Optimizing Click-through in Online Rankings for Partially Anonymous Consumers*

Babur De los Santos† Sergei Koulayev‡

November 15, 2014
First Version: December 21, 2011

Abstract

Consumers incur costly search to evaluate the increasing number of product options available from online retailers. Presenting the best alternatives first reduces search costs associated with a consumer finding the right product. We use rich data on consumer click-stream behavior from a major web-based hotel comparison platform to estimate a model of search and click. We propose a method of determining the hotel ordering that maximizes consumers' click-through rates (CTRs) based on partial information available to the platform at that time of the consumer request, its assessment of consumers' preferences, and the expected consumer type based on request parameters from the current visit. Our method has two distinct advantages. First, rankings are targeted to anonymous consumers by relating price sensitivity to request parameters, such as the length of stay, number of guests, and day of the week of the stay. Second, we take into account consumer response to new product ordering through the use of search refinement tools, such as sorting and filtering of product options. Product ranking and search actions together shape the consideration set from which clicks are made. We find that predicted CTR under our proposed ranking are almost double those under the online platform's default ranking.

Keywords: consumer search, hotel industry, popularity rankings, platform, collaborative filtering, click-through rate, customization, targeting, sorting, filtering, search refinement

*A previous version of the paper was circulated under the title: "Using Consumer Preferences to Improve upon Popularity Rankings in Online Markets." We thank Elizabeth Honka, Ali Hortaçsu, Joowon Kim, Matthijs Wildenbeest, and participants of the Industrial Organization Society sessions at the Allied Social Sciences Association annual meeting, the 4*th* Workshop of Switching and Search Costs, and seminar participants at the University of Iowa and the University of Arizona for helpful comments. The views expressed are those of the authors and do not necessarily reflect those of the Consumer Financial Protection Bureau or the United States.

†Kelley School of Business, Indiana University, E-mail: babur@indiana.edu.

‡Consumer Financial Protection Bureau, E-mail: sergei.koulayev@gmail.com.

1 Introduction

In recent years, Internet commerce has achieved significant advances in the reach and complexity of recommendation systems that shape consumer choices in almost all areas of commerce. Consumers search for product information in online search-engine results, in order to evaluate a wide range of products, such as consumer electronics, books, mortgages, hotel rooms, and flights. Although the problem of optimal recommendation is not new—salespeople everywhere have long struggled with which product to endorse to a particular consumer—the vast amount of consumer information generated online has revolutionized the way firms collect and analyze consumer data, customize product choices, as well as target product recommendations (see, e.g., Ansari and Mela, 2003).

At the core of product recommendations is the challenge of matching a set of products to a set of consumers whose tastes are heterogeneous and often unobserved. The accuracy of the match depends on how firms leverage available information to infer consumer preferences. For example, Amazon makes recommendations in diverse product categories by exploiting product correlations present in other customers' transaction histories. Although potential buyers' preferences for a new product are unobservable, the modern recommendation systems are typically based on the notion that preferences for various products are similar across the set of customers who bought or rated other products in a similar way. Such is the underlying assumption of collaborative filtering and other hybrid methodologies including Anderson, Ball, Boley, Greene, Howse, Lemire, and McGrath (2003); Basilico and Hofmann (2004); Huang, Zeng, and Chen (2004); Melville, Mooney, and Nagarajan (2001); Vozalis and Margaritis (2008); Yu, Schwaighofer, and Tresp (2003); and Moon and Russell (2008).

Unfortunately, the collaborative approach is infeasible in the case of anonymous consumers, for whom the platform has no knowledge of past choices, and who comprise a substantial share of all platform visitors. Recommendations for such visitors can be made using distributional information regarding tastes for the population, which can be inferred from utility-based models of discrete choice. From the existing literature on consumer choice, we know that such distributional parameters play an important role in the substitution patterns of consumer demand, which are at the core of the recommendation problem.

This paper proposes a model of search and click which endogenizes consumers' search refinement actions, such as sorting and filtering, as well as product clicks on a search platform. Based on this model, we propose a method of determining the hotel ordering that maximizes consumers' click-through rates (CTRs) based on partial information available to the platform at the time of the consumer request. Our approach offers several novel features.

First, our method is a utility-based ranking that targets individuals with rankings by exploiting the relationship between the request parameters submitted to the platform and consumer tastes for hotel attributes, including price. After request parameters are controlled for, there remains a significant unobserved component in consumer tastes. The discrete choice model delivers estimates regarding the shape of the distribution of that component, which we also employ in the construction of the ranking. Because the objective function is non-linear in the unobserved component, always targeting the median type is sub-optimal.

Second, we emphasize that consumers' search actions, in addition to their product choices, are informative and useful in estimating consumers' preferences and should be employed in building a recommendation. One way to elicit such information is through a structural model of consumer search. In doing so, we follow the recent literature that incorporates search into discrete choice models of consumer demand. Notably, Kim, Albuquerque, and Bronnenberg (2010) use aggregate consumer search data from Amazon to estimate demand models of durable products. Other papers, such as De los Santos, Hortaçsu, and Wildenbeest, 2012; Ghose, Iperiotis, and Li, 2014; and Koulayev, 2014, incorporate micro-data on search activity.

The idea of utility-based ranking has received attention in the literature on recommendation systems. For instance, Ansari et al. (2000), Belanger (2005), Burke (2000), Ricci et al. (2011). In the marketing literature, the closest work is by Ghose et al. (2012), who incorporate user-generated content (e.g., online hotel reviews) into the ranking, based on a discrete choice model. The main difference in our approach is that we explicitly account for consumers' search refinement actions in construction of the ranking, and we directly compare this ranking to those obtained from static models. The evaluation methods also differ: while Ghose et al. (2012) use lab experiments, we estimate counterfactuals in our model of search and click to compute predicted CTR under alternative rankings.

Section 2 presents a model of consumer choice where both search actions and clicks are jointly determined. The model explicitly accounts for the endogeneity of consumers' search refinement actions on the ranking, such as sorting and filtering of search results. Accounting for search actions is important as the composition of consideration sets results from both the default product ranking and user search refinements. For example, searchers who sort hotels by increasing price tend to observe cheaper hotels that are further away from city center. If the proportion of such searchers is substantial, one might improve the diversity of alternatives by giving prominence to hotels closer to city center or of higher quality. As future search refinements are unobserved, we evaluate a candidate ranking by integrating over potential search strategies.

Section 3 analytically derives the platform's problem of recommendation ranking. We propose a targeted ranking of search results that maximizes the aggregate click-through rate (CTR), which is a standard measure of product relevance. Click maximization is an important factor both for short-term profit objectives and for the long-term success of a search platform, for which consumer satisfaction and retention are key factors.

A major challenge for a recommendation system is to serve the category of partially anonymous consumers. Such customers are anonymous to the platform and their past behavior is unobserved (including both their search and clicking behavior) but relevant observable information is available in the current customer visit. For instance, a search request for a hotel room in an online travel platform, which includes the date of stay and the number of guests, reveals that the consumer is more likely to be a business traveler if the stay is during weekdays, if the room request is for single occupancy, and if the request was made fewer (vs. more) days in advance of the stay. Even though preferences of a particular consumer are unobserved, a significant portion of heterogeneity can be inferred from the request parameters and can be used to target rankings of search results to individuals.

Furthermore, the information about the general shape of the distribution of the unobserved component of consumer tastes can also be employed to achieve better targeting. Indeed, targeting the median type is not optimal because the CTR is non-linear in the unobserved component. For example, even if price sensitive consumers account for the majority of the customers, it may be optimal to prioritize higher-quality, more expensive products during periods of high prices: in such periods, the CTR gains among quality centered consumers may outweigh click losses among price-sensitive ones.[1]

[1]There are several other sources information about the consumer that can be used to improve and target rankings.

Section 4 presents the empirical application of the model. We use rich data on consumer search and click-stream behavior within a major web-based hotel comparison platform. We compare estimates of a search and click random coefficient model to different discrete choice specifications in terms of clicks that do not endogenize consumer search. The search and click model not only provides a better fit in terms of search actions and clicks, but also a better fit of clicks conditional on observed search behavior. The reason the search model is better able to explain clicks is that it uses inequalities imposed by search decisions to update expectations regarding the distribution of consumer tastes. In this way, search decisions are actually informative about consumer preferences and ignoring such decisions might lead to biased estimates.

Section 5 presents comparisons of the performance of optimal ranking from the search and click model with various alternative rankings, including the default ranking that was implemented by the platform at the time. The search model ranking leads to an almost twofold increase in CTR compared to the default ranking used by the platform at the time of data collection. While the exact formula used by the platform is proprietary, its primary component is a hotel's past popularity, measured by the number of clicks the hotel received during a specified time window. Although intuitively appealing, the popularity ranking has two major drawbacks that might explain the large gain observed for the search model ranking. First, it fails to adjust to changes in price, which is particularly problematic for the hotel industry where prices change frequently due to demand fluctuations and revenue management models. Second, the popularity ranking suffers from a feedback effect, which causes the ranking to persist over time.

Section 6 concludes with managerial implications. Online platforms have the unprecedented ability to offer consumers a means of analyzing and comparing product attributes for a large set of products. Incorporating consumer choice models that allow for heterogeneous tastes on a variety of attributes lets companies more precisely present consumers with preferred alternatives. As Hendricks, Sorensen and Wiseman (2010) point out, in these situations superior alternatives placed in worse positions in the ranking may be ignored while consumers waste time evaluating inferior products that appear in better positions. These findings suggest that the order in which options

For instance, the widespread use of "cookies" allow platforms track consumers' online behavior. Even in the absence of cookies, other useful information might include the user's geographical location, type of browser, IP address, etc. as well as the consumer's browsing behavior. A recent example of the use of such information is the new ranking strategy of travel site Orbitz, which after observing that Mac users spend more on hotel rooms, decided to present pricier options to consumers who were searching from a Mac (Wall Street Journal, June 26, 2012).

are presented greatly influences consumer choices. In addition to increased profitability for the platform through higher CTRs, reducing search frictions will likely increase consumer satisfaction, as well as retention rates.

2 A Model of Consumer Search and Click

In this section, we present a model of consumer choice where both search and click decisions are jointly determined. The main innovation of the paper is that the model allows us to incorporate a potentially large set of search actions into a discrete choice framework. Search actions, including browsing, sorting, and filtering the set of default results, produce a different set of results for each consumer. The main challenge is the non-stationary dependence of search results to previous search actions. Instead of a dynamic model which potentially imposes restrictions on heterogeneity, we propose a flexible repeated static model that allows us to identify the distribution of expected utilities faced by a consumer at any stage of their search process.

Let i index consumers and let $t = 1, \ldots, T_i$ index pages that a consumer observes during her search process, including search refinement actions. A consumer search session consists of a sequence of search actions and hotel clicks. Every search action results in a display of hotels, therefore a search session also consists of a sequence of displays, or ranked sets of hotels, denoted by $\mathbf{D}_{ti}$. Potentially, several search sessions may belong to the same consumer, but we do not observe this; therefore, we will assume that every search session is made by a separate consumer.

When a new set of hotels appears on the screen, the consumer clicks on the hotel that she finds most attractive. At the click stage, let u_{ij} be the attractiveness of some hotel j for consumer i. In general, because the consumer has limited information about a hotel at the click stage, this click utility is not the actual booking utility. Rather, u_{ij} can be interpreted as an expected utility, which will be refined after the consumer clicks through and finds additional information on the hotel details page.

Even though consumers observe many hotels during their search, only a few actually are clicked. The relative rarity of hotel clicks indicates that clicks are costly (for example, due to the cost of processing information on the landing page). Structural models of costly click, such as Ghose et al. (2012), imply that a click is made only if the utility of the clicked hotel exceeds a certain consumer-specific threshold. In our model, we capture this effect in reduced form by introducing a consumer-specific outside option and allow a consumer to make a choice or a search action on each page t she observes. We further assume that at most one click per page t can be made. This assumption is made to simplify computations. The data show that fewer than 2 percent of displays have had more than one click, making this a reasonable assumption.

Let $j \in \mathbf{D}_{ti}$ index hotels and let j^* denote hotel clicked by consumer i in page t. Notice that although for simplicity we write j instead of j_{ti}, there is a choice for each page j^*_{ti}. Define outside option as $j = 0$ and the choice of not clicking on a page as $j^* = 0$. Similar to Kim, Albuquerque, and Bronnenberg (2010) we use utility inequalities derived from the search and click process. In our case, the observed click patterns involve the following inequalities for each display $\mathbf{D}_{ti}$ observed by consumer i:

$$u_{ij^*} \geq u_{ij},\ j^* \neq 0 \tag{1}$$

$$u_{ij^*} \geq u_{i0},\ j^* \neq 0 \tag{2}$$

$$u_{ij} \leq u_{i0},\ j^* = 0 \tag{3}$$

Formally, the model of click utility is:

$$u_{ij} = \alpha_i p_{ij} + \beta_i^{'} X_j + \delta_i L_{ij} + \varepsilon_{ij} \tag{4}$$

Parameter α_i measures a consumer's price sensitivity, while β_i is a vector of tastes for a set of non-price characteristics, X_j, that can be observed on the display: brand, star rating, neighborhood, distance to the city center. The parameter δ_i measures the effect of the hotel's position in the display, L_{ij}, on the probability of click. Recall from Figure (2) that on average consumers are more likely to click on hotels located at higher positions. We interpret $\delta_i L_{ij}$ as being related to the cost of within-page search rather than being a part of inherent consumer utility that is attributable to hotel characteristics, $\alpha_i p_{ij} + \beta_i^{'} X_j$. This distinction is not important for prediction of clicks, but will be important for the recommendation ranking. Lastly, ε_{ij} are i.i.d. type I extreme value error terms that represent the consumer's idiosyncratic preference for a particular hotel.

The utility of the outside option is given by:

$$u_{i0} = \gamma_0 + \gamma_1^{'} R_i + \varepsilon_{i0} \tag{5}$$

where R_i is a vector of parameters of request submitted by consumer i. These include dates of search and dates of stay, from which we can derive advance purchase length. For example, a consumer who is searching further in advance might have a higher value for the outside option, due to the possibility of searching later or they may have less flexible plans. Vector R_i also includes number of people, weekend stay, and number of days.

Taste heterogeneity in our model arises due to differences in parameters of request, as well as unobserved heterogeneity:

$$\begin{aligned} \beta_i &= \beta_0 + \beta_1' R_i + \tilde{\beta}_i & (6) \\ \ln(-\alpha_i) &= \alpha_0 + \alpha_1^{'} R_i + \tilde{\alpha}_i & (7) \end{aligned}$$

where β_0 is the vector of baseline tastes, β_1 is the matrix of interaction terms, and $\tilde{\beta}_i \sim N(0, \Sigma_\beta)$ are consumer-specific deviations from the mean.

2.1 Search Tools and Beliefs

In page t, after display $\mathbf{D}_{ti}$ is observed and clicks are made, the consumer is contemplating the next search action. At this point, there are several potential actions using search refinement tools. There are three major categories of search tools: flipping through pages of search results and sorting or filtering results according to certain product attributes such as price or distance. Within sorting and filtering, there are settings that can be fine tuned. For instance one might filter by distance to the desired landmark or filter by price band. Similarly, one might sort by distance to city center, price, star rating, etc. All of these options constitute distinct ways in which a consumer might search at the next step.[2]

For reasons that will be apparent shortly, not all search alternatives are available to each user

[2] A combination of search tools, such as "sort by distance and then filter price between 100 and 200 dollars", could in principle constitute a single alternative. However, it is impractical as the number of potential search options explodes. Thus we choose a more feasible approach, where "sort by distance" and "filter price between 100 and 200 dollars" constitute two consecutive search attempts.

on every page. Let $\mathbf{A}_{ti}$ denote the choice set of search tools available for consumer i at page t in her search sequence, and let $a \in \mathbf{A}_{ti}$ be a generic element of this set. The next display $\mathbf{D}_{(t+1)i}$, depends on the choice of $a \in \mathbf{A}_{ti}$, and generally presents a new set of results. The next display also depends on what we call a "search state", denoted by S_{ti}, which is the combination of paging, filtering, and sorting settings that are active at the time when search decision is made. For example, a search state may be as simple as {"default rank", "page 1"} or something more complex, such as {"price sorting","filter price 100 200", page 2}. The combination of current settings essentially defines the set of searchable hotels, and their ranking. Depending on the search state, the same search action may bring a different set of results. For example, if the next search action is "sort by price", then it matters if results are already constrained to be within [100,200] dollar range.

When consumer contemplates the choice of some $a \in \mathbf{A}_{ti}$, while being in search state S_{ti}, the contents of the next display are given by a mapping $(a, S_{ti}) \rightarrow \mathbf{D}_{(t+1)i}$. This mapping is deterministic, as the resulting display after action is determined by hotel availability and prices which are not affected by consumer decisions. However, the user does not know this mapping, so for him the results of search are random. For each search action $a \in \mathbf{A}_{ti}$, she forms a belief about the distribution of the next display, $\mathbf{D}_{(t+1)i}$, denoted as $H[\mathbf{D}_{(t+1)i} \mid a, S_{ti}]$. This notation highlights the fact that consumers who share the same search state should expect the same distribution of search results in response to a given search action. This fact dramatically reduces the heterogeneity among consumers with respect to benefits of search, making the search model feasible to estimate. In practice, we approximate $H[\mathbf{D}_{(t+1)i} \mid a, S_{ti}]$ by the empirical distribution of displays actually observed by consumers who have chosen search strategy a while being in search state S_{ti}.

2.2 Reservation Values

With an approximation of $H[\mathbf{D}_{(t+1)i} \mid a, S_{ti}]$, we can compute r^a_{ti}, the reservation value of consumer i associated with search strategy $a \in \mathbf{A}_{ti}$, in search state S_{ti}. This is done in a number of steps. First, we simulate a display $\mathbf{D}_{(t+1)i}$ from the empirical density $H[\mathbf{D}_{(t+1)i} \mid a, S_{ti}]$. Second, we compute maximal utility among hotels that belong to the simulated page: $\widetilde{u} = \max\{u_{ij}, j \in \mathbf{D}_{(t+1)i}\}$. By repeating these steps for $s = 1, \ldots, N$ times, we obtain $F_i[\widetilde{u} \mid a, S_{ti}]$, the simulated distribution of maximal utilities associated with search strategy $a \in \mathbf{A}_{ti}$, in search state S_{ti}. The subscript i indicates that this function depends on consumer-specific tastes that enter hotel utilities. Given a search cost c_i, the reservation value is obtained by numerically solving the equation:

$$r^a_{ti}(c_i) \quad : \quad \int_r^{+\infty} (\widetilde{u} - r)\, dF_i[\widetilde{u} \mid a, S_{ti}] = c_i \tag{8}$$

$$\widetilde{u} \quad = \quad \max\{u_{ij}, j \in \mathbf{D}_{(t+1)i}\} \tag{9}$$

This step represents the most computationally intensive part of the estimation: the inversion has to be performed for each draw of consumer tastes, search costs, and each alternative search strategy $a \in \mathbf{A}_{ti}$ available for consumer i at page t. We have developed very fast, vectorized methods of performing this computation, available from the authors upon request.

In a dynamic search model, the reservation value is determined by the distribution of continuation values associated with every search realization rather than immediate benefits of search from a better hotel on the next page. What makes a search decision dynamic is that it creates an option of searching further from a new search state. This option value is most apparent when search is ordered, as in Arbatskaya (2007), who considers search among gas stations. A driver is currently at the gas station #1, and he expects high prices at station #2 down the road, and low prices at station #3. Driving to station #2 is costly and may be worth doing not only because of potential price savings at station #2, but because of the option value of proceeding to station 3 afterwards. A similar assumption was made in Koulayev (2014) to model web search among results sorted by some criterion, such as price.

We relax the assumption that a searcher commits to a particular search strategy and allow him to re-optimize at each step. In fact, the technology of the search platform imposes little constraint on how users search: almost any search tool can be used at almost any page. Because the current choice of search tool usually does not limit future navigational options, the option value of search is likely to be small. For this reason, instead of a fully dynamic approach to search we focus only on the immediate benefits of search, as seen in equation (8). We believe that with this simplification we obtain a more computationally feasible model that can rationalize a much greater portion of search activity.

2.3 Rationalization of Search Activity

Given the diversity of search tools available on this website, the total number of combinations (a, S) is large. In practice and in our application, only a subset of potential search actions are observed with enough frequency—at least 50 times—that the empirical distribution of search results can feasibly be computed. For other combinations of (a, S) we cannot construct an adequate approximation of consumer beliefs about benefits of search. This limits our ability to rationalize the observed choices in two ways.

First, for rarely observed search states we will not be able to explain any search actions made by consumers in those states (even actions as simple as "flip a page"). Therefore all search choices made in such states will be omitted from the estimation.

Second, even in popular search states, such as {"default rank","page 1"} some search actions are rarely or never chosen. Such actions will be removed from the set of alternatives $\mathbf{A}$, available to the user in the corresponding search states. This is the primary reason why $\mathbf{A}_{ti}$ will vary across t and i.

In our application, after all exclusions our model we will be able to rationalize 56.4 percent of observed uses of search tools.[3] This activity covers 97.2 percent of search sessions. A wide coverage of search activity is important in order to simulate consumer response to a new ranking.

2.4 Search Inequalities from Search Decisions

So far we have computed reservation values associated with every search alternative available for consumer i and page t. The option of no search includes the possibility of going back and booking one of the hotels that have been previously clicked on. Let j_t^* define the choice in page t,. Therefore, the value of the fallback option at page t is the utility of the best alternative, including recalling past pages, 1 to $t-1$, reached during consumer i search activity.

$$\hat{u}_{ti} = \max\{u_{ij_1^*}, \ldots, u_{ij_t^*}\} \tag{10}$$

With this notation, we can summarize the inequality conditions related to search decisions. The

[3]This total count does not include uses of search tools that are not actually search actions because there is no uncertainty about the results of such choices. This includes typing the hotel's name and other similarly narrow filtering actions.

actual search alternatives chosen by consumer i in pages reached before t are recorded as a_{ti}, with $a_{ti} \in \mathbf{A}_{ti}$. For each page before the final one, the search decisions imply

$$
\begin{aligned}
r_{ti}^{a_{ti}} + \eta_{ti}^{a_{ti}} &\geq r_{ti}^{a} + \eta_{ti}^{a}, \forall a \in \mathbf{A}_{ti} && (11) \\
r_{ti}^{a_{ti}} + \eta_{ti}^{a_{ti}} &\geq \hat{u}_{ti} && (12)
\end{aligned}
$$

In these expressions, the terms η_{ti}^{a} are i.i.d extreme value type 1 shocks that we add to reservation values to achieve a smooth likelihood function. In the final page, it must hold that

$$
r_{ti}^{a} + \eta_{ti}^{a} \leq \hat{u}_{ti}, \forall a \in \mathbf{A}_{ti} \tag{13}
$$

In other words, the consumer chooses the search action with the highest reservation value, continues searching as long as the best reservation utility exceeds the status quo, and stops otherwise.

2.5 Likelihood Function

We seek the probability that inequalities related to clicking decisions, (1)–(3), and inequalities related to search decisions, (11)-(13), are satisfied. To ease notation, we omit consumer index i for the remainder of this section.

For estimation, we simulate the vector of utilities of clicked hotels $\{u_{j_1}, \ldots, u_{j_T}\}$, as well as consumer traits: tastes α, β, search cost c, and the value of outside option, u_0. For each draw of these parameters, additional quantities are computed: the value of status quo, $\hat{u}_t = \max\{u_{j_1}, \ldots, u_{j_t}\}$, $t = 1, \ldots, T$; and the matrix of reservation values $r_t^a = r_t^a(\alpha, \beta, c)$, $\forall a \in \mathbf{A}_t$, $t = 1, \ldots, T$, corresponding to each search alternative and each page. With simulated quantities being fixed, we integrate out the remaining unobservables from the system of inequalities, (1)–(3), (11)–(13), utility shocks related to non-clicked hotels, ε_{ij}, and alternative specific shocks related to search actions, η_{ti}^{a}. The result of the integration is as follows (see derivations in Appendix B):

$$
\begin{aligned}
L_t\left(j_t^*, a_t, \mathbf{D}_t \mid u_{j_t}, \alpha, \beta, c\right) = &\prod_{j \in \mathbf{D}_t, j \neq j_t} F\left(u_{j_t} - \mu_{ij}\right) \\
\times \frac{\exp\left(r_t^{a_t}\right)}{\sum_{a \in \mathbf{A}_t} \exp\left(r_t^a\right)} &\left(1 - F\left(\hat{u}_t - \ln\left(\sum_{a \in \mathbf{A}_t} \exp\left(r_t^a\right)\right)\right)\right), t < T \\
\times F &\left(\hat{u}_t - \ln\left(\sum_{a \in \mathbf{A}_t} \exp\left(r_t^a\right)\right)\right), t = T
\end{aligned} \tag{14}
$$

To obtain the unconditional likelihood, we multiply page-specific likelihoods and then take an average over a set of Halton draws of unobserved quantities, $\{u_{j_t}^s, \alpha^s, \beta^s, c^s\}$, $s = 1, \ldots, N$:

$$
L\left(\{j_t, a_t, \mathbf{D}_t\}_{t=1}^T\right) = \frac{1}{N} \sum_{s=1}^{N} \left(\prod_t L_t\left(j_t, a_t, \mathbf{D}_t \mid u_{j_t}^s, \alpha^s, \beta^s, c^s\right)\right) \tag{15}
$$

The left-hand side of this expression emphasizes that the likelihood function now depends only on the observed data: $\{j_t, a_t, \mathbf{D}_t\}, t = 1 \ldots T$, clicks, search actions and resulting displays in every page.

3 Platform's Problem of Optimal Ranking

We assume that the goal of the optimal ranking system we derive is to maximize the expected CTR on the organic set of search results. Although in general this goal does not coincide with profit maximization, it does in the case where all revenues are derived from click fees and the fees are constant across products. We consider click maximization an important input into the overall profit maximization problem of a platform. A ranking that maximizes CTR may serve as a starting point, which can be readily modified to satisfy various external factors, such as unequal click fees. Further, a revealed preference approach to click, as adopted here, implies that click maximization amounts to maximization of consumer satisfaction. That in itself is an important long-term goal of a search platform, often separate from the short-term goal of profit maximization. Indeed, today search platforms earn most of their revenues through clearly marked sponsored offerings, bypassing the need to distort organic search results.

We use a discrete choice modeling approach to managing expected CTR on a search platform.

After a user compares the attractiveness of the options observed during the search session, the click is an outcome of optimal consumer choice. The insight of our approach is to select a candidate ranking and use the predictions of a discrete choice model of click to calculate the probability that a random consumer will make at least one click. The discrete choice model creates a link between a candidate ranking and a click through the assumption that a click is the result of optimal consumer choice. By changing the presentation of results in a display, a candidate ranking affects the probability that the consumer's most preferred hotel will be discovered and therefore clicked.

Consumers who visit the website and interact with the results of the request differ in unobserved ways, leading to different responses to the ranking. This represents a major challenge for the model of optimal ranking. Generally, there are two dimensions of unobserved heterogeneity. The first is of course unobserved tastes for a hotel's location, quality parameters, and price sensitivity. Of course, it is possible to ignore the unobserved component and simply offer a ranking that targets an average consumer, but this results in a loss of efficiency, as we will demonstrate later. The second dimension of heterogeneity is related to search preferences. Indeed, the ranking must be formed right after the search request is made and before any search actions are observed. Because future search actions will distort, or even eliminate, the effects of recommended ranking, a model of optimal ranking should anticipate how a consumer will continue searching after the initial display. Furthermore, the search decision itself is endogenous to ranking: for instance, if a consumer finds an attractive option on the initial display, she may decide to stop searching, eliminating potential future clicks.

The model of consumer search and click presented above addresses these two dimensions of consumer heterogeneity. By modeling the click decision structurally, we are able to express it in terms of consumer primitives, such as the distribution of unobserved tastes. Therefore we relate search actions to the contents of displays that are formed by the recommended ranking. For both reasons, we use the search model to evaluate various design approaches to a recommendation ranking.

3.1 Recommendation Ranking

The objective of the recommendation ranking is to manipulate displays of hotels observed by the user: $\mathbf{D}_t,\ t = 1, \ldots, T$, in a way that maximizes the expected click rate, and is computed before any search actions are taken. The initial display seen by a user, $\mathbf{D}_1$, consists of the top 15 hotels ranked by the recommendation system. Beyond $\mathbf{D}_1$, the effect of ranking on the composition of display is influenced by the use of filtering and sorting actions. Sorting hotels by dimensions such as price makes recommendation rankings less effective. Filtering preserves the ranking but reduces the set of hotels that can be recommended. The greatest impact of the recommendation ranking is on passive users, those who only evaluate the initial page of default results and do not engage in search actions, either by flipping through other pages of results or sorting or filtering the search results.

The interference of user search actions with recommendation rankings means the CTR achieved by a candidate ranking will vary greatly across future search paths. Search actions have to be integrated out (simulated) in order to compute the expected click rate under a candidate ranking as future search actions are unknown at the time a ranking needs to be presented. To make accurate predictions, a structural search model must be able to differentiate between a wide array of search alternatives available to the user at every step. For instance, a model that focuses on a few dominant search strategies will make inaccurate predictions among the long tail of consumers who employ rare search strategies. We achieve a wide coverage of search strategy space by allowing the consumer to re-optimize after every display.

Formally, the expected CTR is computed as follows. Let $\mathbf{r} = (r_1, \ldots, r_N)$ be the candidate ranking, and $\tilde{\mathbf{a}} = \{\tilde{a}_1, \ldots, \tilde{a}_T\}$, $\tilde{a}_t \in \mathbf{A}_t$ be a potential search path, with a tilda sign to differentiate from the actual choice observed in the data. Then, the probability that a random user would click on hotels indexed by $\{\tilde{j}_1, \ldots, \tilde{j}_T\}$ is obtained by integrating search paths out of the likelihood function (15):

$$\begin{aligned} P\left(\{\tilde{j}_1, \ldots, \tilde{j}_T\} \mid \mathbf{r}, P, R\right) &= \sum_{\tilde{a}_1, \ldots, \tilde{a}_T} L\left(\{\tilde{j}_t, \tilde{a}_t, \mathbf{D}_t\}_{t=1}^{T}\right) \\ \mathbf{D}_t &= \mathbf{D}_t(\mathbf{r}, \tilde{a}_t),\ t = 1, \ldots, T \end{aligned} \tag{16}$$

The expression $P\left(\{\tilde{j}_1, \ldots, \tilde{j}_T\} \mid \mathbf{r}, P, R\right)$ underscores that the probability of click depends on three factors: ranking system $\mathbf{r} = (r_1, \ldots, r_N)$, current price vector $P = (p_1, \ldots, p_N)$ and parameters of request, R. Because all these quantities are known, the expression (16) is computable. The function $\mathbf{D}_t = \mathbf{D}_t(\mathbf{r}, \tilde{a}_t)$ underscores that the effect of ranking on click probability works through displays, which also depend on search actions, as discussed above.

The formula (16) can be used to select candidate rankings $\mathbf{r}$ that maximize the expected CTR rate on particular hotels (i.e., if revenue per click for these hotels is higher). In our application, we focus on aggregate CTR which is the probability that any hotel is clicked by a random user. Therefore, we integrate identities of clicked hotels out of (16), subject to a condition that at least one $\tilde{j}_t > 0$ (notation $\tilde{j}_t = 0$ corresponds to a "click" on the outside option, or non-click). We also integrate over various search lengths, with $T \leq \bar{T}$. This step gives us the objective function for the optimal ranking which maximizes expected CTR:

$$E[CTR(\mathbf{r}, P, R)] = \sum_{T=1}^{\bar{T}} \sum_{\tilde{j}_1, \ldots, \tilde{j}_T} P\left(\{\tilde{j}_1, \ldots, \tilde{j}_T\} \mid \mathbf{r}, P, R\right) \longrightarrow \max_{\mathbf{r}} \tag{17}$$

Solving this optimization problem head-on is infeasible because there are $N!$ potential rankings, where N is the total number of hotels.

Consumer Types and Targeted Rankings

Given the potentially large number of rankings, our approach to select a ranking for a particular consumer is to create a discrete number of consumer types, G. This is achieved by discretizing the distribution of random consumer tastes for hotel characteristics, which are α_i and β_i (see (7) and (6)), into a grid with pre-defined points. The price sensitivity parameter is defined as: $\ln(-\alpha_i) = \alpha_0 + \alpha_1^{'} R_i + \tilde{\alpha}_i$, where $\tilde{\alpha}_i \sim N(0, \sigma_\alpha)$. With an estimated value of σ_α, we compute a set of quantiles, $\{\alpha_0^1, \ldots, \alpha_0^{G+1}\}$, which covers 99 percent of the density. The grid $\{\alpha_0^1, \ldots, \alpha_0^{G+1}\}$ leads to a finite set of consumer types: a consumer i is said to be of type $g(i)$ if his draw of $\tilde{\alpha}_i$ belongs to an interval $(\alpha_0^g, \alpha_0^{g+1})$. The price sensitivity of a type g is approximated as the midpoint of this interval. A grid can be constructed for the joint distribution of β_i, but for the sake of clarity of exposition we will use the mean, $E[\beta_i] = \beta_0 + \beta_1' R_i$.

As a result of these transformations, we replace the actual mean utility of consumer i:

$$\mu_{ij} = \alpha_i p_{ij} + \beta_i^{'} X_j \tag{18}$$

with its approximation,

$$\hat{\mu}_{ij} = \alpha^{g(i)} p_{ij} + E[\beta_i]^{'} X_j \tag{19}$$

The advantage of the approximation is that now consumer tastes for hotel characteristics are summarized by just two parameters: $\alpha^{g(i)}, R_i$. In other words, if we knew these parameters for an new incoming consumer to the platform, we could compute a vector of mean utilities $\hat{\mu}_{i1}, \ldots, \hat{\mu}_{iN}$ for that user. Once mean utilities are known, the click maximizing ranking is simply to sort hotels by decreasing mean utility. An important fact is that the hotel's position on the click, L_{ij}, does enter these mean utilities because we interpret its effect as being related to the within-page cost of search, rather than an inherent part of user preferences.

As consumer's type $g(i)$ is unobserved, it is not possible to directly offer an optimal ranking to each consumer. Instead, our objective is to "guess" the type of the current user, in a way that maximizes the expected click rate, where expectation is taken across unobserved types. With this approach, a generally intractable problem (17) is reduced to a discrete optimization problem:

$$E[CTR(\mathbf{r}^{\tilde{g}}, P, R)] = \sum_g \phi_g \left[\sum_{T=1}^{\bar{T}} \sum_{\tilde{j}_1, \ldots, \tilde{j}_T} P\left(\{\tilde{j}_1, \ldots, \tilde{j}_T\} \mid \mathbf{r}^{\tilde{g}}, P, R, \alpha^g \right) \right] \longrightarrow \max_{\tilde{g}} \tag{20}$$

where $\mathbf{r}^{\tilde{g}}$ is a ranked set of mean utilities corresponding to "guessed" type $\tilde{g}$. The outer sum represents an integration of CTR over unobserved consumer type, g, which is different from the "guessed" type. The probability of each type, $\phi_g = P(\tilde{\alpha}_i \in (\alpha_0^g, \alpha_0^{g+1}))$ is computed using estimated density $\tilde{\alpha}_i \sim N(0, \sigma_\alpha)$.

From the platform's perspective, the solution of (20) is relatively fast because rankings corresponding to each consumer type (e.g. combination of $\alpha^{g(i)}, R_i$) can be pre-computed before a consumer arrives to the platform. Rankings need to be recomputed when prices change. This ensures speed and scalability of computations, which are critical properties of the recommendation system.

Below we present a few simple examples that help develop some intuition behind the choice of optimal ranking. One may wonder, if we do not know the consumer type, why do we not cater to the average type? Alternatively, what about the dominant (most popular) type? Also, what is the role of search actions? How is prediction of future search actions affecting our choice of ranking?

3.2 Recommended Ranking Two-product Example

We propose a couple of simplified examples to illustrate two key issues that a ranking must address: heterogeneity in unobserved tastes and differences in ranking when users employ sorting and filtering tools. These examples allow us to develop intuition as to why and how a ranking that targets a median consumer will not achieve the same results as a targeted ranking based on distributional parameters. The intuition is as follows: if we have any reason to believe that hotel A will have a higher click rate than hotel B by the current user, we maximize the overall click rate by pushing hotel A to the first page instead of hotel B. Our assessment of click rate is based on observable hotel characteristics, which implies that hotels with higher mean utility are more likely to be clicked.

Consider the next setup as a simplified version of the search environment faced by consumers. Suppose there are two products, characterized by price-quality combinations (p_j, v_j), $j = 1, 2$. Product 2 has higher quality than product 1 (i.e., $v_2 > v_1$) but it is also more expensive ($p_2 > p_1$). Product utilities are $u_{ij} = \alpha_i p_j + v_j + \varepsilon_{ij}$. Consumers can be of two types, which are unobserved, with type 1 being more price sensitive: $\alpha_1 < \alpha_2$. It follows that if the price of the high quality product is not too high, price sensitive consumers prefer the lower quality product. Finally, the share of type 1 in the population is θ, and the share of type 2 is $1 - \theta$.

Upon arrival to the website, all consumers observe the first display, which contains one product, that was ranked highest by the recommendation system. To find the second product, they need to search. With probability π the user will continue searching to discover the second-ranked product, and with probability $1 - \pi$ he will not search. Depending on his search decision, a user may have only one or two products in his choice set. With only one product the click rate of a consumer with $\alpha = a_1, \alpha_2$ for option j is the logit formula:

$$C_j(\alpha) = \frac{\exp(\alpha p_j + v_j)}{1 + \exp(\alpha p_j + v_j)}, \ j = 1, 2$$

The expected click rate for such consumers is $\bar{C}_j = \theta C_j(\alpha_1) + (1-\theta)C_j(\alpha_2)$. If both products are found, the click rate is higher at

$$C_{12}(\alpha) = \frac{\exp(\alpha p_1 + v_1) + \exp(\alpha p_2 + v_2)}{1 + \exp(\alpha p_1 + v_1) + \exp(\alpha p_2 + v_2)}$$

Similarly, the expected click rate among those with two products is $\bar{C}_{12} = \theta C_{12}(\alpha_1)+(1-\theta)C_{12}(\alpha_2)$.

Now we examine the properties of optimal ranking in this setup.

Comparison of Rankings under Random and Average Tastes

Given that consumer types are unobserved, it is not a priori clear how such information could be incorporated into a ranking and why such a strategy may be better than a ranking based on average tastes. The following discussion emphasizes two reasons. First, the click rate of the average type does not equal the expected click rate across types since the expected click rate is a non-linear function of consumer type (the price sensitivity parameter in our case). Second, price changes between the product options affect the CTR differently across types. We show that depending on the relative price levels, the discrepancy from average type ranking and optimal ranking can vary between zero to something substantial. There is simple economic intuition behind both effects, which we illustrate below.

With only two products, one per page, the choice of ranking is between $\{1,2\}$ and $\{2,1\}$. With search intensity π, click rates under these alternative rankings are:

$$CTR(1,2) = (1-\pi)\bar{C}_1 + \pi\bar{C}_{12} \tag{21}$$

$$CTR(2,1) = (1-\pi)\bar{C}_2 + \pi\bar{C}_{12} \tag{22}$$

Indeed, with probability $(1-\pi)$ the user observes only the top ranked product, and with probability π, both products are discovered. The ranking $\{1,2\}$ is optimal if it leads to higher expected CTR:

$$(1-\pi)\bar{C}_1 + \pi\bar{C}_{12} > (1-\pi)\bar{C}_2 + \pi\bar{C}_{12}$$

or simply

$$\bar{C}_1 > \bar{C}_2 \tag{23}$$

Now consider a ranking that targets preferences of the average consumer, whose price sensitivity is $\bar{\alpha} = \theta\alpha_1 + (1-\theta)\alpha_2$. It is straightforward to show that ranking $\{1, 2\}$ is optimal if:

$$C_1(\bar{\alpha}) > C_2(\bar{\alpha}) \tag{24}$$

or simply

$$\bar{\alpha}p_1 + v_1 > \bar{\alpha}p_2 + v_2$$

In other words, targeting the average consumer type amounts to sorting results by declining average utility.

The ranking rules (23) and (24) will generally be different, as the click rate is a non-linear function of unobserved parameter α, so that $\bar{C}_1 \neq C_1(\bar{\alpha})$. However, the magnitude of the discrepancy will vary with product prices, both in absolute and relative terms. To illustrate this point, Figure 1(a) plots ranking rules (23) and (24) in the space of product prices, p_1, p_2.[4] For each rule, there exists a boundary such that for all price points (p_1, p_2) above it the low quality product (product 1) should be placed on top, and vice versa. For instance, at point B both rules will place the expensive product on top. The shaded area is where the recommendations of ranking rules (23) and (24) disagree. In the area with point A, the ranking that targets average tastes will place the high quality item (product 2) in the top position while the customer targeting favors the cheaper product. In the area with point C, the opposite is true.

The reason for disagreement between average and random-taste targeting is the following: the total expected CTR under any ranking is composed of clicks made by consumers who belong to various unobserved types. The contribution of a particular consumer type is determined by its population share (which does not change with prices) and its type-specific expected CTR (which depends on prices). As prices change, so do relative contributions of consumer types, and the optimal ranking must adjust accordingly. For instance, if prices of both products are high, as in point C, the contribution of the price-sensitive type becomes so small that it is optimal to target the high type, even though it is less prevalent in the population. The ranking that targets average type does not detect these shifts in expected CTR composition, and can make erroneous

[4] For this and other figures, we adopt the following parametrization: $v_1 = 10$, $v_2 = 20$, $\alpha_1 = -2$, $\alpha_2 = -1$, $\theta = 60\%$.

recommendations (as in point C where it continues to place the cheaper product on top). The resulting click losses can be quite large depending on price levels.

Ranking under Search Refinement

Limited consumer search is the primary reason for the need for recommendation systems. Furthermore, consumers are very different in their use of search tools. Some do not search at all, while others use filtering and sorting tools to explore additional options in the results. Chen and Yao (2012) find that consumers use these search refinement tools disproportionately if they are uninformed of the default ranking rule. As a result, the same ranking will have different impacts on consumers who search differently.

An ideal recommendation system will choose the contents of the first page in a way that complements future search results. For users who sort by ascending price, one might show more expensive hotels on the first page, thus increasing the diversity of their choice sets. Unfortunately, individual search actions are unobserved at the time when a ranking is to be made. Instead, we observe search actions made by past consumers and their frequencies, much in the same way as we infer consumer types and their probabilities. Such distributional information can be a valuable input into the ranking method.

Returning to our basic setup, we assume that with probability $\pi_{Rec} = 50\%$ the user will search the default recommended options and discover the second-ranked hotel; with probability $\pi_{Price} = 20\%$ the user will sort by price and discover the cheapest hotel. Depending on the ranking method, these strategies may lead to different choice sets. The expected CTR under the ranking $\{1, 2\}$ is:

$$CTR(1,2) = (1 - \pi_{Rec} - \pi_{Price})\bar{C}_1 + \pi_{Rec}\bar{C}_{12} + \pi_{Price}\bar{C}_1 \tag{25}$$

The first part is expected click rate by consumers who did not search: their choice set consists of hotel 1. The second part is expected click rate by those who searched default results and so found both hotels. Finally, there are consumers who sorted by price and did not find any new results, as the cheap product was already on the first page. The CTR under $\{2, 1\}$ ranking is obtained similarly. Taking the difference,

$$CTR(1,2) - CTR(2,1) = (1 - \pi_{Rec} - \pi_{Price})(\bar{C}_1 - \bar{C}_2) - \pi_{Price}(\bar{C}_{12} - \bar{C}_1) \tag{26}$$

If $\bar{C}_1 > \bar{C}_2$ then as we know from the previous example, in the absence of price sorting, it would be optimal to place that product on top. Indeed, in that case the expected click rate by non-searchers will increase by $(1 - \pi_{Rec} - \pi_{Price})(\bar{C}_1 - \bar{C}_2)$. However, with an option of price sorting, there is also a negative effect: consumers who search by price will lose $\pi_{Price}(\bar{C}_{12} - \bar{C}_1)$.

Figure 1(b) illustrates this tradeoff by comparing the optimal ranking rule (26) with a ranking that ignores the presence of the price sorting tool.[5] At point A, both rules prescribe to place the higher quality product on top, because it is not too expensive. The dotted area corresponds to product prices when the prescriptions of the two rules diverge. At point B, the optimal ranking rule (26) still places the higher quality product on top, while the simpler rule dictates that this product is now too expensive and should be downgraded, because now $\bar{C}_1 > \bar{C}_2$. The fact that a share of consumers who always discover the cheap product by price sorting leads to higher prominence of the expensive product under the optimal rule. Another less obvious factor is that some consumers will erroneously use the price sorting tool and may not find their preferred high quality product if it is not placed on top.

4 Empirical Analysis

In this section we present an empirical application of the proposed model using a dataset of search actions and consumers' choices in an online travel platform. After an initial request, a consumer is presented with a default display of hotel alternatives and tools to sort and filter the results. A consumer's choice is represented by a click on one of the hotels and indicates that the clicked hotel is more attractive than other hotels considered in the search process. This assumption is corroborated by previous studies, such as Joachims et al. (2005) and Brynjolfsson and Smith (2002) who compared click choices to actual purchases and found that clicks provide a reasonable proxy for the preferences of an average consumer.

[5] We adopt parametrization: $\pi_{Rec} = 50\%$, $\pi_{Price} = 20\%$, in addition to $v_1 = 10$, $v_2 = 20$, $\alpha_1 = -2$, $\alpha_2 = -1$, $\theta = 60\%$.

4.1 Data

The data consist of consumer search histories for hotel rooms on a major web-based hotel comparison platform. A search history starts with a search request. The request includes city, dates of stay, number of people, and number of rooms. Following the request, the website presents the consumer with an ordering of hotels in the city with available rooms that satisfy the request parameters. The hotel options are organized in pages with 15 hotels per page, displaying hotel name, price, as well as various non-price characteristics: star rating, neighborhood, distance to city center, proximity to airports, and so on. All consumers are presented with a first page of results upon submitting the search request.

After obtaining the default set of results, consumers can either click on a hotel on that page, continue to the next page of results, or employ other search refinement tools such as sorting or filtering results based on hotels' characteristics. In our data, we observe the listing of all hotels displayed as a result of the consumer's search request and all subsequent sorting or filtering actions. We also observe which hotel the consumer clicked as part of their search process. About half of searchers who click do so only once, in which case this is the end of the search session. If more than one click is made in one page, we consider the last click as the hotel that is the closest match to the user's preferences among the searched hotels. Alternatively, the user can leave the website without clicking.

The platform is a search aggregator, which compiles and presents hotel information from other travel websites, but it does not offer hotel bookings. After clicking on a hotel option, the user is redirected to another website where a booking can be made. A potential limitation of the data is that we do not observe whether two sessions were made by the same person (a session is a set of search requests within 24 hours of each other). Therefore, we conduct our analysis as if every search were made by a unique individual. Below we discuss these issues in greater length.

The sample contains 23,959 unique search histories for Chicago, by consumers who searched the platform during May 1–31, 2007. There are 148 Chicago hotels available for online booking during that period. A search typically returns 130 to 140 hotel options, depending on availability. This wealth of lodging options creates a non-trivial search problem for a consumer. Hotels in the Chicago market include those in the city of Chicago itself, the satellite towns (Evanston, Skokie, etc.), as well as those in close proximity to airports (O'Hare and Midway). There are also various neighborhoods within the city center. These geographical parameters are observed by consumers during search.

Table 1 summarizes our data. On average, consumers in our sample search for a hotel room 33.5 days in advance of their stay, 60 percent of them stay over a weekend, and the average number of guests is 1.84. The hotels in the dataset show significant price variation. The range of prices in the sample is from $16 to $1,500, with an average of $230. This reflects both cross-sectional as well as temporal variation in prices. The variation in contents of the first page of results can be summarized by the percentage of observations in which a given hotel appeared on the first page. In the sample, an average hotel appears on the first page about 30 percent of the time, with substantial heterogeneity: some hotels never appear on the first page, while others appear there every second search or more.

In the sample, 33 percent of the search requests ended in a click. Of these clicks, more than half took place on the first page. An individual hotel's CTR ranges between 1.1 and 5.8 percent. Prices of hotels that received a click are somewhat lower, which is expected, as clicks can be considered revealed preferences. There is considerable variation in search behavior among consumers, which takes the form of a long tail on hotel choices. An average consumer observes two pages of results (30 hotels). Table 2 presents the breakdown of the different search refinement strategies of the consumers in the sample. We find that 40.8 percent of the consumers never search—they only observe the first page of results. Another 16.8 percent of visitors continue browsing past page 2 of the hotel listings. Among all consumers, 42.4 percent choose a search refinement action, and contribute a disproportionately higher share of clicks, 48 percent.

Finally, Figure 2 illustrates the extent of within-page search. Although we do not observe this search directly, indirect evidence can be obtained from the distribution of clicks across hotel position (ranging from 1 to 15 on the page of results). Of all clicks on the first page, 22 percent went to the hotel in position 1, in contrast with 11 percent to position 2, and 3 percent to the last position on the page. A similar pattern exists for other pages: overall 19 percent of the clicks were for a hotel located at the top of any page.

4.2 Empirical Strategy

In this section we outline the empirical strategy to estimate the model of search and click. There are three types of unknown parameters: a) baseline consumer tastes for hotel characteristics, including variances of error terms (equations (6) and (7)); b) the effect of request parameters R_i on the probability of non-click and on consumer tastes for hotel characteristics (equation (5), (6) and (7)); and c) mean and variance of the search cost distribution, which is assumed to be log-normal.

Identification

Utility parameters. The browsing and clicking data contains rich sources of variation that can be used to identify consumer preferences for observable hotel characteristics, as well as the distribution of unobserved heterogeneity. These sources are: changing hotel prices, composition of choice sets, and positions of hotels within the display. In our data, no two users observed the same vector of hotel prices and had the same composition of choice set. In other words, every click we observe was made in a different choice context than the next click. This helps identification because we can fix a subset of hotel characteristics (such as neighborhood and quality rating) and still find variation in other parameters (such as price or brand). Because we directly observe the choice sets (e.g., sets of hotels displayed on the screen), we improve the precision of estimates by avoiding ad-hoc assumptions on the distribution of choice sets.

Price sensitivity. Identification of the price effect on hotel click utility represents a particular challenge, because price is the only time-varying characteristic of a hotel, which can be correlated with changes in unobserved hotel quality over time. For instance, a hotel could be located near a sports event—a positive demand shock that drives up both price and demand. To isolate the effect of such unobserved shocks on price variation, we use lagged prices of the same hotel to predict the average hotel's price in the current week; in the estimation, we use the predicted price instead of the actual price that was shown to the consumer. This approach removes the effects of temporal price shocks that are likely driven by transitory shocks to demand. To account for time trends in demand, we include month indicators. A similar technique has been applied by Villas-Boas and Winer (1999), Hausman (1996) and, in the context of hotel demand, by Ghose et al. (2012).

Position effects. The position of a hotel within a display has an important effect on the probability of click, due to the human tendency to read results from top to bottom. However, if the page is sorted by default ranking, which is based on some measure of hotel popularity, the position may also reflect unobserved hotel quality, and for this reason there may be a spurious correlation with click rate.

In the data, about 57 percent of clicks are made on pages sorted by default ranking. To correct for the endogeneity of position on default ranked pages, we construct a "predicted" position of a hotel using hotel's fixed effect, price, and observed click rate over the past two weeks. In a sense, this is an attempt to reverse-engineer the default ranking. This prediction tells us what the position of a hotel should be given the demand for that hotel in the immediate past. Any residual difference between the actual and predicted position, then must be due other factors, such as changes in availability of other hotels. For these reasons, we use the residual value of position in the estimation.

The remaining 43 percent of clicks are made on pages resulting from a search action, such as price, and for these clicks the position effect is exogenous. For instance, if a page is sorted by price, and we already control for price in the click utility, then the only reason the hotel's position is correlated with price is because of the actual position effect, and not due to spurious correlation. The joint variation of positions and identities of clicked hotels in this sub-sample is an important source of identification of the position effect and is complementary to the residual-based approach outlined above.

Search costs. The main challenge in identifying search costs is to separate them from preferences in their effect on search decisions. As we observe both search actions and clicks, the identification is straightforward: while preferences affect both search actions and clicks, search costs affect only search actions. The joint variation in clicks and choice sets allows us to pin down a distribution of preferences; conditional on that, observations of search intensities uncover the search cost distribution. More specifically, because our data comes in the form of conditional search decisions—search actions coupled with preceding displays of results—a search cost distribution is identified.[6] The idea is that consumers who had better fall back option, for instance, better hotel deals on the first page, will search less frequently than those who were not as lucky. The difference between search intensities in two groups is regulated by the parameters of the search cost distribution. The

[6] See Koulayev, 2014 for a formal discussion.

identifying assumption is the contents of the first page, or other displays preceding a search action, are not correlated with preferences in a conditional sense (e.g., after controlling for observables, such as parameters of request, prior search actions, etc.).

4.3 Estimation Results

In this section we present estimates of the search and click model under different specifications: 1) logistic regression; 2) logit model with constant tastes; 3) logit with random tastes; and 4) search and click model. The main distinction is that the first three model specifications explain clicks conditional on search actions, while the search model incorporates search actions and clicks as dependent variables.

Table 3 presents utility parameter estimates of these four discrete choice models. In every specification we include a rich set of interactions between consumer level observables—such as advance purchase, duration of stay, weekend stay and number of travelers—and both price and non-price hotel characteristics. The logistic and logit model with random tastes present similar results. Specifically, the least price sensitive are consumers who search two weeks in advance and include a Saturday night stay, while (perhaps contrary to expectation) the closer the arrival date is, the more price sensitive is demand. This implies that request parameters contain valuable information about consumer preferences that should be used by ranking systems. The estimates of the price coefficient interactions of the random tastes model are larger than the constant taste model, which explain the positive mean price coefficient. The extent of unobserved heterogeneity is substantial, as captured by the standard deviation of the marginal effect of price is 0.81, which is 2 times the mean.

Table 3 also shows that the estimates of the mean price and the interaction with the request parameters are substantially larger in the search and click model than in the alternative models. These findings suggest that elasticity estimates might be biased when search information is ignored and only click information is used to characterize choice. Intuitively, when consumers are allowed to face alternatives exogenously chosen, estimates will attribute the choice of pricier alternatives to price sensitivity instead of to optimally deciding to stop engaging in costly search. An interesting finding that supports this notion is the sign reversal of the coefficient for consumers who search less than two weeks in advance, indicating that consumers are significantly less price sensitive when facing time or deadline constraints. The estimated distribution of search costs is presented in Table 4. Search costs are close at the lower end of the distribution, around \$10. The median search cost is \$20.5.

4.4 Model Fit

In order to test the fit of the search model, which accounts for search in addition to click choices, we augment the non-search models with empirical frequencies of search actions. This enable us to construct the likelihood of the observed data (search and clicks) and compare it to the full search model. Specifically, we multiply the likelihoods by an estimate of search action probabilities using the frequency of a search action in the data. As search and click decisions are independent the effect of adding search decisions to click models is to add a constant to the log-likelihood (LL). Table 3 presents the comparison of LL of the different models. The *LL(search and click)* indicates that the search model significantly outperforms any of the non-search models (chi-square test of log-likelihood differences). The resulting differences in LL between the search model and any of the click models are so large that they are statistically different at any level of confidence.

Another alternative to compare the performance of the search and the logit models regarding their abilities to predict clicks is to compute the expected probability of the observed click decisions, where search decisions are integrated out. In the search model, we sum up probabilities of the joint search and click decisions across potential search paths to estimate the LL only of click decisions. In the other specifications, we compute a weighted average of click probabilities, where weights are sample frequencies of search paths. The result is on the LL (click) in Table 3. A natural benchmark for these values is the average click rate in the data: 35%. This is the probability that a randomly chosen consumer will click at least once. Taking the logarithm of 35% and multiplying by the number of observations, we obtain $LL(click) = 22,777$. In all logit models, the $LL(click)$ is lower than this benchmark, which is to be expected because these models use a set of covariates to explain clicks. Comparing search and logit models, we again find that the differences in the outcome of search and click models are substantial, when it comes to predicting clicks.

Intuitively, the reason for the search model being able to explain clicks better is that the search model is using inequalities imposed by search decisions to update expectations of the distribution of consumer tastes. In this way, the search model brings an additional source of information. In other words, the result that the search model can better explain clicks is simply a testament to the fact that search decisions are actually informative about consumer preferences.

4.5 Targeted Ranking

In this section we present estimates of the methodology to present an individual consumer with a targeted ranking using the information available to the platform about consumer preferences contained in their request parameters, R_i. If consumers are ex-ante anonymous, the platform's problem simplifies to presenting a unique optimal ranking based on average consumer tastes. In this section we offer an illustration of targeting that relates request parameters to consumers' heterogenous price sensitivities. However, this targeting can be extended to a richer set of consumer characteristics.

As a first step we construct 40 consumer types by discretizing the distribution of unobserved heterogeneity, corresponding to different values of parameter α_{0i}, which stands for the unobserved component of the price sensitivity (see equation (7)). These values represent the grid points over the interval containing 99 percent of the probability mass of α_{0i}. For an economic interpretation, we compute price marginal effect for each type: the percentage point change in the expected CTR following a 1 percent increase in prices of all hotels. Figure 3(a) plots the resulting density of marginal effects: the median is around -1 percent, with substantial amount of heterogeneity around the median.

As discussed previously, the observable consumer-level heterogeneity in our sample consists of parameters of request, R_i, and a vector of prices, P_i. For every combination of (R_i, P_i) observed in the sample and possible unobserved type α_0^g, $g = 1, \ldots, 40$, we construct a ranked list of hotels, denoted by vector $\mathbf{r}_i^g$, where each ranking is optimal conditional on the underlying unobserved type. For every ranking $\mathbf{r}_i^g$, we simulate counterfactual choice sets that would have been realized under this ranking. For every simulated choice set, an expected CTR is computed using the model's estimates. A ranking that delivers the highest CTR corresponds to a certain "guessed" consumer type $g_i^* = g^*(R_i, P_i)$, called the "best guess," which is the solution to the platform's optimization problem. The notation $g^*(R_i, P_i)$ implies that the "best guess" will vary with prices and parameters of request.

Repeating the procedure for every observation, we obtain a sample distribution of "best guesses,"

g_i^*, $i = 1, \ldots, M$, shown in Figure 3(b). The median type is most frequently chosen as the "best guess," but only in 30 percent of cases. In the remaining 70 percent of cases, types other than median are optimal for targeting through a customized ranking. In other words, in the majority of cases a simple targeting of a median consumer leads to sub-optimal ranking.

A further insight is obtained from Figure 4, which illustrates the effect of request parameters on the optimal group type that a consumer is assigned to. The figure shows the distribution of optimal "guesses" among users with different request parameters. The left figure shows two hypothetical requests: Guest 1 books less than a week in advance and stays one night over a weekend, Guest 2 books more than a week in advance and stays 3 or more nights with no weekend stay. These two hypothetical guests represent the two extremes of the price sensitivity distribution. The distribution of optimal group types differs greatly among them. Another example is presented in the right figure. It shows the typical business and leisure guests. The business guest travels alone, stays 3 or more nights and does not stay over a weekend. The typical leisure guest books a room for two adults, spends one night, and includes a weekend stay. The figure shows that the most likely ranking of the typical business guest is the one for group 15 and the ranking for typical leisure guest is the ranking for group 17.

5 Evaluation of Ranking Methods

In this section we further explore the validity of our estimates by evaluating different recommendation rankings. The preferable method to evaluate the performance of a ranking method is through a randomized field experiment. In the absence of such an opportunity, we use the structural model of search and click to compare rankings. To the extent that the model adequately reflects the data, and is identified, the model's predictions can inform us about the relative performance of alternative ranking methods. These results can be employed for a better selection of recommendation rankings to evaluate in field experiments, which are costly to implement.

We evaluate recommendation rankings at different ends of the recommendation spectrum, from a simple popularity ranking to our proposed ranking that maximizes click rate in a fully structural model of search and click. In particular we separate the importance of accounting for different aspects of the model: unobserved taste heterogeneity, endogeneity of search decisions to ranking, and targeting using parameters of requests. Additionally, using our model of search and click, we evaluate the relative contributions of these factors to the performance of a ranking, and in this way help managers make better design choices.

Next, we construct two measures of performance of a ranking that this model generates. The first measure is the expected click rate, predicted by the model of search and click. Because the first measure is model-based, this measure properly accounts for unequal click likelihoods and endogenous search response. The second measure is the average rank of hotels that were clicked. As the second measure is model-free, the results of both metrics are not necessarily be the same and hence can be viewed as complementary.

5.1 CTR Comparisons

In order to estimate CTR for different rankings we proceed in two steps. In the first step, we use estimates of the full model of search and click to construct a ranking that maximizes the predicted click rate in that model. In the second step, we build several simpler, or more limited, models that do not take into account some choice factors, such as request parameters, random tastes, and endogenous search decisions. For each model, we construct a ranking that maximizes CTR as predicted by that model, just as a practitioner would do. Then we use the full model to compute CTR for each alternative ranking. Because the full model yields maximum CTR, any alternative ranking would necessarily result in a lower CTR than the fully optimal ranking derived in the first step.

To compare the performance of the different rankings mentioned above, we also simulate two rankings commonly used in recommendation systems: popularity and price sort rankings. The popularity ranking is defined as $r_j = clicks_j/impressions_j$. We then calculate CTR as the expected click rate integrated over future search paths and future clicks, at most one click per page.

Table 5 presents the average CTR for the default ranking, the simulated rankings, and the rankings from the click logit models and the search and click model. The averages are taken over a random sample of 100 ranking scenarios. These scenarios are combinations of parameters of request and hotel prices available at the time. Under a given ranking, CTRs will generally vary across scenarios. For instance, in periods when hotel prices are higher, CTR is lower.

The default ranking is the worst performer of all the rankings: CTR is on average is 0.105. In contrast, the CTR of the simulated popularity ranking is 0.132. The price sort ranking outperforms the default and popularity rankings with an average CTR of 0.157. The rankings from the structural models significantly outperform the default ranking: the average CTR for the constant coefficient ranking is 0.185, 0.192 for the ranking with random coefficient, and 0.203 for the search and click model.

It is important to emphasize the incremental gains, in relative terms, of the different models as CTRs in online environments are typically small. The average CTR ranges between 0.105 and 0.203 for the default ranking and the ranking from the search and click model, respectively. Although this is a twofold increase in CTR, as seen above most of the gains can be achieved with simpler rankings. For instance, the CTR of the price sort ranking is on average 0.158, or (0.158-0.105)/(0.203-0.105) = 54 percent of the average CTR gains brought about by the search and click model.

Average CTR gains from the random coefficient logit model are 89.6 percent of the average CTR gains from the search and click model. These averages mask much larger differences between the two models that are found in the lower tail of the CTR distribution. To illustrate this point, Table 5 also presents the average CTR for each decile of the CTR distribution. For the first decile, the average CTR for the search and click model (0.171) is 130 percent of the CTR of the default ranking (0.171). In this decile, the price sort ranking only realizes (0.103-0.074)/(0.171-0.074) = 28 percent of the average CTR gains brought about by the search and click model. In turn, average CTR gains from the random coefficient logit model represent 73 percent of the average CTR gains from the search and click model.

Figure 5 illustrates these distributional differences between the rankings. Figure 5(a) shows that the simple simulated price sort ranking distribution of CTRs dominates that of the default and popularity rankings. In Figure 5(b) illustrates that overall the largest gains in CTR are brought about by the utility-based rankings. Figure 5(c) illustrates that the incremental CTR gains of the search and click model are mainly at the lower end of the distribution.

To summarize, we find that simpler utility-based rankings obtained from a click-only model are very effective, with average performance in the proximity of the first best (from the model of search and click). However, there are also two important benefits of the search and click models. First, the search and click model is still 1 to 2 percentage points ahead of the click-only models, which in absolute terms is a large amount when it comes to click optimization. For a modest-sized online search platform with 1 million visitors a month, this will result in 10,000 to 20,000 extra clicks relative to the random coefficient model. Second, and perhaps more important, the CTR gains might be larger for some groups of users than the average suggests. For this reason, it is important to consider the distributional impact when judging the performance of a recommendation ranking. In this metric, the search and click model performs much better than click-only models, as found in the evidence from quintiles of CTR.

5.2 Rank of Clicked Hotel Comparisons

In this section we compare the average rank of hotels that were clicked under different ranking methodologies. This metric is common in recommender systems design (see Ricci et al., 2010), and the idea behind it is straightforward: a superior recommendation system should help the user find his preferred product earlier in the search process. A major advantage of this metric in our context is that it is completely model-free (i.e., not affected by the assumptions on which a ranking is built). However, this metric also has several drawbacks. First, and foremost, it says nothing about the click rates; second, all observed clicks are treated equally, while in fact the attractiveness of some clicked hotels may be higher than others; finally, no account is made for a possible response of search activity to ranking.

There are two distinct ways in which our proposed ranking method may lead to higher click rates than the default ranking. The most important effect comes from changes in the composition of choice sets: the types of hotels that are displayed to consumer. Hotels that were available but neglected by the default ranking are introduced for consumers' consideration. At the same time, less relevant options are moved out of the choice set. Another effect is re-ordering hotels on the page of results: by placing better alternatives closer to the top of the page, we improve the quality of the choice set, even without changing its contents. This is particularly true for the first page of results, as it is observed by all users. The optimal ranking derived from the model of search and clicks with targeting places 43.8 percent of the clicked hotels in the first page with a seventh position on average. In contrast, 23 percents of the clicks occur on the first page under the default ranking.

6 Conclusions

This paper proposes a model of search and click which endogenizes consumers' search refinement actions and proposes an optimal recommendation ranking for online platforms. Additionally, the paper derives a methodology to target individuals with rankings by exploiting the relationship between the parameters of request available to the platform to the expected price sensitivity of the consumer.

It is worth noting that the structural approach allows us to link a choice with the different consideration sets observed by the consumer as part of her search process. This modeling approach also allows for search and click prediction under alternative rankings and changing market conditions (availability and prices). This is difficult to achieve with reduced-form approaches to click, such as simple popularity rankings or more generally, machine-learning algorithms. Formally, the click maximization is achieved by improving the expected values of choice sets presented to consumers. The optimal ranking is computed for every individual consumer request, while conditioning on the set of currently available products and their prices. In this way, the proposed algorithm employs technological capacities of a search platform: its ability to collect information and manipulate displays in real time.

Counterfactual ranking comparisons demonstrate the poor performance of the default ranking used by the platform even when compared with simple popularity and price sort rankings. The largest gains of performance, as measured by CTRs, are derived from a simple discrete choice model based on hotel characteristics. Further gains of CTR are obtained by models that jointly explain search and click and by an optimized ordering of hotel results tailored to individual consumers.

This paper has important managerial implications. Online platforms offer consumers the ability to analyze and compare product attributes for a large set of distinct alternative products. Incorporating consumer choice models that allow for heterogeneous tastes on a variety of attributes lets companies more precisely present consumers with preferred alternatives. In addition to increased profitability through higher CTRs, reducing search frictions will likely increase consumer satisfaction, as well as retention and return rates.

References

Anderson, M., M. Ball, H. Boley, S. Greene, N. Howse, D. Lemire, and S. McGrath (2003): "RACOFI: A Rule-Applying Collaborative Filtering System," Working paper.

Ansari, A., S. Essegaier, and R. Kohli (2000): "Internet recommendation systems," *Journal of Marketing Research*, 37(3), 363–375.

Ansari, A., and C. F. Mela (2003): "E-Customization," *Journal of Marketing Research*, 40(2), 131–145.

Arbatskaya, M. (2007): "Ordered search," *RAND Journal of Economics*, 38(1), 119–126.

Basilico, J., and T. Hofmann (2004): "Unifying collaborative and content-based filtering," in *ICML*, pp. 65–72. ACM Press.

Belanger, F. (2005): "A conjoint analysis of online consumer satisfaction," *Journal of Electronic Commerce Research*, 6, 95–111.

Billsus, D., and M. J. Pazzani (1998): "Learning Collaborative Information Filters," in *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 46–54, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Burke, R. (2000): "Knowledge-based recommender systems," *Encyclopedia of Library and Information Science*, 69(32), 180–200.

Canny, J. (2002): "Collaborative filtering with privacy via factor analysis," in *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 238–245, New York, NY, USA. ACM.

Chen, Y., and S. Yao (2012): "Search with Refinement," Working paper.

——— (2014): "Sequential Search with Refinement: Model and Application with Click-stream Data," Working paper.

Ghose, A., P. Ipeirotis, and B. Li (2012a): "Designing Ranking Systems for Hotels on Travel Search Engines by Mining User-Generated and Crowd-Sourced Content," *Marketing Science*, 31(3), 492–520.

Ghose, A., P. G. Ipeirotis, and B. Li (2012b): "Designing Ranking Systems for Hotels on Travel Search Engines by Mining User-generated and Crowdsourced Content," *Marketing Science*, 31(3), 493–520.

——— (2014): "Examining the Impact of Ranking on Consumer Behavior and Search Engine Revenue," forthcoming at *Management Science*.

Hausman, J. (1996): "Valuation of new goods under perfect and imperfect competition," in *The Economics of New Goods, Studies in Income and Wealth*, ed. by T. Bresnahan, and R. Gordon, pp. 207–248. National Bureau of Economic Research.

Hendricks, K., A. Sorensen, and T. Wiseman (2010): "Observational Learning and Demand for Such Goods," Working paper.

Hofmann, T. (1999): "Latent Class Models for Collaborative Filtering," in *In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pp. 688–693.

Huang, Z., D. D. Zeng, and H. Chen (2004): "A unified recommendation framework based on Probabilistic Relational Models," in *In Fourteenth Annual Workshop on Information Technologies and Systems (WITS)*, pp. 8–13.

Kim, B.-D., and S.-O. Kim (2001): "A New Recommender System To Combine Content-Based And Collaborative Filtering Systems," *Journal of Database Marketing*, 8.

Kim, J. B., P. Albuquerque, and B. J. Bronnenberg (2010): "Online Demand under Limited Consumer Search," *Marketing Science*, 29(6), 1001–1023.

Koulayev, S. (2014): "Search for differentiated products: identification and estimation," *RAND Journal of Economics*, 45(3).

Melville, P., R. J. Mooney, and R. Nagarajan (2001): "Content-Boosted Collaborative Filtering," in *In Proceedings of the 2001 SIGIR Workshop on Recommender Systems*.

Moe, W. W. (2003): "Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream," *Journal of Consumer Psychology*, 13(1), 29–39.

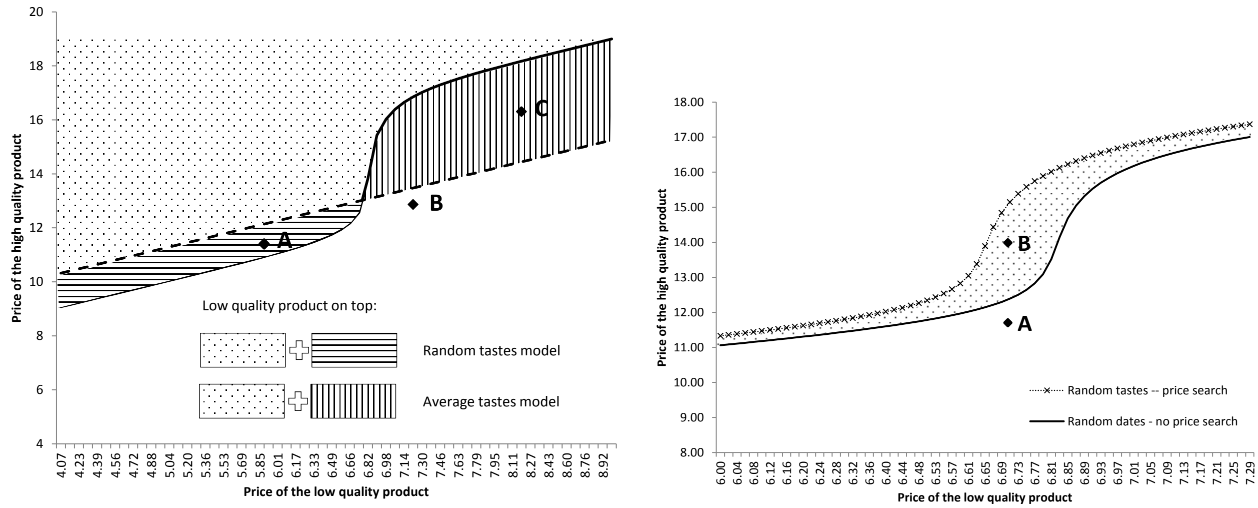
Montgomery, A. L., S. Li, K. Srinivasan, and J. C. Liechty (2004): "Modeling online browsing and path analysis using clickstream data," *Marketing Science*, 23(4), 579–595.

Moon, S., and G. J. Russell (2008): "Predicting Product Purchase from Inferred Customer Similarity: An Autologistic Model Approach," *Management Science*, 54(1), 71–82.

Vozalis, M., and K. G. Margaritis (2008): "Unison-CF: A Multiple-component, Adaptive Collaborative Filtering System," Working paper.

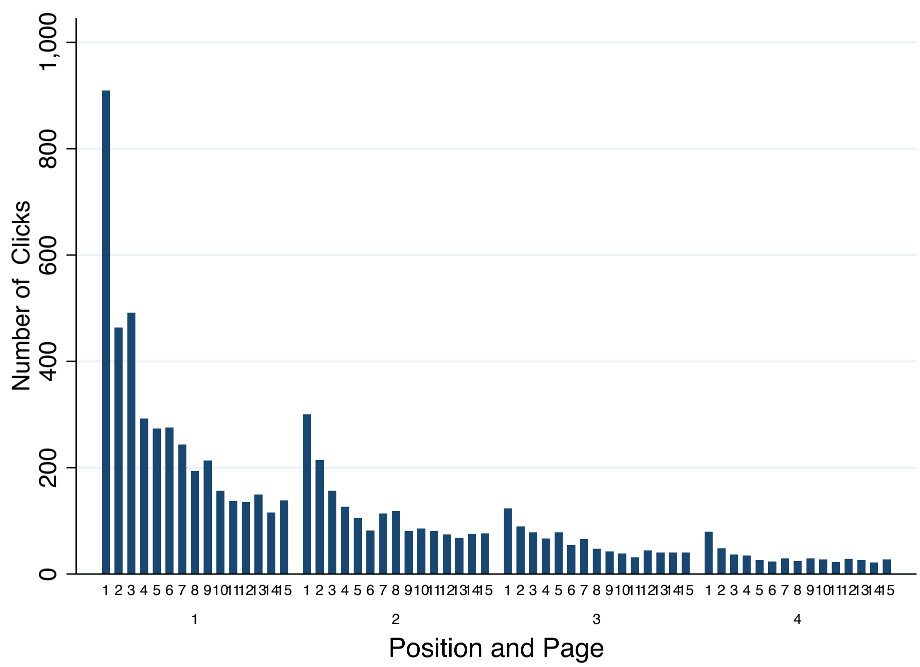
Yu, K., A. Schwaighofer, and V. Tresp (2003): "Collaborative ensemble learning: Combining collaborative and content-based information filtering via hierarchical bayes," in *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*, pp. 616–623.

Figure 1: Two Product Ranking Examples



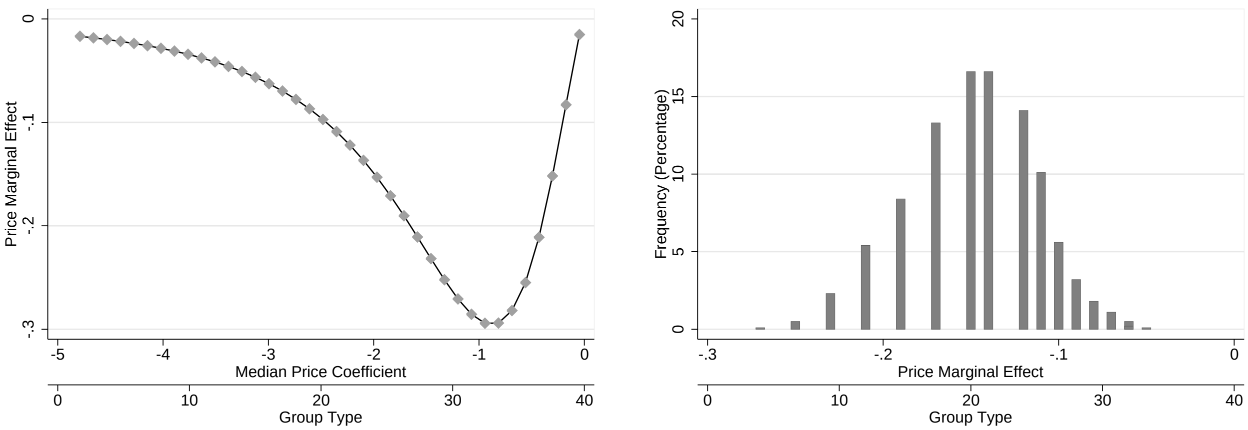
(a) No Search (b) Price Search

Figure 2: Clicks by Page Reached and Position



Note: The figure shows a breakdown of clicks by position and page where the click occurred. There are 15 results per page. The figure is truncated to the first four pages.

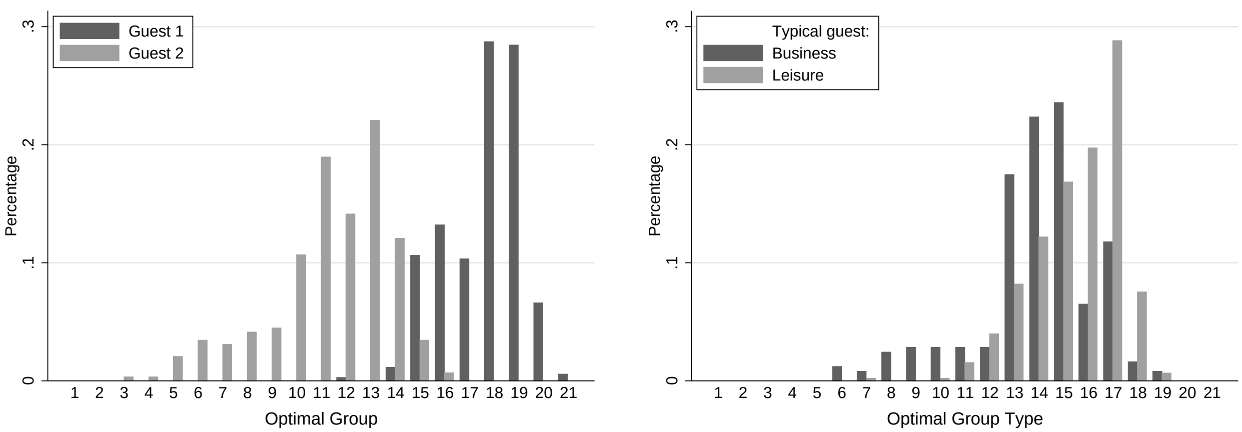
Figure 3: Marginal Effects and Consumer Type



(a) Price Marginal Effects and Consumer Type

(b) Sample Frequency of “Best Guesses”

Figure 4: Sample Frequencies of “Best Guess” by Request Parameters

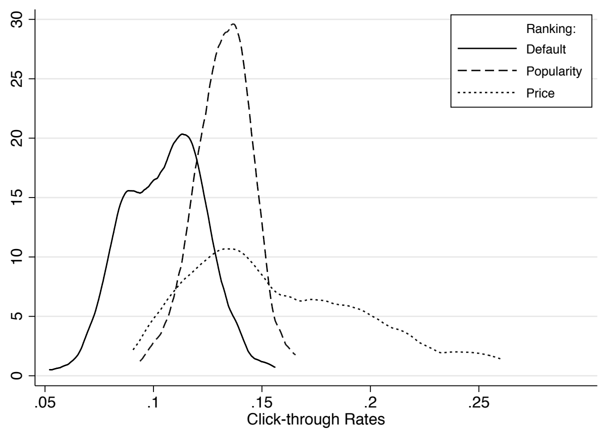


(a) Two Users with Extreme Price Elasticities

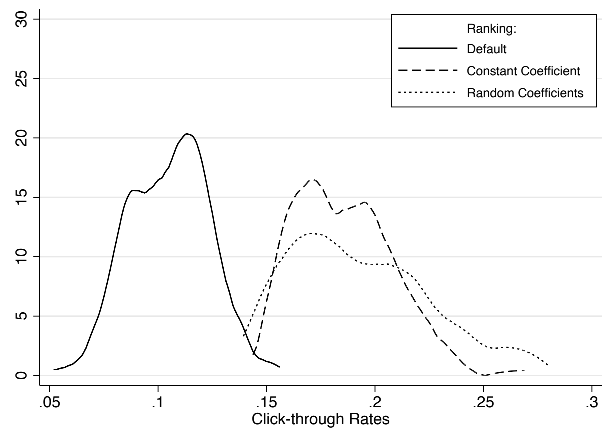
(b) Two Typical Users

Note: The figure shows the distribution of optimal “guesses” among users with different request parameters. In the left figure, guest 1 represents all users booking less than a week in advance and staying one night over weekend; guest 2 represents users staying 3 or more nights, booking more than a week in advance, with no weekend stay. In the right figure, the typical business guest travels alone, stays 3 or more nights, and does not include a weekend stay. The typical leisure guest consists of two adults, spending one night, and including a weekend stay.

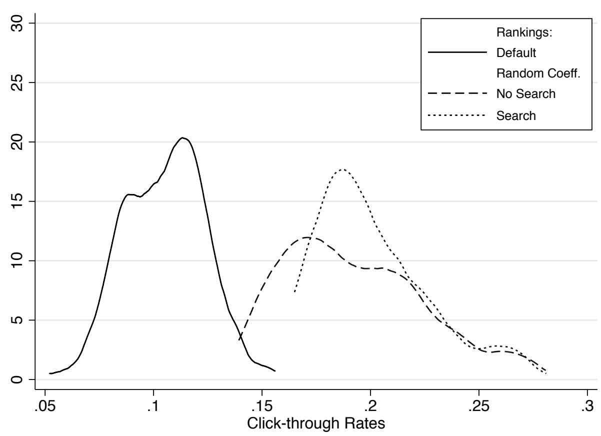
Figure 5: Distributions of CTRs under Alternative Rankings



(a) Simulated Rankings



(b) Ranking from Click Models



(c) Ranking from Search and Click Model

Table 1: Descriptive Statistics

	Mean	Median	Std. Dev.	Min.	Max.	Obs.
Request parameters						
Advance search (days)	33.50	21.00	36.63	1.00	364.00	23,959
Weekend stay	0.60	1.00	0.49	0.00	1.00	23,959
Number of people	1.84	2.00	0.97	1.00	8.00	23,959
Length of stay (days)	2.44	2.00	1.65	1.00	30.00	23,959
Across hotels						
First page location (% of obs)	30.64	22.18	28.33	0.00	98.35	148
Price (100s of dollars)	2.30	2.00	1.27	0.16	15.00	721,848
Clicked price (100s of dollars)	2.00	1.75	1.17	0.26	14.41	8,007
CTR (%)	1.11	0.80	0.94	0.00	5.82	148
Across consumers						
Any click	0.33	0	0.47	0	1	23,959
Click on the first page	0.17	0	0.38	0	1	23,959
Size of choice set (# of hotels)	30.00	26.00	21.11	10.00	135.00	23,959

Table 2: Breakdown of Consumers by Search and Click Actions

Search category	% of sample	% of total clicks
Browse default rankings		
1 page	40.8	36.3
2 pages	6.7	6.7
3 pages	4.4	4.2
More than 3 pages	5.7	4.8
Use sorting or filtering tools		
2 pages or more	42.4	48.0
Total	100	100

Table 3: Estimates of Discrete Choice Models of Hotel Choice

	Search model		Logit model random tastes		Logit model constant tastes		Logistic regression	
Distribution of price coefficient								
$E(log(-\alpha))$	-0.52	(0.19)	0.59	(0.32)	-0.21	(0.11)	-0.35	(0.11)
$SD(log(-\alpha))$	1.01	(0.41)	0.81	(0.34)				
Interactions of price with								
Saturday night stay	-0.65	(0.12)	-0.27	(0.14)	-0.16	(0.07)	-0.16	(0.10)
One guest	-0.61	(0.30)	-0.35	(0.13)	-0.06	(0.12)	-0.03	(0.02)
Less than 2 weeks advance	0.64	(0.27)	-0.46	(0.19)	-0.17	(0.12)	-0.24	(0.10)
Less than 3 days of stay	0.30	(0.11)	0.19	(0.18)	0.21	(0.15)	0.21	(0.15)
Non-price hotel characteristics								
Position effect (in dollars)	-35.32	(0.07)	-7.76	(0.05)	-18.51	(0.08)	-24.12	(0.09)
Distance to city center	0.04	(0.10)	-0.26	(0.21)	-0.25	(0.16)	-0.28	(0.13)
Random effect SD	0.28	(0.12)	0.27	(0.17)				
Distance to O'Hare	0.48	(0.16)	0.68	(0.15)	0.44	(0.25)	0.34	(0.12)
Random effect SD	0.15	(0.10)	0.2	(0.12)				
Outside option and request interactions								
Saturday night stay	0.72	(0.23)	0.44	(0.22)	0.19	(0.18)	0.15	(0.16)
One guest	0.80	(0.17)	0.97	(0.39)	0.31	(0.15)	0.15	(0.14)
Less than 2 weeks advance	-1.03	(0.14)	0.72	(0.17)	0.17	(0.12)	0.12	(0.14)
Less than 3 days of stay	-0.59	(0.31)	-0.13	(0.12)	-0.22	(0.18)	-0.22	(0.15)
Other Non-price hotel characteristics								
Star rating indicators	Yes		Yes		Yes		Yes	
Random effect SD	Yes		Yes		No		No	
Neighborhood indicators	Yes		Yes		Yes		Yes	
Random effect SD	Yes		Yes		No		No	
Interactions with requests	Yes		Yes		Yes		Yes	
Month indicators	Yes		Yes		Yes		Yes	
Log-likelihood	144,477		42,277		42,918		45,578	
Log-likelihood: search & click	144,477		173,957		174,598		177,258	
Log-likelihood: click	19,323		20,222		21,311		21,899	
No. of Observaions	21,366		21,366		21,366		21,366	

Notes. This table presents estimates of utility parameters for various models of discrete choice (see text for full description). The search model has hotel clicks and search actions as dependent variables; the other three models explain hotel clicks only. The random effects, if included, take normal distribution (and log-normal for price coefficient). Advance search is defined as the number of days between the date of search and the date of arrival. Default categories for request parameters are: no Saturday night stay, more than one guest, more than 2 weeks advance search, more than two days of stay. Standard errors in parentheses. * significant at 10%, ** significant at 5%, *** significant at 1%.

Table 4: Search Cost Distribution

Quantiles	Search Costs ($)
10	9.91
20	10.11
30	11.82
40	16.24
50	20.53
60	24.47
70	28.22
80	32.97
90	49.68

Table 5: Average CTRs under Alternative Rankings by Decile

		Simulated Rankings		Click Model		Search
	Default	Popularity	Price	Constant	Random	and click
Average	0.105	0.132	0.158	0.186	0.193	0.203
Decile 1	0.074	0.108	0.103	0.156	0.150	0.171
2	0.086	0.119	0.120	0.163	0.160	0.177
3	0.091	0.124	0.124	0.166	0.165	0.183
4	0.095	0.128	0.133	0.171	0.173	0.189
5	0.105	0.132	0.143	0.181	0.183	0.194
6	0.110	0.135	0.153	0.187	0.194	0.200
7	0.114	0.138	0.170	0.194	0.203	0.206
8	0.118	0.142	0.191	0.200	0.216	0.219
9	0.124	0.145	0.200	0.209	0.226	0.229
10	0.136	0.154	0.240	0.229	0.257	0.259

Table 6: Distribution of Clicks by Page

	Search and click	Default
Page	(%)	(%)
First	43.8	23.0
Second	20.9	18.2
Third	13.3	22.9
Fourth	9.5	15.5
Fifth	5.7	9.8
Other	6.9	10.7
Total	100	100

A Appendix: Additional Tables

Table A-1: Distribution of Search Strategies by First Search Action

	Strategy	Observations	% of sample
	Browse default rankings		
	Initial Page Only	8,540	35.6
1	1 additional page	1,295	5.4
2	2 additional pages	899	3.8
3	3 additional pages	398	1.7
4	4 additional pages	299	1.2
5	5 additional pages	240	1.0
	Search Refinement Strategies		
	Sort by		
1	Increasing price	2,507	10.5
2	Price and page turn	1,209	5.0
3	Distance to city center	581	2.4
4	Distance and page turn	313	1.3
5	Decreasing star rating	287	1.2
	Filter by		
6	Price—max 200	518	2.2
7	Price—max 300	281	1.2
8	Price—max 400	222	0.9
9	Distance to city center—5 miles	414	1.7
10	Distance to city center—2 miles	266	1.1
11	Distance to city center—10 miles	269	1.1
12	Landmark—O'Hare airport	441	1.8
13	Landmark—Navy Pier	250	1.0
14	Reset Landmark filters	211	0.9
15	Neighborhood—Gold Coast	269	1.1
16	Neighborhood—Loop	228	1.0
	Other search strategies	4022	16.8
	Total	23,959	100.0

B Appendix: Valuable properties of the extreme value distribution

Suppose x is EV Type 1 random variable with location parameters a and a unit scale. Its CDF and PDF are:

$$\begin{aligned} F_x(x) &= \exp(-e^{-(x-a)}) \\ f_x(x) &= \exp(-e^{-(x-a)})e^{-(x-a)} \end{aligned}$$

If $F(x)$ is a CDF of a standard EV Type 1 (with location zero and scale one), then $F(x-a) = F_x(x)$.

Claim 1 *The distribution of a maximum of n independent EV Type 1 random variables with location parameters $a_1, \ldots, a_n$ and unit scale, is also EV Type 1 with location parameter given by $M(a_1, \ldots, a_n) = \ln(\exp(a_1) + \cdots + \exp(a_n))$.*

Proof. The CDF of the maximum is: $P(\max(x_1, \ldots, x_n) < x) = F(x-a_1) \cdot F(x-a_2) \cdots F(x-a_n)$. The product of CDF's can be written as:

$$\begin{aligned} F(x-a_1) \cdot F(x-a_2) \cdots F(x-a_n) &= \exp\left(-e^{-(x-a_1)} \cdots - e^{-(x-a_n)}\right) \\ &= \exp(-e^{-x}e^{a_1} \cdots - e^{-x}e^{a_n}) \\ &= \exp(-e^{-x}(e^{a_1} + \cdots + e^{a_n})) \\ &= \exp(-e^{-(x-M(a_1,\ldots,a_n))}) \\ &= F(x - M(a_1, \ldots, a_n)) \end{aligned}$$

■

Claim 2 *Let x, y be independent extreme value type 1 random variables with location parameters μ_x and a, respectively. The probability of an event: $x > y, x_L < x < x_H$, where $x_L < x_H$ are constants, is given by:*

$$\begin{aligned} P(x &> y, x_L < x < x_H) = \int_{x_L}^{x_H} F_y(x) f_x(x) dx \\ &= \frac{\exp(\mu_x)}{\exp(M(a, \mu_x))} \left(F(x_H - M(a, \mu_x)) - F(x_L - M(a, \mu_x))\right) \end{aligned}$$

Proof. First, we substitute the definition of CDF and PDF of extreme value distribution and make some simplifications:

$$\begin{aligned}
\int_{x_L}^{x_H} F_y(x) f_x(x) dx &= \int_{x_L}^{x_H} F(x-a) f_x(x) dx \\
&= \int_{x_L}^{x_H} \exp(-e^{-(x-a)}) \exp(-e^{-(x-\mu_x)}) e^{-(x-\mu_x)} dx \\
&= \int_{x_L}^{x_H} \exp(-e^{-x} e^{a} - e^{-x} e^{\mu_x}) e^{-x} e^{\mu_x} dx \\
&= \int_{x_L}^{x_H} \exp(-e^{-x}(e^{a} + e^{\mu_x})) e^{-x} e^{\mu_x} dx
\end{aligned}$$

Now we can make a substitution: $t = e^{-x}$, $dt = -e^{-x} dx$.

$$\begin{aligned}
\int_{x_L}^{x_H} \exp(-e^{-x}(e^{a} + e^{\mu_x})) e^{-x} e^{\mu_x} dx &= \int_{\exp(-x_H)}^{\exp(-x_L)} \exp(-t(e^{a} + e^{\mu_x})) e^{\mu_x} dt \\
&= -\frac{e^{\mu_x}}{(e^{a} + e^{\mu_x})} \exp(-t(e^{a} + e^{\mu_x}))|_{\exp(-x_H)}^{\exp(-x_L)} \\
&= \frac{e^{\mu_x}}{(e^{a} + e^{\mu_x})} \left(F(x_H - a) F(x_H - \mu_x) - F(x_L - a) F(x_L - \mu_x)\right) \\
&= \frac{\exp(\mu_x)}{\exp(M(a, \mu_x))} \left(F(x_H - M(a, \mu_x)) - F(x_L - M(a, \mu_x))\right)
\end{aligned}$$

■

C Default Popularity Rankings

In the data, passive users, those who only browse the default hotel ranking, contribute a smaller share of clicks than consumers who refine their search. This finding questions the relevance of the default ranking system. Specifically, we want to know whether the current information available to the search engine—parameters of user request and current hotel prices—affects a hotel's display prominence. We test these hypotheses using regression analysis of pairwise hotel rankings. We create all possible hotel pairs from the hotel listings ranked by the default criteria. The resulting dataset consists of 9,734 hotel pairs and over 3 million observations of these pairs over time. For each hotel within a pair, we observe the page index, the rank (position on the page), and the price. For every search s we define an indicator variable Y_{jks} of whether hotel j is ranked higher than hotel k, $Y_{jks} = 1\,(position_{js} < position_{ks})$. By considering hotel pairs, rather than hotels in isolation, we eliminate contemporaneous shocks that affect both hotels' chances for prominence.

Table A-2 presents results of regression of Y_{jks} on request parameters (weekend stay, number of people, advance search) as well as differences in the prices of the two hotels. The price difference between hotels has a negligible effect on relative ranking: if hotel j becomes relatively more expensive than hotel k by \$100, the probability of hotel j being in a *lower* position increases by 6 percent. This does not reflect consumer preferences: other things equal, lower-priced hotels should instead have better rankings, as they are more attractive to consumers. Similarly, parameters of search request have little explanatory power for pairwise rankings.

To summarize, the data rejects the hypothesis that the search engine is using current information to rank hotels. Instead, it appears that the ranking is based on a measure of past hotel popularity (a fact confirmed by the website's managers). As does any ranking method based on past consumer choice data, the popularity ranking suffers from a feedback effect: hotels that initially obtained prominence will continue to enjoy it, even though their actual attractiveness may change. As a result of this persistence, consumers ignore superior options or waste resources learning about inferior ones, a phenomenon often referred to as a bad-herding behavior.[7] In contrast, the optimal ranking is based on current information available to the search engine.

[7] See for example Hendricks, Sorensen and Wiseman (2010) for a model of the effects of observational learning on herding behavior. Collaborative-filtering algorithms have been proposed extensively in the literature, e.g. Hofmann (1999), Billsus and Pazzani (1998), Kim (2001), Canny (2002) and Basilico and Hofmann(2004).

Table A-2: Effects of Price and Parameters of Request on Pairwise Rankings

Variable	OLS Y_{jks} (1)	Logit Marg. Effects Y_{jks} (2)
$P_{js} - P_{ks}$ (in \$100s)	0.058	0.064
	(0.000)***	(0.001)***
Weekend stay	-0.003	0.006
	(0.001)***	(0.005)
Advance search	0.000	0.000
	(0.000)***	(0.000)
Number of searchers	0.000	0.000
	(0.000)***	(0.000)***
Sample	Full	Random
Hotel fixed effects	Yes	Yes
R^2	0.29	
Log-likelihood		-26,067.32
Number of observations	3,124,288	49,796

Note: The table presents the effects of prices and other request parameters on the relative position of pairs of hotels using a binary dependent variable Y_{jks}, which takes the value of 1 if the position of hotel j is lower than hotel k and the value of 0 otherwise. For each user request, from the 148 hotels in the sample we constructed all the hotel pair combinations of the observed hotels resulting in 9,734 pairs. The explanatory variables include the price difference of the hotel pair, weekend stay dummy, the number of days in advance the hotel search is made, and the number of people searching that request and hotel dummy variables for each hotel. Standard errors in parentheses. * significant at 10%, ** significant at 5%, *** significant at 1%