable variables and parameters. We do so by using the following equation:

$$\begin{aligned} \delta_j(a_j, F_j, w, X_j, \xi_j; \theta) &= S(a_j, w, X_j, \xi_j; \theta) + \beta_d' X_{dj} - \alpha_y F_j + \xi_{dj} \\ &= \frac{\eta_j}{2} q_j^2 + \beta_d' X_{dj} - \alpha_y F_j + \xi_{dj}. \end{aligned} \tag{27}$$

By inserting equation (27) into equation (26), we can express the share equation in terms of the parameters, observables, and values of ξ_{dj}. Then, we compute the implied value of ξ_d through a BLP-type inversion (Berry et al., 1995). Let $\xi_d(\theta_d)$ denote the implied values because the equation only depends on θ_d, given that the dependence of S on a_j, w, X_{uj}, ξ_{uj} works only through q_j and e_j in equation (27). This dependence results from the functional-form assumption in (12), a trick that allows us to separate the elicitation of ξ_d and ξ_u and substantially facilitates computation.

Additionally, note that this argument works because we did not allow random-coefficients for usage-related indirect utility. If the coefficients of X_{uj} were stochastic across consumers, then q_j in equation (27) would have been stochastic across consumers and indexed as q_{ij}. If we had consumer-level usage data, we could estimate a distribution of q_{ij} and integrate q_{ij} out from equation (26) under the condition that the conditional distribution of q_{ij} on that app j is chosen is the same as its unconditional distribution. The latter condition holds if the random coefficients on X_{dj} and X_{uj} are independent and the random coefficients on X_{uj} are realized after the consumer actually downloads the app.

Because we do not have consumer-level data, we cannot follow this approach. Then, allowing for random coefficients on X_{uj} requires us to solve the distribution of q_j under candidate parameters to evaluate an objective function of an estimator. This requirement significantly complicates the computational task. We stress that these restrictions on unobserved heterogeneity and functional form are utilized primarily to facilitate computation but not for identification.

Solving for a Next, we elicit advertising intensity $\{a_j\}_{j\in\mathcal{J}}$. To elicit the advertising intensity $\{a_j\}_{j\in\mathcal{J}}$, we utilize the first-order conditions for advertising intensity:

$$s_j q_j r + s_j \frac{\partial q_j}{\partial a_j}(a_j r - \lambda_j) + s_j(1-\rho)\frac{\partial e_j}{\partial a_j} + \sum_{k\in\mathcal{J}_d} \frac{\partial s_k}{\partial a_j}\{(1-\rho)(F_k + e_k) - \lambda_k + q_k a_k r\} \le 0, \tag{28}$$

where

$$\frac{\partial q_j}{\partial a_j} = -\frac{\alpha_{aj}}{\eta_j}, \frac{\partial e_j}{\partial a_j} = -\frac{\alpha_{aj} q_j \xi_{ej}}{4\alpha_y^2 \eta_j}, \tag{29}$$

and

$$\frac{\partial s_k}{\partial a_j} = \begin{cases} -\alpha_{aj} q_j \int_{\mathbb{R}^{K_d}} \frac{\exp(\delta_k + \nu_i' \Sigma X_{dk'})}{\left[1+\sum_{k'}\exp(\delta_k + \nu_i'\Sigma X_{dk'})\right]} \left[1 - \frac{\exp(\delta_k + \nu_i'\Sigma X_{dk'})}{\left[1+\sum_{k'}\exp(\delta_k+\nu_i'\Sigma X_{dk'})\right]}\right] dG_{\nu_i}(\nu_i) & \text{for } k = j \\ -\alpha_{aj} q_j \int_{\mathbb{R}^{K_d}} \frac{\exp(\delta_j + \nu_i'\Sigma X_{dj})\exp(\delta_k+\nu_i'\Sigma X_{dk'})}{\left[1+\sum_{k'}\exp(\delta_k+\nu_i'\Sigma X_{dk'})\right]^2} dG_{\nu_i}(\nu_i) & \text{for } k \neq j \end{cases} \tag{30}$$

for all $j \in \mathcal{J}$. Given the data $(s_j, q_j, p_j)_{j\in\mathcal{J}}$ and computed values of $(\xi_{ej}(\theta), \xi_{dj}(\theta))_{j\in\mathcal{J}}$, we can compute the simulated value of $\partial s_k/\partial a_j$. Solving this system of equations and inequalities is complicated in general. However, with our model setting, we can compute the values of advertising intensity $(a_j)_{j\in\mathcal{J}}$ that satisfy the system of first-order conditions (28) by solving quadratic programming. The details of the procedure and the proof are in the Appendix B. Let $a(\theta)$ denote the elicited advertising intensity.

Solving for ξ_u By plugging $\xi_e(\theta_d)$ and $a(\theta)$ into the first-order condition for usage (12), we obtained the implied value of ξ_{uj}:

$$\xi_{uj} = \eta_j q_j + \alpha_{aj} a_j(\theta) + \alpha_y w - \beta_u' X_{uj} - \frac{\xi_{ej}(\theta_d)}{2\alpha_y}. \tag{31}$$

Let $\xi_{uj}(\theta)$ be the implied value of ξ_{uj} for each $j \in \mathcal{J}$.

Moment conditions Let $Z_{uj} \in \mathbb{R}^{L_u}$ and $Z_{dj} \in \mathbb{R}^{L_d}$ are sets of instrumental variables for app j that satisfy:

$$\mathbb{E}[\xi_{uj}(\theta_0)|Z_{uj}] = \mathbb{E}\left[\xi_{dj}(\theta_{d0})|Z_{dj}\right] = 0, \tag{32}$$

which implies:

$$\mathbb{E}[\xi_{uj}(\theta_0)Z_{uj}] = \mathbb{E}\left[\xi_{dj}(\theta_{d0})Z_{dj}\right] = 0, \tag{33}$$

for any $j \in \mathcal{J}$.

Objective Function Let $t \in \mathcal{T} \coloneqq \{1, \dots, T\}$ be the set of indices of markets, $\mathcal{J}_t$ the set of apps in market t, and $(X_{ujt}, X_{djt}, Z_{ujt}, Z_{djt}, e_{jt}, F_{jt}, q_{jt}, s_{tj})$ the list of variables regarding app j in market t. Let $N \coloneqq \sum_{t=1}^{T} J_t$. Let:

$$\xi_{ut}(\theta) \coloneqq [\xi_{u1t}(\theta), \cdots, \xi_{uJ_tt}(\theta)]', \xi_{dt}(\theta_d) \coloneqq [\xi_{d1t}(\theta_d), \cdots, \xi_{dJ_tt}]', \tag{34}$$

be the J_t-dimensional vector of product-specific unobserved heterogeneity in market t and:

$$\xi_u(\theta) \coloneqq [\xi_{u1}(\theta)', \cdots, \xi_{uT}(\theta)']', \xi_d(\theta_d) \coloneqq [\xi_{d1}(\theta_d)', \cdots, \xi_{dT}(\theta_d)']', \tag{35}$$

be the $\sum_{t=1}^{T} J_t$-dimensional vector of product-market-specific unobserved heterogeneity.

Similarly, for instrumental variables and product characteristics, let:

$$Z_{\iota t} \coloneqq \begin{pmatrix} Z'_{\iota 1t} \\ \vdots \\ Z'_{\iota J_t t} \end{pmatrix}, X_{\iota t} \coloneqq \begin{pmatrix} X'_{\iota 1t} \\ \vdots \\ X'_{\iota J_t t} \end{pmatrix}, \iota \in \{u, d\}, \tag{36}$$

be the $J_t \times L_\iota$-dimensional matrix of instrumental variables and the $J_t \times K_\iota$-dimensional matrix of instrumental variables in market t and:

$$Z_\iota \coloneqq \begin{pmatrix} Z_{\iota 1} \\ \vdots \\ Z_{\iota T} \end{pmatrix}, X_\iota \coloneqq \begin{pmatrix} X_{\iota 1} \\ \vdots \\ X_{\iota T} \end{pmatrix}, \iota \in \{u, d\}, \tag{37}$$

be $\sum_{t=1}^{T} J_t \times L_\iota$-dimensional matrix of instrumental variables and $\sum_{t=1}^{T} J_t \times K_\iota$-dimensional matrix of product characteristics.

Finally, let:

$$g^D(\theta) \coloneqq \frac{1}{\sum_{t \in \mathcal{T}} J_t} \begin{pmatrix} Z'_u \xi_u(\theta) \\ Z'_d \xi_d(\theta_d) \end{pmatrix} \tag{38}$$

be the $L_u + L_d$-dimensional moments related to the demand with elicited advertising.

6.2 Moment Conditions for App Developer's Choice

Optimality conditions for download price For each $j \in \mathcal{J}_t$ in each $t \in \mathcal{T}$, the following equality holds:

$$\epsilon^P_{jt}(\theta) \coloneqq \frac{\partial \Pi_{jt}(\theta)}{\partial F_{jt}} 1\{F_{jt} > 0\} + \max\left\{\frac{\partial \Pi_{jt}(\theta)}{\partial F_{jt}}, 0\right\} 1\{F_{jt} = 0\} = 0. \tag{39}$$

We construct a corresponding moment such as:

$$g^P(\theta) \coloneqq \frac{1}{\sum_{t \in \mathcal{T}} J_t} Z'_u \epsilon^P(\theta), \tag{40}$$

where $\epsilon^P(\theta) \coloneqq [\epsilon^P_{11}(\theta), \cdots, \epsilon^P_{J_T T}(\theta)]'$.

Advertising matching Although advertising intensity is not observed, we observe whether or not an app shows advertisements or not. Given the true parameter, we expect that approximately the following equation holds:

$$\epsilon_{jt}^{A}(\theta) \coloneqq [A_{jt} - 1\{a_{jt}(\theta) > 0\}] = 0, \tag{41}$$

where A_{jt} takes the value of 1 if app j shows advertisements in market t and takes the value of 0 otherwise. We construct a corresponding moment such as:

$$g^{A}(\theta) \coloneqq \frac{1}{\sum_{t \in \mathcal{T}} J_t} Z_u' \epsilon^{A}(\theta), \tag{42}$$

where $\epsilon^{A}(\theta) \coloneqq [\epsilon_{11}^{A}(\theta), \cdots, \epsilon_{J_T T}^{A}(\theta)]'$.

6.3 Generalized Method-of-Moments Estimator

Definition We define a generalized method-of moments (GMM) estimator $\hat{\theta}$ by:

$$\hat{\theta} \in \mathrm{argmin}_{\theta \in \Theta} g(\theta)' \Phi^{-1} g(\theta), \tag{43}$$

where $g(\theta) \coloneqq [g^{D\prime}(\theta), g^{P\prime}(\theta), g^{A\prime}(\theta)]'$ and Φ is a positive-definite weighting matrix.

We start with an initial weighting matrix blkdiag$[Z_u'Z_u, Z_d'Z_d, Z_u'Z_u, Z_u'Z_u]$. Then, in the second step, we use the sample covariance of the moments evaluated at the initial estimates. We first estimate a model without random coefficients nor heterogeneity in $\alpha_{aj}, \eta_j, \lambda_j$. Then, we estimate the model by first adding the random coefficients and, second, the heterogeneity, using the previous estimates as the initial values. Because of the computational burden, we estimate the parameters by randomly sub-sampling 20 markets (weeks) from the entire data. This sub-sampling provides approximately 10,000 observations at the app-market level. We obtain the confidence intervals by repeatedly estimating the parameters using a randomly selected list of sub-samples. We minimize the objective function by using an adaptive barrier algorithm implemented by the *constrOptim* function in R to impose non-negativity constraints on the parameters except for the parameters governing the heterogeneity of $\alpha_{aj}, \eta_j, \lambda_j$.

Choice of instrumental variables Z_{djt} includes 1, X_{djt}, X_{djt}^2 and differentiation instrumental variables (Gandhi and Houde, 2019). Specifically, for each app, for $\iota \in \{d, u\}$, compute the difference from the other apps in the product characteristics space:

$$d_{\iota jkt} \coloneqq \sqrt{\sum_{l=1}^{K_\iota} (X_{\iota jlt} - X_{\iota klt})^2}, \tag{44}$$

and compute the average and variance of the differences within the same app developer and outside the app developer:

$$\frac{1}{J_{d(j)t}} \sum_{k \in \mathcal{J}_{d(j)t}} d_{\iota jkt}, \frac{1}{J_t - J_{d(j)t}} \sum_{k \in \mathcal{J}_t \setminus \mathcal{J}_{d(j)t}} d_{\iota jkt}, \frac{1}{J_t - 1} \sum_{k \in \mathcal{J}_t} \left(d_{\iota jkt} - \frac{1}{J_t} \sum_{k \in \mathcal{J}_t} d_{\iota jkt} \right)^2. \tag{45}$$

Moreover, we include hourly wage and advertising price as market-level demand and cost shifters. Z_u includes the corresponding variables except for X_{ujt}^2, because there is no random-coefficient in the usage-related utility.

Linear and non-linear parameters We can further accelerate the computation by distinguishing between *linear* and *non-linear* parameters; linear parameters can be explicitly derived by minimizing the objective function in equations (42), given the rest of the parameters. Specifically, the linear parameters in θ_d and θ_u in our framework are $\theta_{d1} \coloneqq \beta_d$ and $\beta_{u1} \coloneqq \beta_u$, and the non-linear parameters in θ_d and θ_u in our framework are $\theta_{d2} \coloneqq [\alpha_y, \eta_j, \mathrm{vec}(\Sigma)']'$ and $\theta_{u2} \coloneqq \alpha_{aj}$. Given $\theta_2 \coloneqq (\theta_{2d}', \theta_{2u}', \lambda_j)'$, the residuals in the demand-related moment condition are written as:

$$\begin{aligned} \xi_d(\theta) &= y_d(\theta) - X_d \beta_d, \\ \xi_u(\theta) &= y_u(\theta) - X_u \beta_u, \end{aligned} \tag{46}$$

with:

$$\begin{aligned} y_d(\theta) &\coloneqq \delta(\theta) - \frac{\eta_j}{2} q^2 + \alpha_y F, \\ y_u(\theta) &\coloneqq \eta_j q + \alpha_{aj} a(\theta) + \alpha_y w - \frac{\xi_e(\theta_d)}{4\alpha_y}, \end{aligned} \tag{47}$$

where $\delta(\theta), q, e, F, a(\theta), w$, and $\xi_e(\theta_d)$ are vectors in which corresponding elements are stacked first by apps and then by markets.

Both $y_d(\theta)$ and $y_u(\theta)$ depend on θ_1 through $\delta(\theta)$, $a(\theta)$, and $\xi_e(\theta_d)$. However, in our specification of the model, $\delta(\theta)$, $a(\theta)$, and $\xi_e(\theta_d)$ are independent of θ_1 conditional on the observables $(s, q, e, X_d) = \{(s_j, q_j, e_j, X_{dj})_{j \in \mathcal{J}_t}\}_{t=1,\dots,T}$ and non-linear parameters θ_2 for the following reasons. First, equation (26) implies that $\delta(\theta)$ can be computed only using (s, X_d), denoted by $\hat{\delta}(\theta_2, s, X_d)$. Similarly, equation (25) implies that $\xi_e(\theta_d)$ can be computed only using q, e, and α_y, denoted by $\hat{\xi}_d(\theta_{2d}, q, e)$. Finally, equations (28), (29), and (30) jointly imply that $a(\theta)$ can be computed only using variables (s, q, e, X_d), $\xi_e(\theta_d)$, $\delta(\theta)$ and non-linear parameters θ_2, denoted by $\hat{a}(\theta_2, s, q, e, X_d)$. Therefore, $y_d(\theta)$ can be evaluated as $\hat{y}_d(\theta_2, s, q, e, F, X_d)$ and $y_u(\theta)$ can be evaluated as $\hat{y}_u(\theta_2, s, q, e, w, X_d)$. Similarly, $g^P(\theta)$ can be evaluated by (s, q, e, F, r), λ_j, and $\hat{a}(\theta_2, s, q, e, X_d)$, denoted by $\widehat{g}^P(\theta_2, s, q, e, F, r, X_d)$. Finally, $g^A(\theta)$ is evaluated only with $\hat{a}(\theta_2, s, q, e, X_d)$, denoted by $\widehat{g}^A(\theta_2, s, q, e, X_d)$.

As a result, given observables and fixed non-linear parameters, we can ignore the impact of linear parameters on $y_d(\theta)$ and $y_u(\theta)$, and $g^P(\theta)$ and $g^A(\theta)$, which enables us to explicitly derive the estimates of linear parameters conditional on non-linear parameters as follows:

$$\hat{\theta}_1(\theta_2) = \left(X'Z\Phi^{D-1}Z'X\right)^{-1} X'Z\Phi^{D-1}Z'\hat{y}(\theta_2, s, q, e, F, r, w, X_d), \tag{48}$$

where $\hat{y}(\theta_2, s, q, e, F, r, w, X_d) := [\hat{y}_d(\theta_2, s, q, e, F, X_d)', \hat{y}_u(\theta_2, s, q, e, w, X_d)']'$, $Z := \text{blkdiag}(Z_u, Z_d)$, and $X := \text{blkdiag}(X_u, X_d)$. Φ^D is a submatrix of Φ corresponding to demand-releated moments.

6.4 Incorporating Semantic Vectors of Product Description

Semantic vectors without random coefficients The product description of an app is represented by a semantic vector, which we denote by $W_j \in \mathbb{R}^P$. Product attributes encoded in W_j surely affect consumer demand; however, which of them will do so is not *a priori* clear. Therefore, we allow data to indicate the dimension of W_j that is particularly relevant. The interpretation of X_{dj}, X_{uj}, and W_j is that X_{dj} and X_{uj} are variables that certainly affect utility, and W_j represents variables with uncertain influence. First, we assume that no consumer-level heterogeneity exists regarding the coefficients for W_j. If this is the case, W_j should be part of the unobserved heterogeneity ξ_{dj} and ξ_{uj} in the previous model:

$$\begin{aligned} \xi_{dj} &= \gamma_d' W_j + \Delta\xi_{dj}, \\ \xi_{uj} &= \gamma_u' W_j + \Delta\xi_{uj}, \end{aligned} \tag{49}$$

where $\Delta\xi_{dj}$ and $\Delta\xi_{uj}$ represent residual unobserved heterogeneity that is not correlated with W_j. Using fitted values $\hat{\xi}_{dj}(\hat{\theta})$ and $\hat{\xi}_{uj}(\hat{\theta})$ based on the GMM estimator $\hat{\theta}$, we can estimate γ_d and γ_u by regressing the fitted values on W_j. Because the semantic space of the in-text product description is high-dimensional, the ordinary least square estimates may over-fit to the training data. Therefore, to improve generalization performance, we estimate using rigorous post-LASSO estimators, in which the penalty loading of each variable is calculated depending on the variables and allowing for heteroskedasticity (Belloni and Chernozhukov, 2013).

Semantic vectors with random coefficients Next, we consider a model in which consumers have heterogeneous tastes for the features represented by the semantic vector of a mobile app in an unobserved manner. Then, the model is no different from the previous model in which X_{dj} and X_{uj} are replaced with $(X_{dj}', W_j)'$ and $(X_{uj}', W_j')'$, respectively. However, estimating this model is practically not possible, because too many parameters need to be estimated. Therefore, we adopt a short cut and we only use dimensions of W_j that are found to be relevant in the rigorous post-LASSO estimator assuming that no random coefficients exist as for the semantic vectors.

6.5 Eliciting download-related residual terms

To conduct counterfactual simulations, we elicit residual terms in the pricing first-order-condition implied by the estimated parameters. To simultaneously elicit the download-related residual terms

$(\epsilon_j)_{j\in\mathcal{J}}$ and advertising intensities $(a_j)_{j\in\mathcal{J}}$, we use the equilibrium conditions for both prices and advertisements.

First, we use the equilibrium condition for prices (23) to elicit the "effective marginal cost" $\widetilde{mc}_j := \epsilon_j - q_j(a_j - \lambda_j)$. To do this we rewrite the equilibrium condition (23) as

$$(1-\rho)s_j + \sum_{k\in\mathcal{J}_d} \frac{\partial s_k}{\partial F_j}\left[(1-\rho)(F_k + e_k) - \widetilde{mc}_j\right] \begin{cases} = 0 & \text{if } F_j > 0 \\ < 0 & \text{if } F_j = 0. \end{cases}$$

Because the first-order conditions for free apps are given in the form of inequalities, we cannot derive the exact values of $\widetilde{mc}_j$ by using the system of first-order conditions alone. To elicit the exact value of marginal costs that is consistent with the equilibrium conditions, we adopt the following procedure: fixing a positive constant $\chi > 0$, for apps with positive download prices, we use the equality first-order condition (23), for apps with zero download prices, we assume that if the marginal costs of free apps were greater by χ, the first-order condition (23) would be satisfied with equality.

Then, the elicited value of the effective marginal cost $\widetilde{mc}_j$ satisfies

$$(1-\rho)s_j + \sum_{k\in\mathcal{J}_d} \frac{\partial s_k}{\partial F_j}\left[(1-\rho)(F_k + e_k) - \widetilde{mc}_j - 1\{F_k = 0\}\chi\right] = 0 \tag{50}$$

for $j \in \mathcal{J}$. Solving this system of equations yields the elicited values of effective marginal costs $\widetilde{mc}_j$, $j \in \mathcal{J}$.

Using the elicited effective marginal costs, we find the advertising intensities that are consistent with the equilibrium condition of advertisements (24). Using equation (50), equation (24) can be rewritten as below:

$$s_j\left[q_j r + \frac{\partial q_j}{\partial a_j}(a_j r - \lambda_j) + (1-\rho)\frac{\partial e_j}{\partial a_j} - (1-\rho)\frac{q_j \alpha_{aj}}{\alpha_y}\right] - \frac{q_j \alpha_{aj}}{\alpha_y} + \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d} \frac{\partial s_k}{\partial F_j} 1\{F_k = 0\}\chi \le 0, \tag{51}$$

which holds as equality if $a_j > 0$. We can solve this equation for each a_j because these equations are independent of each other for given parameter values and observed data (s, q, e, X_d). Finally, using the equation $\widetilde{mc}_j = \epsilon_j - q_j(a_j - \lambda_j)$, we find the value of download-related marginal cost ϵ_j.

In calculating download-related residual terms and advertising intensities, we set $\chi = 50$, which amounts to assuming that if all the download-related marginal costs of free apps were higher by more than JPY 50, those apps would have positive download prices.

7 Estimation Result

Estimates of non-linear and linear parameters Tables 4 and 5 provide a summary of the estimation results of the non-linear parameters. In the high-dimensional heterogeneous mixed-logit specification, the estimate of α_y is 0.00409 and 0.00092 for applications and games, respectively.

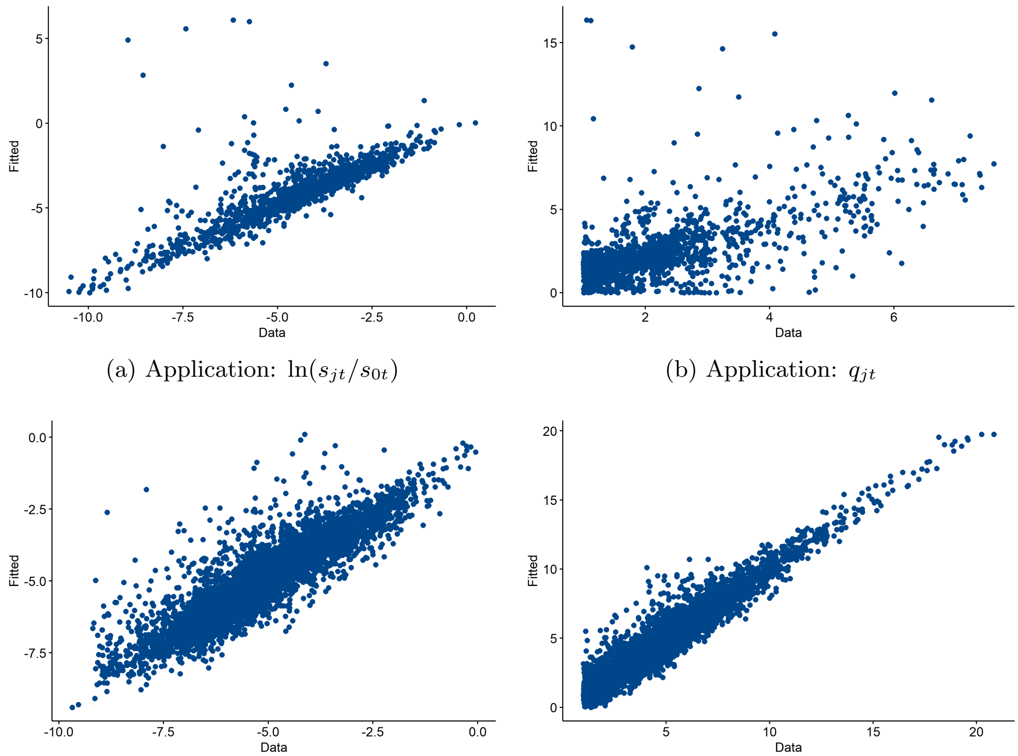
Thus, the standard deviation in the download preference shock is huge and larger for games. The estimate of α_{a0} is 0.114 and 0.104, respectively. The parameter estimates indicate that on average the disutility of watching one unit of advertisements is JPY 14 for applications and JPY 56.4 for games. Because the average advertising price is JPY 425.8, these amount to 3.1% and 12.4% of the app's advertising revenue. These numbers are smaller than the estimate of Facebook presented in the study of Benzell and Collis (2020). Based on a survey, they estimated that the disutility of advertising is approximately 20% of Facebook's advertising revenue. There are several explanations for this difference. First, it may be merely an estimation error due to the difference in the approach: survey based or revealed preference based. Second, it may be because of the difference in the nature of the apps. Our estimates suggest that the disutility is higher for games, in which consumer spend more times than they spend on applications. Facebook is classified into applications in our analysis, but is closer to games in the usage pattern. Third, it may be because we use a nominal unit of advertisement as a basis of calculation. Advertisements are often transacted at the nominal number of impressions, but it is not certain whether a consumer actually pays attention to and is "impressed" by an ad shown. Thus, the disutility of watching advertisement can be estimated to be less based on the revealed preference.

The estimation results of the linear parameters and the standard deviation of random coefficients are in the online appendix.

Model fit To obtain a fitted value, we need to take expectation with respect to the underlying shocks ξ_{djt}, ξ_{ujt}, and ϵ_{ijt}. Taking expectation with respect to idiosyncratic shocks ϵ_{ijt} is straightforward, because they are assumed to be an i.i.d. type-I extreme random variable. However, we need a numerical integration for ξ_{djt} and ξ_{ujt}. Specifically, we first regress $\hat{\xi}_{djt}(\hat{\theta})$ and $\hat{\xi}_{ujt}(\hat{\theta})$ on endogenous variables a_{jt}, p_{jt}, q_{jt}, and $j-$ and $t-$dummies using a gradient boosting decision tree, because they can be the unobserved heterogeneity can be correlated with those variables, and obtain residuals. For each observation, we draw samples of ξ_{djt} and ξ_{ujt} by re-sampling the residuals and adding the predicted value from the decision tree model. For each draw of ξ_{djt} and ξ_{ujt}, we solve consumer usage and download decisions and take an average across the samples to obtain the fitted value of download share and usage time.

To examine the fit, we drop an app-week that is used less than an hour per week because the usage data in this range is noisy and it requires a large negative value of ξ_{ujt} to fit the model to the data. If we replace ξ_{ujt} for such an app-week with a resampled one, the resulting indirect utility of usage often becomes extraordinarily large and absorbs almost the entire download share. In the counterfactual simulations, these app-weeks are appropriate as long as we use the implied ξ_{ujt}. We do not claim that our model can explain the noisy variation in using the app for less than an hour per week. Rather, we focus on the model performance for app-weeks whose usage is more than an hour per week.

Figure 4 and 5 display the scatter plot between data and fitted values for share $\ln(s_{jt}/s_{0t})$ and usage q_{jt}. Table 6 shows that the R-squared of regressing the in-sample data on fitted values is 0.95



(a) Application: $\ln(s_{jt}/s_{0t})$

(b) Application: q_{jt}

(c) Game: $\ln(s_{jt}/s_{0t})$

(d) Game: q_{jt}

Figure 4: Fit to download share and usage: In-sample

(0.98) for download and 0.78 (0.98) for usage for applications (games). The out-of-sample fit of the model is also good except for the download of non-game applications. The R-squared of regressing the out-of-sample data on fitted values is 0.16 (0.96) for download and 0.51 (0.95) for usage for applications (games). The out-of-sample fit of non-game applications is worse than others because the usage tends to be lower and noisier for non-game applications. Thus, although the model is highly stylized, it fits the download and usage data well.

Relevance of semantic vectors Figure 6 shows the estimates of γ_d and γ_u in descending order with the size of the absolute value of the estimates. Among the 300 dimensions, approximately 20-30 dimensions are picked by the rigorous post-Lasso method and the joint-significance test was rejected at the 1% level for both download and usage, which underscores the relevance of the semantic vectors in the heterogeneity in the marginal utility of download and usage.

8 Defining Relevant Markets

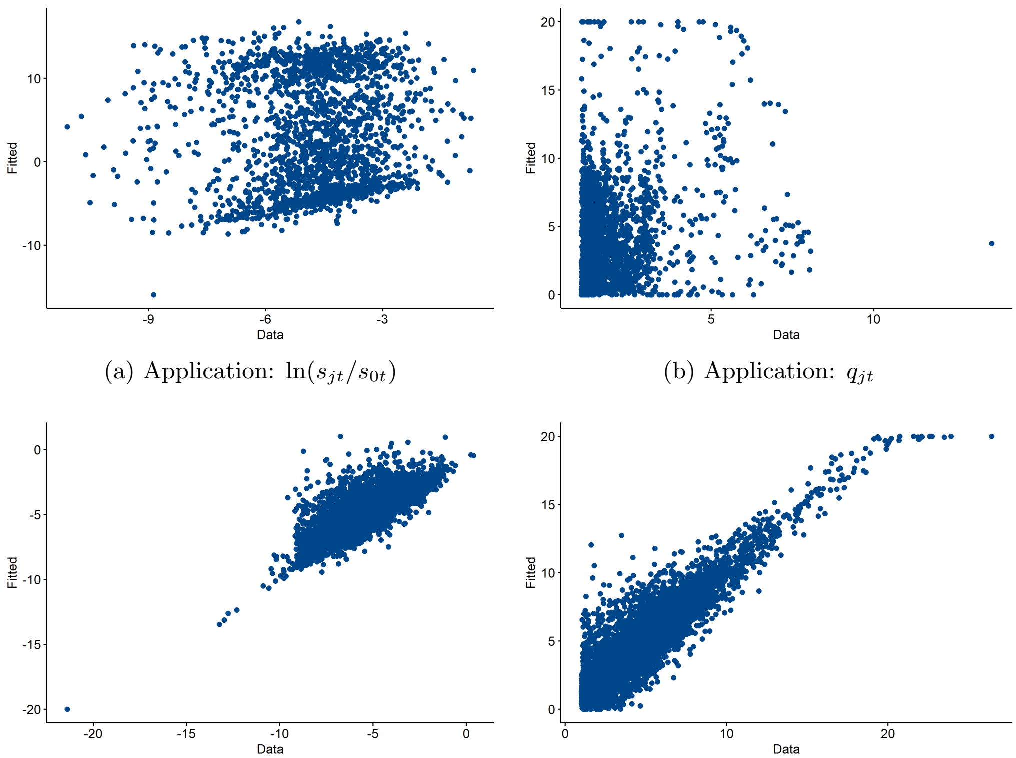
In the antitrust policy, the relevant market of a product typically needs to be defined to initiate the investigation of the case. The definition can be based on qualitative information such as the

Table 4: Estimation results for non-linear parameters: Applications

	Low-dimensional		**High-dimensional**
	Homogeneous	Heterogeneous	Heterogeneous
Utility per JPY (α_y)	0.00390	0.00390	0.00409
Disutility per advertising (α_{a0})	0.11444	0.11444	0.11451
Lifestyle		0.00000	0.00015
Video players		0.00000	0.00015
Entertainment		0.00000	0.00037
Comics		0.00000	0.00344
Music and audio		0.00000	0.01067
Photography		0.00000	0.00015
News and magazines		0.00175	0.00191
Tools		0.00000	0.00000
Social		0.00000	−0.00003
Maps and navigation		0.00000	0.00059
Degree of usage satiation (η)	0.11559	0.11559	0.11565
Lifestyle		0.01052	0.01067
Video players		0.00000	0.00101
Entertainment		0.00000	0.00015
Comics		0.00000	0.00015
Music and audio		0.00000	0.01023
Photography		0.00000	0.00015
News and magazines		0.00132	0.00147
Tools		0.00000	−0.00006
Social		0.00000	0.00015
Maps and navigation		0.00000	0.00015
Marginal cost (λ)	0.14028	0.14028	0.14029
Lifestyle		0.00000	0.00015
Video players		0.00000	0.00015
Entertainment		0.00000	0.00015
Comics		0.00000	0.00015
Music and audio		0.00000	0.00015
Photography		0.00000	0.01385
News and magazines		0.02805	0.04563
Tools		0.01425	0.01440
Social		0.00022	0.00038
Maps and navigation		0.00000	0.00015

Table 5: Estimation results for non-linear parameters: Games

	Low-dimensional		**High-dimensional**
	Homogeneous	Heterogeneous	Heterogeneous
Utility per JPY (α_y)	0.00091	0.00091	0.00092
Disutility per advertising (α_{a0})	0.10380	0.10380	0.10378
Simulation		0.00000	0.00083
Casual		0.00180	0.00262
Card		0.00000	0.00059
Role playing		0.00000	0.00089
Puzzle		0.00000	0.00082
Casino		0.00000	0.00104
Sports		0.00000	0.00243
Strategy		0.00000	−0.00064
Action		0.00000	0.00080
Degree of usage satiation (η)	0.10171	0.10171	0.10084
Simulation		0.00000	0.00082
Casual		0.01262	0.01344
Card		0.00000	0.00682
Role playing		0.00000	−0.00707
Puzzle		0.00000	−0.00019
Casino		0.00000	−0.00015
Sports		0.00000	0.00085
Strategy		0.00000	−0.00047
Action		0.00000	−0.00074
Marginal cost (λ)	0.14418	0.14418	0.14436
Simulation		0.00000	0.00079
Casual		0.00251	0.01236
Card		0.00000	0.00548
Role playing		0.00000	0.00013
Puzzle		0.00000	0.00079
Casino		0.00000	0.00077
Sports		0.00000	0.00264
Strategy		0.00000	0.00014
Action		0.00000	0.00076



(a) Application: $\ln(s_{jt}/s_{0t})$ (b) Application: q_{jt}

(c) Game: $\ln(s_{jt}/s_{0t})$ (d) Game: q_{jt}

Figure 5: Fit to download share and usage: Out-of-sample

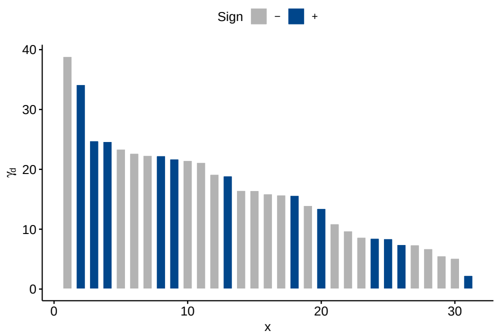
Table 6: Goodness-of-fits to download share and usage

(a) Application

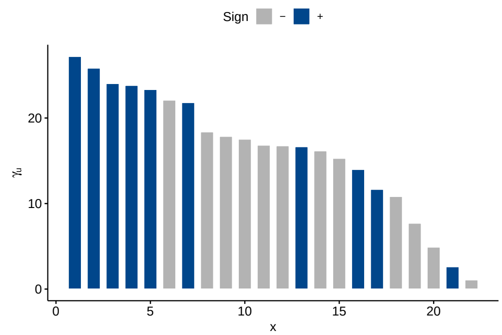
	In-sample		**Out-of-sample**	
	Download	Usage	Download	Usage
Residual standard deviation	1.0515	1.0953	2.3850	1.4908
Multiple R-squared	0.9490	0.7835	0.1613	0.5135
Adjusted R-squared	0.9490	0.7834	0.1608	0.5133

(b) Game

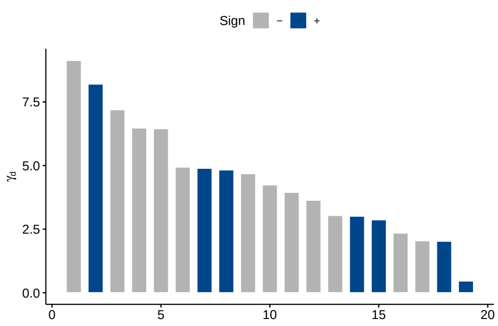
	In-sample		**Out-of-sample**	
	Download	Usage	Download	Usage
Residual standard deviation	0.6661	0.7566	1.0053	1.2286
Multiple R-squared	0.9849	0.9798	0.9682	0.9513
Adjusted R-squared	0.9849	0.9798	0.9682	0.9513



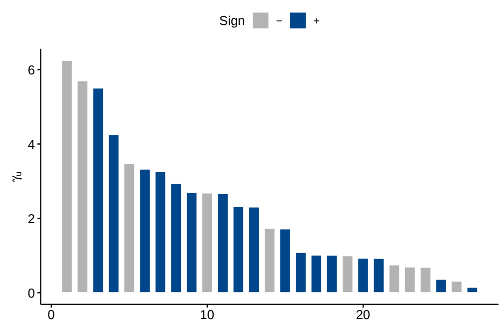
(a) Application: γ_d from high-dimensional model



(b) Application: γ_u from high-dimensional model



(c) Game: γ_d from high-dimensional model



(d) Game: γ_u from high-dimensional model

Figure 6: Estimation results

product category, such as game app, music app, and chat app in case of the App industry, or can be based on the quantitative analysis of price and quantity. SSNIP test is one of such methodology to define a relevant market. However, this test is not directly applicable to a free product because it checks whether the hypothetical monopolist owning the product can profitably increase the price when it owns other products. Without prices, whether the price increases cannot be determined. Even for a paid product, if the product generates revenue by showing advertisements, we need to consider the changes in advertising intensity to define a relevant market. Otherwise, the resulting market can be misleading.

Newman (2015) introduced the concept of *A Small, Non-transitory but Significant Increase in Cost (SSNIC)* test to resolve this problem. The SSNIC examines whether the cost to a consumer—not only the price—can be profitably increased. For the App economy, advertising intensity is the consumer's non-price cost that increases consumer's inconvenience while generates revenues for the producer. Thus, for free products, we can still define a relevant market, by focusing on advertising intensity. We call this version of the test a SSNIC test.

In contrast to the standard demand model, the model in this paper endogenizes advertising intensity, thus, allowing us to define a relevant market in the App economy. This section discusses how to define a relevant market using the model and the estimation results from the previous section. It also illustrates the misleading nature of the definition of a relevant market using the standard demand model that endogenizes advertising intensity.

8.1 Methods

Store category As an example of relevant market definition using the existing product category, we use the categories defined in Google Play. Google Play defines Apps and Games as the upper-level categories. Below the Apps and Games categories, 49 categories are defined, as noted in Section 3.2.

Small but significant and non-transitory increase in cost (SSNIC) tests We introduce a formal definition of SSNIC test that changes the cost to consumers using the competition-in-utility framework in Section 5.4. We adopt the formalization introduced by Ivaldi and Lorincz (2011) for the SSNIP test, which attempts to describe the European Commission guidelines (European Commission, 1997) and U.S. guidelines for 1992.

First, we define the notion of cost c_j of app j as the gap between its maximal mean utility δ_j^0 and its actual mean utility δ_j, that is, $c_j = \delta_j^0 - \delta_j$. For example, when an app j is ad-free and only charges download price F_j, the cost is given by $c_j = \alpha_y F_j$. Setting the cost c_j leads to the mean utility $\delta_j = \delta_j^0 - c_j$ and gives the per-consumer profit $\bar{\pi}_j(\delta_j^0 - c_j)$ in our competition-in-utility framework.

Let $c = \{c_j\}_{j \in \mathcal{J}}$ be the costs of the apps and $s(c) = \{s_j(c)\}_{j \in \mathcal{J}}$, $q(c) = \{q_j(c)\}_{j \in \mathcal{J}}$, and $\pi(c) = \{\pi_j(c)\}_{j \in \mathcal{J}}$ be the download shares, usages, and profits under c, respectively. Let c^* denote the benchmark equilibrium costs.

Then, the SSNIC relevant market of app j is formally defined as follows: Let $\mathcal{M} \subset \mathcal{J}$ and $j \in \mathcal{M}$. Let c_l^{SSNIC} be a cost equal to $(1+\kappa)c_l^*$ if $l \in \mathcal{M}$, and c_l^* otherwise, where $0 < \kappa \leq 0.1$. Then, $\mathcal{M}$ is the SSNIC relevant market of app j if and only if:

1. $\Delta\pi_{\mathcal{M}}^{SSNIC} > 0$, where

$$\Delta\pi_{\mathcal{M}}^{SSNIC} \equiv \left(\frac{\sum_{l\in\mathcal{M}}[\pi_l(c^{SSNIC}) - \pi_l(c^*)]}{\sum_{l\in\mathcal{M}} \pi_l(c^*)} \right); \tag{52}$$

2. for all $\mathcal{M}' \subset \mathcal{J}$ such that $j \in \mathcal{M}'$ and $\mathcal{M}'$ satisfies (1), $\#(\mathcal{M}) \leq \#(\mathcal{M}')$.

This definition of SSNIC test encompasses the notion of SSNIP test because when there is no advertising, a κ% increase in the costs is equivalent to a κ% increase in the prices. Furthermore, by summarizing the impacts of advertising intensities and prices into uni-dimensional costs, our definition of SSNIC test enables us to compare the apps with different monetization policies.

Order of testing price/cost increase For SSNIC tests, we sequentially add new products to the portfolio of the hypothetical monopolist. The profit change attributes to SSNIC depends on the order of adding products. This definition of a relevant market only refers to the minimal set of products that increases profits, but is silent about the procedure to determine the order to achieve the minimal set. In practice, the analyst often picks up an ad-hoc but intuitive criterion such as the degree of the cross-price elasticity to determine the order of adding products. Following this practice, we consider an order based on the size of cross price elasticity with the target app. Moreover, we also use a strategy based on the cosine similarity in the semantic vector space.

8.2 Comparing SSNIC-based relevant markets with the product category

Illustration with a social app Because of the confidentiality term of the data contract, we can only pick up apps based on the existing product category rankings, and must anonymize the app names. First, for illustration, we show the results for a top social networking app with the largest number of downloads in the category. We pick up a social networking app for illustration, because the merger and acquisition in this category is often under debate.

Figure 7 plots the change in profits of the hypothetical monopolist along the path of the SSNIC tests. In the plots, the x-axis indicates the index of added apps and the y-axis indicates the change in profits when the cost is increased by 5%. The number of apps included in the relevant market based on the elasticity-based order is 77. The number is close to the similarity-based order.

Relevant markets for top apps in each category Next, we select apps from each product category with the largest number of downloads. Then, we defined relevant markets by SSNIC using the elasticity-based order. Then, we calculated the share of the top app in the relevant market and the HHI of each relevant market.

The results are summarized in Table 7. In both cases of games and non-game applications, the number of apps that comprise a relevant market is larger than the number of apps in each category.

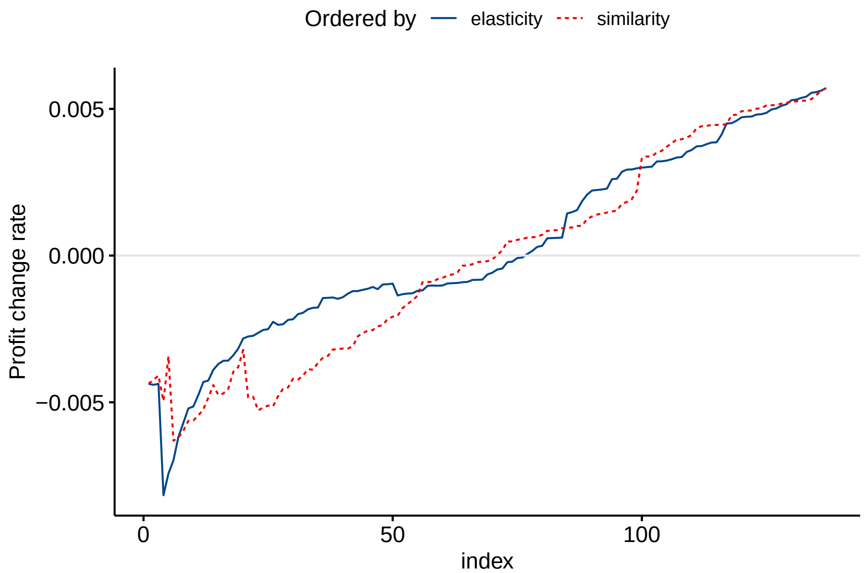


Figure 7: SSNIC test: Social top app

Accordingly, the market share of the top app and the concentration of the relevant market are lower under the SSNIC test than under the category-based market. For example, for the top comics app, the number of apps is 79 with a SSNIC test compared with 7 in the category. The share of the top app and the HHI of the relevant market are lower in the relevant market with a SSNIC test than in the category market. This result implies that the arbitrary definition of a relevant market can lead to a misleading outcome.

Table 8 shows the download share of apps within each category that are considered to form a relevant market for a target top app. For example, the first row of the upper panel indicates that the entertainment apps included in the relevant market of the top comics app share 2% of the downloads of entertainment apps, and the video players apps included in the relevant market share 23% of the downloads of video players apps. As the off-diagonal elements of the tables are mostly zero or close to zero, the relevant markets are largely limited within the same category. An exception is the top role-playing game, whose relevant market spans to other genres of games.

9 Merger Simulation

9.1 Merger Simulation of Top Apps

To evaluate the competitiveness of the app market, we pick up the top social networking apps in terms number of downloads again. We consider a situation in which the developer of the top app acquires the closest apps in terms of elasticity. Table 9 shows the changes in the equilibrium variables and surplus accompanying a merger among 10 social networking apps. As the merging apps are fee and ad-sponsored, the merger results in an increase in advertisements but not prices.

Table 7: Comparison of download share and HHI between SSNIC and category based relevant markets of Top apps

(a) Application

		Number		Share		HHI	
	Top app in:	SSNIC	Category	SSNIC	Category	SSNIC	Category
1	Comics	79	7	0.02	0.32	0.026	0.224
2	Entertainment	82	7	0.013	0.393	0.026	0.237
3	Lifestyle	79	7	0.018	0.269	0.026	0.169
4	Maps and Navigation	78	7	0.029	0.322	0.025	0.231
5	Music and Audio	81	7	0.025	0.435	0.025	0.286
6	News and Magazines	78	12	0.048	0.238	0.025	0.177
7	Photography	80	9	0.028	0.184	0.025	0.138
8	Social	77	8	0.048	0.379	0.027	0.28
9	Tools	79	13	0.052	0.23	0.027	0.145
10	Video Players	81	12	0.039	0.215	0.026	0.16

(b) Game

		Number		Share		HHI	
	Top app in:	SSNIC	Category	SSNIC	Category	SSNIC	Category
1	Action	52	21	0.089	0.199	0.057	0.122
2	Card	64	13	0.097	0.691	0.056	0.488
3	Casino	55	9	0.029	0.477	0.059	0.272
4	Casual	53	12	0.062	0.215	0.06	0.149
5	Puzzle	48	23	0.128	0.164	0.064	0.091
6	Role Playing	65	45	0.076	0.133	0.05	0.048
7	Simulation	63	19	0.098	0.372	0.054	0.197
8	Sports	74	9	0.04	0.585	0.044	0.372
9	Strategy	61	8	0.09	0.425	0.051	0.259

Table 8: Percentage share of relevant markets of Top apps

(a) Application: Download

Target	Comics	Entertainment	Lifestyle	Maps	Music	News	Photography	Social	Tools	Video Players	Others
Comics	0.32	0.02	0.00	0.00	0.01	0.01	0.00	0.05	0.00	0.01	0.01
Entertainment	0.05	0.42	0.00	0.00	0.01	0.01	0.00	0.05	0.00	0.01	0.01
Lifestyle	0.05	0.02	0.27	0.00	0.01	0.01	0.00	0.05	0.00	0.01	0.01
Maps	0.00	0.02	0.00	0.32	0.01	0.01	0.00	0.05	0.00	0.01	0.01
Music	0.05	0.02	0.00	0.00	0.44	0.01	0.00	0.05	0.00	0.01	0.01
News	0.05	0.02	0.00	0.00	0.01	0.25	0.00	0.05	0.00	0.01	0.15
Photography	0.00	0.02	0.00	0.00	0.01	0.01	0.18	0.05	0.00	0.01	0.01
Social	0.00	0.02	0.00	0.00	0.01	0.01	0.00	0.43	0.00	0.01	0.01
Tools	0.05	0.02	0.00	0.00	0.01	0.01	0.00	0.05	0.23	0.01	0.01
Video Players	0.05	0.02	0.00	0.00	0.01	0.01	0.00	0.05	0.00	0.23	0.15

(b) Game: Download

Target	Action	Card	Casino	Casual	Puzzle	Role Playing	Simulation	Sports	Strategy	Others
Action	0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Card	0.0	0.69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Casino	0.0	0.00	0.60	0.00	0.00	0.05	0.00	0.03	0.00	0.08
Casual	0.0	0.00	0.13	0.22	0.00	0.05	0.00	0.03	0.00	0.08
Puzzle	0.0	0.00	0.13	0.00	0.16	0.05	0.00	0.03	0.00	0.08
Role Playing	0.0	0.04	0.13	0.02	0.00	0.25	0.00	0.18	0.02	0.08
Simulation	0.0	0.00	0.13	0.00	0.00	0.05	0.37	0.03	0.00	0.08
Sports	0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.58	0.00	0.00
Strategy	0.0	0.00	0.13	0.00	0.00	0.05	0.00	0.03	0.43	0.08

This increase in advertisements reduces the downloads and in-app purchases, reducing the consumer surplus, platform's profit, and total surplus. However, the magnitude of those changes are small; all the changes in the surplus are less than 0.005%. In this regard, the app market is competitive in the sense that mergers among top social networking apps have a negligible impact on surplus. Table 10 shows that similar results hold for a merger among top 10 simulation game apps. Thus, small mergers in a particular category seem to have limited impacts on surplus.

To examine the competitive effects of mergers among dominant apps, we consider the situation in which the developer of a the top app in the download ranking acquires the rest of the top apps. Table 11 shows the changes in the equilibrium variables and surplus accompanying mergers among the top 10 apps for application and game apps, respectively. Like a merger among the top social networking apps, the merger among top 10 application apps has only small effects on equilibrium variables and surplus. By contrast, the merger among the top 10 game apps has some impacts on equilibrium variables and surplus. For example, on average, the price of acquired apps increased by JPY 117, consumer surplus decreased by 2.79%, the profits of apps increased by 1.2%, the platform's profit increased by 0.7%, and the total surplus decreased by 0.8%. In this regard, a merger between dominant game developers are likely to hurt consumers, relative to that between other app developers.

Table 12 shows the impacts of larger mergers that involve the top 30 apps in the application and game apps markets. The merger among the top 30 application apps increases advertisements of merged apps and shifts the merged apps from free to paid apps; on average, the prices of merged apps increased by approximately JPY 25. Furthermore, consumer surplus decreased by 5.3%, and the total surplus decreased by 1.9%. This result illustrates the importance of endogenizing the

Table 9: The effects of mergers among 10 social apps

(a) Acquiring up to the top 10 apps: Endogenous variables

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	0.000000	-0.000001	0.034403
Acquired	0.000000	0.000986	-0.867519
Acquirer	0.000000	0.000188	-0.468932

(b) Acquiring up to the top 10 apps: Surplus

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	0.000130	0.000106	0.001586	0.000214
Acquired	0.000000	-0.000281	-0.005370	-0.000511
Acquirer	0.000002	-0.000046	0.000000	-0.000024
Total	0.000132	-0.000220	-0.003784	-0.000322

business models of app developers because the shift from free to paid apps would be ignored if we consider exogenous business models. The merger among the top 30 game apps has a larger impact on surplus. The average price of acquired apps increased by JPY 287, advertisements also increased, consumer surplus decreased by 12.8%, and the total surplus decreased by 4.7%.

In total, although mergers involving a large number of dominant apps have significant negative effects on consumers and welfare, mergers involving a small number of apps only have a moderate effect on welfare depending on their category. Note that the analysis in this section purely focuses on the static effects of a merger through the adjustment of the price and advertising intensity.

10 Platform Transaction Fee Reduction

The framework can be used to conduct other types of counterfactual analysis. For example, we can examine the effects of a reduction in the platform fee. Currently, Google Play charges a 30% transaction fee based on the download and in-app purchase revenue. Figures 8 and 9 provide the changes in the key variables and welfare measures because the transaction fee is gradually reduced from 30% to 0%. In all the figures, the value of each variable when the transaction fee is 30% is normalized to 1.

Both figures show that as transaction fee declines, advertisements declines and download prices increase. There are two mechanisms behind this result. First, the lower the transaction fee is, the more it is profitable for app developers to collect revenues from download prices. This leads to an increase prices and reduction in advertisements because of substitution. Second, because the transaction fee takes the form of a proportional fee, when the "effective marginal cost," the marginal cost minus advertising revenue and in-app-purchase revenue, is negative, the presence of

Table 10: The effects of mergers among 10 simulation apps

(a) Acquiring up to the top 10 apps: Endogenous variables

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	0.000011	-0.000046	1.240431
Acquired	0.000000	0.009132	-30.119007
Acquirer	0.000000	0.004500	-17.901005

(b) Acquiring up to the top 10 apps: Surplus

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	0.007964	0.002794	0.009728	0.005805
Acquired	0.000030	-0.012687	-0.007693	-0.007987
Acquirer	0.000278	-0.004422	-0.008318	-0.003973
Total	0.008272	-0.014316	-0.006283	-0.006155

proportional fee can reduce the prices. To illustrate, suppose that a single-product price-setting firm with negative effective marginal cost $-\psi < 0$ faces a demand function $d(p)$ and proportional fee ρ, its profit is given by

$$d(p)[(1-\rho)p+\psi] = (1-\rho)d(p)\left[p+\frac{\psi}{1-\rho}\right].$$

Then, the app developer optimally sets the price that satisfies

$$p = \left|\frac{d(p)}{d'(p)}\right| - \frac{\psi}{1-\rho}.$$

We can see that p decreases with ρ as long as $\psi > 0$. Those two mechanisms drive our result that prices decrease with transaction fees.

The effects of transaction fees on the profits of apps and platforms are straightforward. Because the fee on revenue become high, app developers' profits shrink, and under the estimated parameter values, the platform's revenue increases with transaction fees as long as it is no greater than 30%.

The impacts on consumer and total surplus are ambiguous and differ across application and game apps. For games, 12%-15% transaction fee maximizes total surplus, achieving 2.4% higher total surplus and 44% higher app developers' surplus compared to that under 30% transaction fee. For other apps, the total surplus is flat around 30%, and the reduction in the fee reduces the total surplus. A similar pattern is also observed for the effects on consumer surplus. In theory, an increase in transaction fees can cause the following welfare effects. First, as discussed before, an increase in the transaction fee rate reduces prices if effective marginal costs are negative. This aspect has a positive effect on consumer surplus. The total surplus also increases with the reduction

Table 11: Application with the efficient weight/The effects of mergers top 10 apps

(a) Acquiring up to top 10 apps: Endogenous variables of applications

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	0.000000	-0.000047	0.884679
Acquired	0.000000	0.002707	-12.272778
Acquirer	0.000000	0.003262	-25.220212

(b) Acquiring up to top 10 apps: Surplus of applications

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	0.004677	0.001495	0.001202	0.002689
Acquired	0.000877	-0.006135	0.000000	-0.003072
Acquirer	-0.000100	-0.006147	-0.133327	-0.011952
Total	0.005455	-0.010786	-0.132125	-0.012335

(c) Acquiring up to top 10 apps: Endogenous variables of games

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	0.002107	-0.000503	160.762662
Acquired	117.903021	0.008739	-5717.270257
Acquirer	47.106133	0.009945	-914.075801

(d) Acquiring up to top 10 apps: Surplus of games

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	1.089097	0.285327	1.238974	0.724560
Acquired	0.069303	-2.661012	-0.392610	-1.385643
Acquirer	0.105877	-0.415165	-0.098326	-0.197672
Total	1.264277	-2.790850	0.748038	-0.858755

Table 12: Application with the efficient weight/The effects of mergers top 30 apps

(a) Acquiring up to top 30 apps: Endogenous variables of applications

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	0.000058	-0.000131	282.426205
Acquired	27.082681	0.003346	-2887.510701
Acquirer	25.909335	0.006582	-2448.175935

(b) Acquiring up to top 30 apps: Surplus of applications

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	1.332344	0.326665	0.580994	0.725982
Acquired	0.259284	-5.032489	10.157280	-2.048225
Acquirer	0.035988	-0.598942	-4.364647	-0.597155
Total	1.627615	-5.304766	6.373627	-1.919398

(c) Acquiring up to top 30 apps: Endogenous variables of games

	Price (JPY)	Advertising (Count/Hour)	Download (Count/Week)
Outsiders	-0.052256	-0.000945	507.286218
Acquired	287.854258	0.009143	-7456.868600
Acquirer	195.613663	0.009750	-3187.544909

(d) Acquiring up to top 30 apps: Surplus of games

	App profit (%)	Consumer surplus (%)	Platform Profit (%)	Total surplus (%)
Outsiders	3.057110	0.718880	3.725121	2.048408
Acquired	1.764993	-12.697794	-2.832329	-6.421838
Acquirer	0.260919	-0.888974	0.142030	-0.334963
Total	5.083022	-12.867888	1.034822	-4.708394

in prices unless the transaction fee rate is very high. Second, an increase in the transaction fee rate increases advertisements, which has a negative effect on consumer surplus and an ambiguous effect on total surplus (Anderson and Coate, 2005). For application apps, transaction fees are positively related to both consumer surplus and total surplus as long as the transaction fee rate is not greater than 30%, indicating that the positive effect of price reduction is dominant. For game apps, the effects of transaction fees on consumer surplus and total surplus have inverted-U shapes, indicating that after certain points, the negative effects of an increase in advertisements dominate.

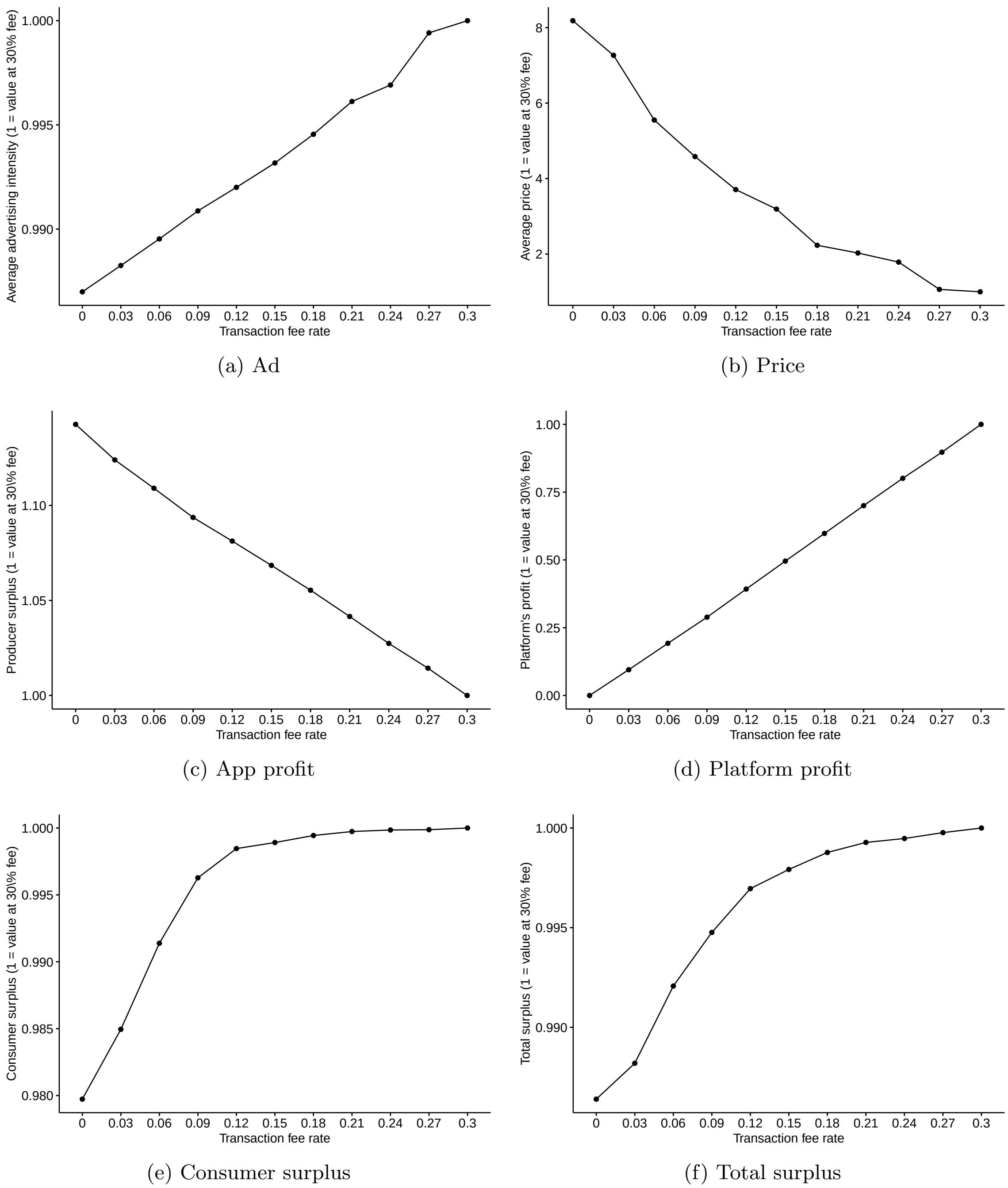
Note that in a standard vertical structure, an increase in fees increases the equilibrium prices. This is not the case in our setting, because the fee is proportional, and effective marginal costs are often negative due to advertising revenues. This result suggests that in two-sided markets, the standard result of double-marginalization may not hold and that policymakers should be careful about naively applying the existing results in one-sided markets.

There are several caveats to the interpretation of the impacts of platforms fees. Because our structural model is a static model of competition in prices and advertisements, our framework ignores long-run effects on investments, entry, and any other dynamics, which makes our results about the platforms fees on welfare bias in either a positive or negative direction. On the one hand, with a high transaction fee, few app developers would enter, and app developers would invest less, which would generate the negative welfare effects of platform fees. On the other hand, the presence of platform fees facilitates the platform's investments in market infrastructures such as payments services, SDKs, and recommendation systems, which would generate the positive welfare effects of platform fees. Those two long-run factors are absent in our static framework, which would make the results about the welfare effects of platform fees bias in either a positive or negative direction. In this regard, our result can be interpreted as a benchmark result that would be in effect in the very short run.

11 Conclusion

In this paper, we proposed a new model of imperfect competition of ad-sponsored media, which can sell "free" products, and applied it to the mobile app industry in Japan. The model allowed firms to compete with respect to both pricing and advertising revenues and endogenously select business models (paid or free media). The model incorporated word embeddings to numerically represent product information, allowing relevant markets to be defined in newly created and quickly redefined industries. We estimated the model using a rich data set on consumer downloads, usage, in-app purchases, and price and advertising information. The model introduced a novel identification strategy of unobserved advertising intensity that exploits a unique condition in the mobile app industry: that no direct marginal cost is associated with gathering sponsored advertisements.

Because of these newly introduced features of the model, we could conduct a version of the SSNIP test in which both price and sponsored advertisements are increased. We found that the relevant market defined by the test is larger than the market defined by the product category



(a) Ad

(b) Price

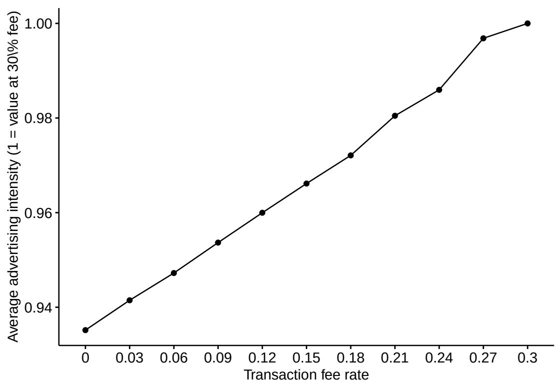
(c) App profit

(d) Platform profit

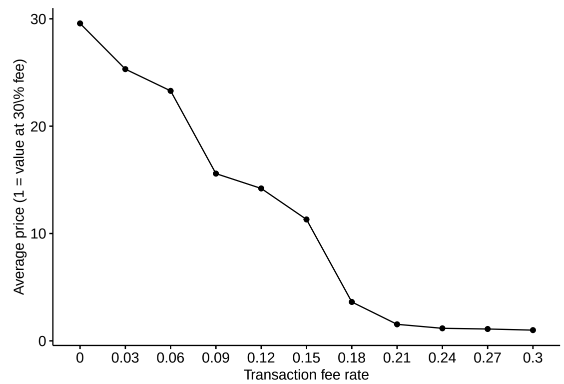
(e) Consumer surplus

(f) Total surplus

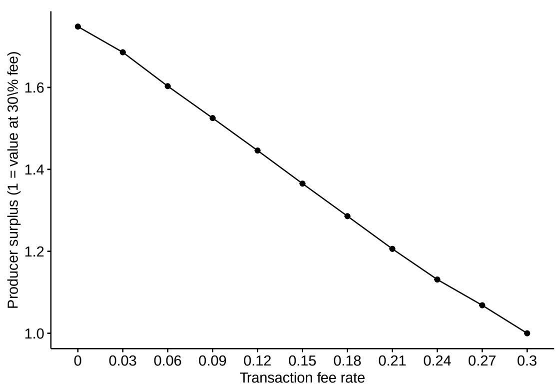
Figure 8: Application with the efficient weight/The effects of changes in platform fee



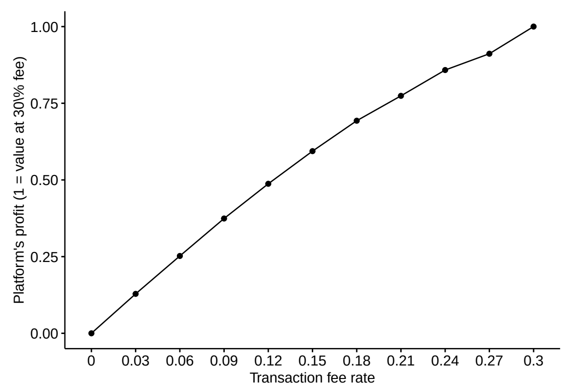
(a) Ad



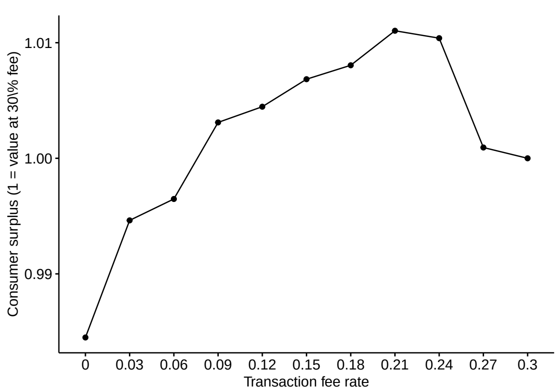
(b) Price



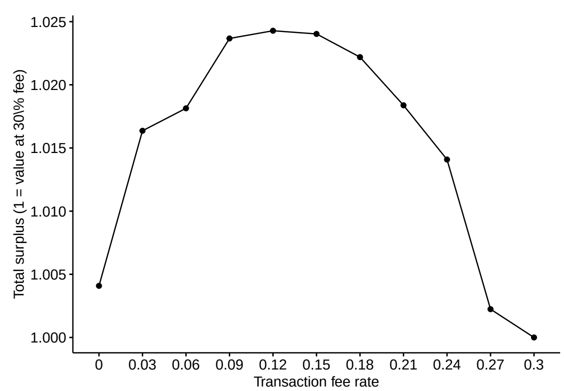
(c) App profit



(d) Platform profit



(e) Consumer surplus



(f) Total surplus

Figure 9: Game with the efficient weight/The effects of changes in platform fee

specified in Google Play. The result indicates that the defining a relevant market based on an a prior knowledge can be misleading. By conducting a full-equilibrium merger simulation, we found that the app markets were competitive, although game apps were less competitive. We also found that the vertical relationship between the platform and app developers could have more substantial welfare implications. We showed that the welfare implications of raising the transaction fee are ambiguous. The welfare-maximizing transaction fee is around 12%-15% for games, whereas it was around 30% for other apps.

This paper has several limitations. First, we assume that all firms use the ad network as price takers. In reality, some developers would not use the ad network to exert their market power in the ad market. To address this issue, we must directly observe individual advertising intensity and advertising revenue. Second, the coefficients in the usage-related indirect utility were deterministic. Making them random increases the difficulty of the computation, but would be desirable. Third, the market definition is restricted to the mobile app market. From a consumer's perspective, some apps can be a substitute for a service outside the mobile app market. For example, mobile payment services compete with credit cards. Studying the interactions between the mobile app market and the outside market is essential in analyzing the app economy. Fourth, the model is static. The dynamic aspect of the app economy, such as the entry of new apps and the growing dominance of several platforms, can be more important in merger analysis. Nevertheless, our model can serve as a static benchmark.

References

Ackerberg, Daniel A. and Marc Rysman, "Unobserved Product Differentiation in Discrete-Choice Models: Estimating Price Elasticities and Welfare Effects," *The RAND Journal of Economics*, 2005, *36* (4), 771–788.

Affeldt, Pauline, Lapo Filistrucchi, and Tobias J. Klein, "Upward Pricing Pressure in Two-sided Markets," *The Economic Journal*, November 2013, *123* (572), F505–F523.

Anderson, Simon P. and Jean J. Gabszewicz, "The Media and Advertising: A Tale of Two-Sided Markets," in Victor A. Ginsburg and David Throsby, eds., *Handbook of the Economics of Art and Culture*, Vol. 1, Elsevier, 2006, pp. 567–614.

_ and Stephen Coate, "Market Provision of Broadcasting: A Welfare Analysis," *The Review of Economic Studies*, October 2005, *72* (4), 947–972.

_ , Øystein Foros, and Hans Jarle Kind, "The Importance of Consumer Multihoming (Joint Purchases) for Market Performance: Mergers and Entry in Media Markets," *Journal of Economics & Management Strategy*, 2019, *28* (1), 125–137.

Angus, Ryan W., "Problemistic Search Distance and Entrepreneurial Performance," *Strategic Management Journal*, December 2019, *40* (12), 2011–2023.

App Annie, "App Annie 2017 Retrospective Report," Technical Report 2017.

_ , "The State of Mobile 2019," Technical Report 2019.

Armstrong, Mark and John Vickers, "Competitive Price Discrimination," *The RAND Journal of Economics*, 2001, *32* (4), 579–605.

Asahara, Masayuki, "NWJC2Vec: Word Embedding Dataset from 'NINJAL Web Japanese Corpus'," *Terminology: International Journal of Theoretical and Applied Issues in Specialized Communication*, 2018, *24* (2), 7–25.

_ , Kikuo Maekawa, Mizuho Imada, Sachi Kato, and Hikari Konishi, "Archiving and Analysing Techniques of the Ultra-Large-Scale Web-Based Corpus Project of NINJAL, Japan," *Alexandria*, 2014, *25* (1-2), 129–148.

Barlow, Matthew A., J. Cameron Verhaal, and Ryan W. Angus, "Optimal Distinctiveness, Strategic Categorization, and Product Market Entry on the Google Play App Platform," *Strategic Management Journal*, April 2019, pp. 1219–1242.

Belleflamme, Paul and Eric Toulemonde, "Tax Incidence on Competing Two-Sided Platforms," *Journal of Public Economic Theory*, 2018, *20* (1), 9–21.

Belloni, Alexandre and Victor Chernozhukov, "Least Squares after Model Selection in High-Dimensional Sparse Models," *Bernoulli*, May 2013, *19* (2), 521–547.

Benzell, Seth and Avinash Collis, "Multi-Sided Platform Strategy, Taxation, and Regulation: A Quantitative Model and Application to Facebook," *SSRN Electronic Journal*, 2020.

Berry, Steven, James Levinsohn, and Ariel Pakes, "Automobile Prices in Market Equilibrium," *Econometrica*, July 1995, *63* (4), 841.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, December 2017, *5*, 135–146.

Brynjolfsson, Erik and JooHee Oh, "The Attention Economy: Measuring the Value of Free Digital Services on the Internet," in "International Conference on Information Systems," Vol. 4 December 2012, pp. 3243–3261.

Calvano, Emilio and Michele Polo, "Strategic Differentiation by Business Models: Free-To-Air and Pay-TV," *The Economic Journal*, July 2019, *130* (625), 50–64.

Carare, Octavian, "The Impact of Bestseller Rank on Demand: Evidence from the App Market," *International Economic Review*, 2012, *53* (3), 717–742.

Cayseele, Patrick Van and Stijn Vanormelingen, "Merger Analysis in Two-Sided Markets: The Belgian Newspaper Industry," *Review of Industrial Organization*, May 2019, *54* (3), 509–541.

comScore, "The Global Mobile Report," 2017.

Crawford, Gregory S., Robin S. Lee, Michael D. Whinston, and Ali Yurukoglu, "The Welfare Effects of Vertical Integration in Multichannel Television Markets," *Econometrica*, 2018, *86* (3), 891–954.

Crémer, Jacques, Yves-Alexandre de Montjoye, Heike Schweitzer, European Commission, and Directorate-General for Competition, *Competition Policy for the Digital Era.* May 2019.

Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman, "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, 1990, *41* (6), 391–407.

Deng, Yiting, Anja Lambrecht, and Yongdong Liu, "Spillover Effects and Freemium Strategy in Mobile App Market," *SSRN Electronic Journal*, 2018.

Eisenstein, Jacob., *Introduction to Natural Language Processing*, Cambridge, MA: The MIT Press, 2019.

Emch, Eric and T Scott Thompson, "Market Definition and Market Power in Payment Card Networks," *Review of Network Economics*, 2006, *5* (1), 45–60.

Ershov, Daniel, "Consumer Product Discovery Costs, Entry, Quality and Congestion in Online Markets," 2020.

European Commission, *Commission Notice on the Definition of Relevant Market for the Purposes of Community Competition Law*, Brussels: European Commission, 1997.

Evans, D. S. and M. D. Noel, "The Analysis of Mergers That Involve Multisided Platform Businesses," *Journal of Competition Law and Economics*, September 2008, *4* (3), 663–695.

Fan, Ying, "Ownership Consolidation and Product Characteristics: A Study of the US Daily Newspaper Market," *American Economic Review*, 2013, *103* (5), 1598–1628.

Filistrucchi, Lapo, Tobias J. Klein, and Thomas O. Michielsen, "Assessing Unilateral Merger Effects in a Two-Sided Market: An Application to the Dutch Daily Newspaper Market," *Journal of Competition Law and Economics*, 2012, *8* (2), 297–329.

Firth, J. R., *Papers in Linguistics, 1934-1951*, London: Oxford University Press, 1957.

Gandhi, Amit and Jean-François Houde, "Measuring Substitution Patterns in Differentiated Products Industries," Working Paper 26375, National Bureau of Economic Research October 2019.

Gentzkow, Matthew and Jesse M. Shapiro, "What Drives Media Slant? Evidence From U.S. Daily Newspapers," *Econometrica*, 2010, *78* (1), 35–71.

_ , Bryan Kelly, and Matt Taddy, "Text as Data," *Journal of Economic Literature*, September 2019, *57* (3), 535–574.

_ , Jesse M. Shapiro, and Michael Sinkinson, "Competition and Ideological Diversity: Historical Evidence from US Newspapers," *American Economic Review*, October 2014, *104* (10), 3073–3114.

Ghose, Anindya and Sang Han, "Estimating Demand for Mobile Applications in the New Economy," *Management Science*, 2014, *60* (6), 1470–1488.

Gilbert, Richard J., *Innovation Matters: Competition Policy for the High-Technology Economy.*, Cambridge, MA:, 2020.

Goolsbee, Austan and Peter J Klenow, "Valuing Consumer Products by the Time Spent Using Them: An Application to the Internet," *American Economic Review*, 2006, *96* (2), 7.

Han, Sang Pil, Sungho Park, and Wonseok Oh, "Mobile App Analytics: A Multiple Discrete-Continuous Choice Framework," *MIS Quarterly*, 2016, *40* (4), 983–1008.

Ifrach, Bar and Ramesh Johari, "The Impact of Visibility on Demand in the Market for Mobile Apps," *SSRN Electronic Journal*, 2014.

Ivaldi, Mark and Szabolcs Lorincz, "Implementing Relevant Market Tests in Antitrust Policy: Application to Computer Servers," *Review of Law and Economics*, 2011, *7* (1), 31–73.

Japan Fair Trade Commission, "The Revised Guidelines to Application of the Antimonopoly Act Concerning Review of Business Combination," December 2019.

Jara-Díaz, Sergio and Jorge Rosales-Salas, "Beyond Transport Time: A Review of Time Use Modeling," *Transportation Research Part A: Policy and Practice*, March 2017, *97*, 209–230.

Jeon, Doh-Shin, Byung-Cheol Kim, and Domenico Menicucci, "Second-Degree Price Discrimination by a Two-Sided Monopoly Platform," *Working Paper*, 2016.

Jeziorski, PrzemysŁaw, "Effects of Mergers in Two-Sided Markets: The US Radio Industry," *American Economic Journal: Microeconomics*, 2014, *6* (4), 35–73.

Kesler, Reinhold, Michael E. Kummer, and Patrick Schulte, "Mobile Applications and Access to Private Data: The Supply Side of the Android Ecosystem," *SSRN Electronic Journal*, 2017.

Kudo, Taku, “MeCab : Yet Another Part-of-Speech and Morphological Analyzer,” 2005.

Kwark, Young and Paul A Pavlou, “On the Spillover Effects of Online Product Reviews on Purchases : Evidence from Clickstream Data,” *SSRN Electronic Journal*, 2019.

Leyden, Benjamin T, “There ’ s an App for That,” 2018.

Lin, Song, “Two-Sided Price Discrimination by Media Platforms,” *Marketing Science*, 2020, *39* (2), 317–338.

Liu, Yongdong, “Mobile App Platform Choice,” *SSRN Electronic Journal*, 2017, pp. 1–48.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, “Distributed Representations of Words and Phrases and Their Compositionality,” in C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds., *Advances in Neural Information Processing Systems 26*, Curran Associates, Inc., 2013, pp. 3111–3119.

_ , Kai Chen, Greg Corrado, and Jeffrey Dean, “Efficient Estimation of Word Representations in Vector Space,” *arXiv:1301.3781 [cs]*, September 2013.

Newman, John M., “Antitrust in Zero-Price Markets: Foundations,” *University of Pennsylvania Law Review*, 2015, *164*, 149–206.

P., Simon Anderson and Martin Peitz, “Media Sea-Saws: Winners and Losers in Platform Markets,” *Journal of Economic Theory*, 2020, *186.*

Pantea, Smaranda and Bertin Martens, “The Value of the Internet as Entertainment in Five European Countries,” *Journal of Media Economics*, January 2016, *29* (1), 16–30.

Pennington, Jeffrey, Richard Socher, and Christopher Manning, “Glove: Global Vectors for Word Representation,” in “Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)” Association for Computational Linguistics Doha, Qatar 2014, pp. 1532–1543.

Pervin, Nargis, Narayan Ramasubbu, and Kaushik Dutta, “Habitat Traps in Mobile Platform Ecosystems,” *Production and Operations Management*, October 2019, *28* (10), 2594–2608.

Recode, “Spotify Says Apple Won’t Approve a New Version of Its App Because It Doesn’t Want Competition for Apple Music - Recode,” https://www.recode.net/2016/6/30/12067578/spotify-apple-app-store-rejection June 2016.

Řehůřek, Radim and Petr Sojka, “Software Framework for Topic Modelling with Large Corpora,” in “Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks” ELRA Valletta, Malta May 2010, pp. 45–50.

Sato, Susumu, “Freemium as Optimal Menu Pricing,” *International Journal of Industrial Organization*, 2019, *63*, 480–510.

Sato, Toshinori, Taiichi Hashimoto, and Manabu Okumura, “Implementation of a Word Segmentation Dictionary Called Mecab-Ipadic-NEologd and Study on How to Use It Effectively for Information Retrieval (in Japanese),” in “Proceedings of the Twenty-Three Annual Meeting of the Association for Natural Language Processing” 2017, pp. NLP2017–B6–1.

Song, Minjae, "Estimating Platform Market Power in Two-Sided Markets with an Application to Magazine Advertising," *SSRN Electronic Journal*, 2011.

A Competition-in-Utility Transformation

We transform the original model of price and advertising competition into a framework of competition-in-utility. A competition-in-utility framework considers a per-consumer profit function $\bar{\pi}_j(\delta_j)$ for each app j that gives mean utility δ_j. The analysis below derives the exact form of the per-consumer profit function $\bar{\pi}_j$

Suppose that a developer d provides mean utility δ_j to consumers who use app $j \in \mathcal{J}_d$. To guarantee that an app j provides utility δ_j, it must satisfy equation (27). Thus, for fixed a_j and δ_j, F_j must satisfy

$$\alpha_y F_j = \max\left\{\frac{\eta_j}{2}\tilde{q}_j^2 + \beta_d' X_{dj} + \xi_{dj} - \delta_j, 0\right\}.$$

Let $\bar{F}_j(a_j, \delta_j)$ be defined by the value of F_j that satisfies the above equation. Then, the profit of app j from providing mean utility δ_j and advertisements a_j is given by

$$s_j \hat{\pi}_j(a_j) = s_j\left[(1-\rho)(\bar{F}_j + \tilde{e}_j) + \tilde{q}_j(a_j r - \lambda_j) - \epsilon_j\right].$$

Suppose that $\bar{F}_j > 0$. Then, $\hat{\pi}_j(a_j)$ can be written as

$$\frac{(1-\rho)}{\alpha_y}\frac{\eta_j}{2}\tilde{q}_j^2 + (1-\rho)\tilde{e}_j + \tilde{q}_j(a_j r - \lambda_j) + \frac{(1-\rho)}{\alpha_y}\left[\beta_d' X_{dj} + \xi_{dj} - \delta_j\right] - \epsilon_j$$

Then, the optimal advertising intensity if characterized by the first-order condition

$$\frac{\partial \tilde{q}_j}{\partial a_j}\left[\frac{(1-\rho)}{\alpha_y}\eta_j \tilde{q}_j + a_j r - \lambda_j\right] + (1-\rho)\frac{\partial \tilde{e}_j}{\partial a_j} + \tilde{q}_j r \leq 0,$$

with equality if $a_j > 0$. Solving this first-order condition, we obtain the value of a_j that maximizes $\bar{\pi}_j$. Let a_j^{int} be such a value, which is independent of δ_j. Thus, as long as $\bar{F}_j(a_j^{int}, \delta_j) > 0$, the optimal advertising intensity is given by a_j^{int}, and $\partial \bar{F}_j / \partial \delta_j = -1/\alpha_y$.

Next, consider the case where $\bar{F}(a_j^{int}, \delta_j) = 0$. In this case, we must have

$$\frac{\eta_j}{2}\tilde{q}_j^2 + \beta_d' X_{dj} + \xi_{dj} - \delta_j = 0.$$

If there is a solution to this equation in $[0, a_j^{int}]$, such advertising intensity gives the mean utility δ_j. Let $\hat{a}_j(\delta_j)$ be such an advertising intensity. The derivative of $\hat{a}_j$ is given by

$$\hat{a}_j'(\delta_j) = \frac{-1}{\alpha_{aj}\tilde{q}_j}.$$

Let $\bar{\delta}_j$ be the value of mean utility such that

$$\frac{\eta_j}{2}\tilde{q}_j^2 + \beta_d' X_{dj} + \xi_{dj} - \delta_j = 0$$

holds at $a_j = a_j^{int}$ and δ_j^0 be the mean indirect utility of app j at $a_j = F_j = 0$. Then, the app j can provide mean utility $\delta_j \in (-\infty, \delta_j^0]$, and the per-consumer profit function $\bar{\pi}_j(\delta_j)$ is given by

$$\bar{\pi}_j(\delta_j) = \begin{cases} \frac{(1-\rho)}{\alpha_y}\frac{\eta_j}{2}(q_j^{int})^2 + (1-\rho)e_j^{int} + q_j^{int}(a_j^{int} r - \lambda_j) + \frac{(1-\rho)}{\alpha_y}\left[\beta_d' X_{dj} + \xi_{dj} - \delta_j\right] - \epsilon_j & \text{if } \delta_j < \bar{\delta}_j, \\ (1-\rho)\hat{e}_j + \hat{q}_j(\hat{a}_j r - \lambda_j) - \epsilon_j & \text{if } \delta_j \in [\bar{\delta}_j, \delta_j^0], \end{cases} \tag{53}$$

which has the derivative

$$\bar{\pi}_j'(\delta_j) = \begin{cases} -\frac{1-\rho}{\alpha_y} & \text{if } \delta_j < \bar{\bar{\delta}}_j \\ -\frac{r}{\alpha_{aj}} + \frac{1}{\eta \hat{q}_j}\left[(1-\rho)\frac{\hat{e}_j}{\hat{q}_j} + \hat{a}_j r - \lambda_j\right] & \text{if } \delta_j \in [\bar{\bar{\delta}}_j, \delta_j^0], \end{cases} \tag{54}$$

where the variables with hats $(\hat{q}, \hat{e})$ and a superscript int, (q^{int}, e^{int}) are the values of functions evaluated at $\hat{a}_j$ and a_j^{int}. Note that we have $\lim_{\delta_j \searrow \bar{\bar{\delta}}_j} \bar{\pi}_j'(\delta_j) = -\frac{1-\rho}{\alpha_y}$, which implies that $\bar{\pi}_j$ is smooth at every point in $(\infty, \delta_j^0]$.

Using this transformation, we can reformulate the developer's problem as

$$\max_{\{(\delta_j)\}_{j \in \mathcal{J}_d}} \sum_{j \in \mathcal{J}_d} s_j(\delta) \bar{\pi}_j(\delta_j) \tag{55}$$

$$\text{s. t.} \quad \delta_j \leq \delta_j^0, \quad j \in \mathcal{J}_d. \tag{56}$$

The first-order conditions for this problem are:

$$s_j \bar{\pi}_j'(\delta_j) + \sum_{k \in \mathcal{J}_d} \frac{\partial s_k}{\partial \delta_j} \bar{\pi}_k(\delta_k) \leq 0, \tag{57}$$

with equality if $\delta_j < \delta_j^0$.

B Quadratic-Programming for Eliciting Advertising Intensities

In this section, we outline the quadratic-programming procedure for eliciting advertising intensities used in Section 6.

Fix θ. Given the average indirect utilities $(\delta_j)_{j \in \mathcal{J}}$, and parameters θ, we can compute $\partial s_{mk}/\partial a_j$ independent of $(a_j)_{j \in \mathcal{J}}$. Making use of this property, we derive the profile of advertising intensity by conducting the following quadratic-programming procedure. First, to elicit the advertising intensities $\{a_j\}_{j \in \mathcal{J}}$, we utilize the first-order conditions for profit-maximizing advertising intensities given each parameters, which is characterized by

$$\begin{aligned} &s_j \left(q_j r + \frac{\partial q_j}{a_j}(a_j r - \lambda_j) + (1-\rho)\frac{\partial e_j}{\partial a_j} \right) \\ &-\alpha_{aj} q_j \sum_{k \in \mathcal{J}_d} \frac{\partial s_k}{\partial \delta_j}((1-\rho)(F_j + e_j) + q_j(a_j r - \lambda_j)) + \mu_j = 0, \mu_j a_j = 0, \mu_j \geq 0, a_j \geq 0, \end{aligned} \tag{58}$$

for all $j \in \mathcal{J}$. Given the data $(s_j, q_j, e_j)_{j \in \mathcal{J}}$, mean indirect utilities $(\delta_j)_{j \in \mathcal{J}}$, and parameters θ, we can compute the simulated value of $\partial s_k/\partial\delta_j$. Let

$$\begin{aligned}\omega_j(a) =& s_j\left(q_j r + \frac{\partial q_j}{\partial a_j}(a_j r - \lambda_j) + (1-\rho)\frac{\partial e_j}{\partial a_j}\right) - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j}((1-\rho)(F_k+e_k) + q_k(a_k r - \lambda_k)) \\ =& s_j q_j r - s_j\frac{\partial q_j}{\partial a_j}\lambda_j + (1-\rho)\frac{\partial e_j}{\partial a_j} - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j}((1-\rho)(F_k+e_k) - q_k\lambda_k) \\ &+ s_j\frac{\partial q_j}{\partial a_j} r a_j - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j} q_k r a_k.\end{aligned} \tag{59}$$

Let $\tilde{a}_j = q_j a_j$ and $\tilde{a} \coloneqq (\tilde{a}_j)_{j\in\mathcal{J}}$. Then, $\omega_j(a)$ can be written as a function of $\tilde{a}$, $\tilde{\omega}(\tilde{a})$ as

$$\tilde{\omega}_j(\tilde{a}) = s_j q_j r + (1-\rho)\frac{\partial e_j}{\partial a_j} - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j}(1-\rho)(F_k+e_k) + s_j\frac{\partial q_j}{\partial a_j}\frac{r}{q_j}\tilde{a}_j - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j} r\tilde{a}_k, \tag{60}$$

which can be written in vector form

$$\tilde{\omega}(\tilde{a}) = \gamma - \Gamma\tilde{a}, \tag{61}$$

where $\gamma = (\gamma_j)_{j\in\mathcal{J}}$ is given by

$$\gamma_j = s_j q_j r + (1-\rho)\frac{\partial e_j}{\partial a_j} - \alpha_{aj} q_j \sum_{k\in\mathcal{J}_d}\frac{\partial s_k}{\partial \delta_j}(1-\rho)(F_k+e_k)$$

for each $j \in \mathcal{J}_d$, and $\Gamma = (\Gamma_{ij})_{i,j\in\mathcal{J}}$ is given by

$$\Gamma_{ij} = \begin{cases} s_j\frac{\partial q_j}{\partial a_j}\frac{r}{q_j} - \alpha_{aj} q_j \frac{\partial s_j}{\partial \delta_j} r & \text{if } i = j, \\ -\alpha_{aj} q_j \frac{\partial s_j}{\partial \delta_j} r & \text{if } i \neq j, \text{ and there exists } d\in\mathcal{D} \text{ such that } i,j\in\mathcal{J}_d \\ 0 & \text{otherwise.}\end{cases}$$

Then, we claim that we can obtain the values of $\{a_j\}_{j\in\mathcal{J}}$ by solving the following minimization problem

$$\max_{\tilde{a}} \gamma'\tilde{a} - \frac{1}{2}\tilde{a}'\Gamma\tilde{a} \quad \text{subject to } \tilde{a} \geq 0 \tag{62}$$

and plugging the solution into $a_j = \tilde{a}_j/q_j$.

Whenever possible, solving this maximization problem yields exactly the same first-order condition with the original problem at the true parameters (θ) and average indirect utilities $(\delta_j)_{j\in\mathcal{J}_d}$ which is shown by the following logic. First, when the quadratic programming (62) has a concave objective function, we can find a unique pair of profiles of advertising intensities and Lagrange multipliers $(a_j, \mu_j)_{j\in\mathcal{J}}$ that achieves the global maximum of the objective function. Next, we claim that the true values $(\tilde{a}_j) = (q_j a_j)$ and $(\mu_j)_{j\in\mathcal{J}}$ in the equilibrium satisfy the necessary and sufficient condition for the solution to this quadratic programming. The Kuhn-Tucker condition for the quadratic programming (62) is given by

$$\tilde{\omega}(\tilde{a}) + \tilde{\mu}_j = 0, \tilde{\mu}_j\tilde{a}_j = 0, \tilde{\mu}_j \geq 0, \tilde{a}_j \geq 0,$$

which is equivalent to the first-order condition of the original model (58). Thus, the solution to this quadratic programming corresponds to the equilibrium value at the true parameter value.

C Monte Carlo Simulation

Tables 13-15 illustrate how the equilibrium prices and advertising intensities depend on the underlying model parameters. In the simplest case, both α_{aj} and η_j are common across the apps. In each environment, we set the baseline parameters as $\alpha_{aj} = 0.05$, $\alpha_y = 0.1$ $\eta_j = 2$, $\lambda_j = 0$ for all j, $\beta_u = 0.1$, $\beta_d = 1$, and $\sigma_k = 0$ for all k. For advertising prices and wages, we set $r = 5$ and $w = 0$, and for firm-level quality shocks, we set $\xi_{uj} = 0$, $\xi_{dj} = -3$, and $\xi_{ej} = 0.0001$ for all j. We consider three scenarios: symmetric duopoly, asymmetric duopoly with heterogeneity in ξ_{ej}, and asymmetric duopoly with heterogeneity in ownership structure. In an asymmetric duopoly with heterogeneity in the ownership structure, one firm provides three products, and another firm provides only one product.

Table 13 shows how equilibrium advertising intensities and download prices depend on parameters α_y, α_{aj}, and η_j. We set the value of the equilibrium variable under baseline parameters to 1 and show the relative values of the equilibrium variables with different parameters values. The greater the marginal utility of income, the greater the equilibrium advertising intensities and the smaller the equilibrium download prices. This occurs because of the substitution from download prices to advertising intensities to effectively collect revenues. A greater marginal utility of income increases consumers' costs for downloading apps, leading to lower equilibrium download prices. This reduces the foregone revenues from the reduction in downloads induced by advertisements, leading to higher advertising intensities. By contrast, an increase in α_j decreases equilibrium advertising intensities and increases equilibrium download prices. Finally, an increase in η_j increases download prices but does not affect advertising intensities as long as download prices are positive. Because an increase in satiation reduces usage time, advertising revenues become smaller. As a result, firms have weaker incentives to lower prices to attract consumers, leading to higher download prices. The neutrality of η_j on advertising intensities stems from the fact that usage time is proportional to η_j and that optimal advertising intensities for paid app are independent of the scale of usage times.

Tables 14 and 15 show similar comparative statics with asymmetric firms. In each case, we name the firm with a higher equilibrium download price as the "strong" firm and the firm with a lower equilibrium download price as the "weak" firm. We set the value of the equilibrium variables of the weak firm to 1 and show the relative value of other equilibrium variables with different parameters. In Table 14, apps differ in ξ_{ej}, and firms with larger values of ξ_{ej} set higher download prices and smaller advertising intensities. A larger ξ_{ej} results in a firm being more willing to collect revenues from in-app purchases rather than advertising revenues, leading to lower advertising intensities. The effect of ξ_{ej} depends on its effects on in-app purchases and market shares. Whereas an increase in ξ_{ej} increases in-app purchases, which tend to decrease download prices to attract more consumers, the increase in market share accompanying the increase in the usage value leads to the greater market power and higher download prices. In our example, the latter effect dominates and, thus, the firm with a large value of ξ_{ej} sets higher download prices. In Table 15, firms differ in the number of apps they provide, and firms with a larger set of apps set higher download prices to avoid cannibalization among apps provided by the same developer. However, firms set the same advertising intensities as those with a smaller set of app, as long as the equilibrium download prices are positive. This phenomenon results from the well-known fact in the media literature that, as long as the price can be flexibly chosen, any impact of competition on advertising intensities is neutralized by changes in download prices (Anderson and Gabszewicz, 2006).

Our model also allows monetization policies to vary depending on the ownership structure. Table 16 provides examples of equilibria under two environments in which only the ownership structures differ. One equilibrium is under a single-product duopoly, and another is under a two-product monopoly. Under a single-product duopoly, download prices are set to zero and are strictly

Table 13: Comparative statics in symmetric oligopoly

Baseline parameter, $(\alpha_y, \alpha_a, \eta) = (0.1, 0.05, 2)$:

	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
Ad	1	1.0003	0.9897	1

	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
price	1	0.99	1.0001	1.0001

Table 14: Comparative statics with asymmetry in ξ_e

Baseline parameter, $(\alpha_y, \alpha_a, \eta) = (0.1, 0.05, 2)$:

	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
Ad (weak firm)	1.0000	1.0003	0.9897	1.0000
Ad (strong firm)	0.9953	0.9957	0.9850	0.9953

	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
Price (weak firm)	1.0000	0.9900	1.0001	1.0001
Price (strong firm)	1.0001	0.9901	1.0002	1.0002

Table 15: Comparative statics with asymmetry in ownership

Baseline parameter, $(\alpha_y, \alpha_a, \eta) = (0.1, 0.05, 2)$:

compara	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
Ad (weak firm)	1	1.0003	0.9897	1
Ad (strong firm)	1	1.0003	0.9897	1

	Baseline	$\alpha_y = 0.101$	$\alpha_a = 0.0505$	$\eta = 2.02$
Price (weak firm)	1.0000	0.990	1.0001	1.0001
Price (strong firm)	0.8596	0.851	0.8597	0.8597

positive under a two-product monopoly because single-product firms have greater incentives to set lower prices to attract consumers from apps provided by other firms. This example illustrates how monetization policies can vary according to changes in ownership structures, such as mergers.

Table 16: Ownership structures and monetization regimes

	ad	price
multi-product monopoly	1.0484	1.3403
single-product duopoly	1.0472	0.0000