Optimizing Multi-Stage Personalization in the Customer Journey

Yu Song*

September 18, 2025

Click here for the most recent version.

Abstract

Firms increasingly leverage personalization to influence product discovery and engagement throughout the customer journey. However, implementing effective multi-stage personalization is challenging because each stage's impact depends on customer intent and attention, and cross-stage spillovers may reduce overall effectiveness. This paper examines personalization at two key stages: an early, firm-driven recommendation stage and a later, customer-initiated query stage. Using data from a field experiment on a major e-commerce platform, I find that recommendation-stage personalization increases immediate revenue but reduces revenue at the query stage, resulting in no net gain. I then develop and deploy a personalized search-ranking algorithm in a subsequent field experiment. The results show that query-stage personalization increases total transactions without cannibalizing revenue from the recommendation stage. To identify the revenue-maximizing multi-stage personalization strategy across platform designs and customers, I build a structural search model. I exploit experimental variation and estimate the model using a neural network approach to address computational challenges. Counterfactual simulations reveal that personalizing only at the query stage results in higher total revenue than personalizing at both stages or only at the recommendation stage. Moreover, applying personalization only to customers with consistent preferences further improves revenue. These findings identify the conditions under which personalization is most effective and offer firms guidance on optimizing it across stages and customers.

Keywords: Multi-Stage Personalization, Personalized Recommendations, Personalized Search Ranking, Field Experiments, Consumer Search, Customer Journey

*I am a Ph.D. candidate in Marketing at the Stephen M. Ross School of Business, University of Michigan. Email: yyusong@umich.edu. I am deeply grateful to Jessica Fong and Puneet Manchanda for their invaluable mentorship and guidance. I thank Anocha Aribarg, Zach Brown, Alice Li, Jun Li, Lan Luo, Yeşim Orhun, Gary Russell, S. Sriram, Raphael Thomadsen, Candice Wang, Qingliang Wang, Shane Wang, and seminar participants at the University of Michigan for their helpful feedback and suggestions. I also thank the Mercari team, especially Ajay Daptardar, Michael Manzon and Kumon Takuma Yamaguchi, for their tremendous support in implementing the experiment and for the many insightful conversations. The views expressed in this paper are solely my own and do not necessarily reflect those of Mercari, Inc.

1 Introduction

As digital markets mature, personalization has become a central tool influencing how customers discover and engage with products. Firms that excel at personalization generate 40% more revenue than their peers.[1] Platforms such as Amazon and Spotify attribute 35% of transactions and a large share of a 1,000% increase in revenue, respectively, to personalized recommendation systems (MacKenzie et al., 2013; Abraham and Edelman, 2024). Given its potential, firms like Netflix and eBay have begun to apply personalization to multiple stages of the customer journey (Ostuni et al., 2023).[2] Customers often begin their journey by browsing recommendations on the homepage, and then have the option to initiate search queries to view additional products relevant to their queries. This process is prevalent across various platforms, including Amazon and eBay for e-commerce, TikTok and YouTube for video sharing, and Uber Eats and DoorDash for food delivery. However, implementing personalization effectively in the context of a multi-stage customer journey is challenging.[3] Each stage—from passive exposure to active search—differs in user intent, attention, and responsiveness, which could make personalization effective in some stages but ineffective in others. In addition, optimizing each stage independently may overlook important cross-stage spillovers. If negative spillovers exist, deploying personalization to more stages might be counterproductive. These trade-offs underscore the importance of considering multi-stage personalization holistically, rather than treating each stage in isolation.

I study two stages in which personalization is commonly applied to impact customer awareness and consideration: the recommendation stage and the query stage. The rec-

[1] McKinsey & Company (2022), "The Value of Getting Personalization Right—or Wrong—is Multiplying," `https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/the-value-of-getting-personalization-right-or-wrong-is-multiplying`, accessed June 2025.

[2] Modern Retail (2023), "eBay Steps Up Personalization Efforts to Improve Search, Ads and Experience," `https://www.modernretail.co/retailers/ebay-steps-up-personalization-efforts-to-improve-search-ads-and-experience/`, accessed June 2025.

[3] Adobe (2022), "Orchestrate the Customer Journey by Harmonizing the Four Instruments of Personalization at Scale," `https://business.adobe.com/blog/how-to/orchestrate-the-customer-journey-by-harmonizing-the-four-instruments-of-personalization-at-scale`, accessed June 2025.

ommendation stage, delivered through the homepage, represents an early, platform-driven interaction, while the query stage, triggered by a query, reflects a later, customer-initiated engagement. In this paper, I investigate how firms can optimize personalization at these two stages by accounting for both direct effects within each stage and spillovers across stages. Specifically, I attempt to answer two research questions. First, how do the effects of personalization vary across different stages of the customer journey? I focus on its effects on customer behavior (e.g., clicks and purchases) and platform revenue. Second, how do these effects differ across customers? If personalization is effective for some customers but generic (e.g., popularity-based) experiences work better for others, firms may be able to leverage this heterogeneity to "personalize personalization"—that is, to target personalization efforts toward those most likely to benefit while leaving others with a more generic experience.

It is ex-ante unclear at which stages of the customer journey firms should apply personalization. First, firms may offer personalized recommendations in the first step of the customer journey. On the one hand, customers may still be forming their preferences and rely more on recommendations that align with their prior preferences, especially when their preferences are consistent (Simonson, 2005; Bleier and Eisenbeiss, 2015). On the other hand, customer tastes can evolve over time (Yoon and Simonson, 2008), so that recommendations based on past activities (e.g., searches or purchases) may become outdated.

Second, firms may personalize the query stage, after customers initiate a search query and have more narrowly defined demand (Lambrecht and Tucker, 2013). At this point, platforms can leverage both historical behavior and real-time signals from the query to generate more accurate predictions (Li and Ma, 2020). Personalizing ranking at the query stage can decrease search costs and increase purchase rates (Donnelly et al., 2024; Korganbekova and Zuber, 2023). However, if customers do not find relevant products early in the recommendation stage, may make inference and exit the platform before reaching the query stage (Chen et al., 2025). Thus, the marginal benefit of personalizing rankings is minimal and can even turn negative once the extra latency caused by real-time personalization computation is

considered. Increased latency can raise search costs and reduce user engagement (Arapakis et al., 2021).

In addition to its direct effects *within* a single stage, personalization may produce spillover effects *across* different stages of the customer journey. Both direct and spillover effects contribute to the overall impact of multi-stage personalization. On the one hand, personalizing the recommendation stage could enable customers to find their desired products more efficiently, so that they have more time searching other products later in their journey (Song, 2021). On the other hand, by matching customers with satisfactory options too early, it reduces their incentive to engage in deeper search (Chen et al., 2025), which will likely reduce total revenue (Yuan et al., 2025). Overall, the impact of multi-stage personalization on overall outcomes depends on the magnitude and direction of spillovers across stages. My paper aims to understand the role of personalization in the customer search process and to identify the optimal stage for personalization within a unified framework.

To study these questions, I collaborate with Mercari, Inc. (Mercari), an e-commerce platform that enables individuals to buy and sell items across a wide range of categories. Founded in Japan, it has become Japan's largest community-powered marketplace and expanded its operations to the United States (US) in 2014. This paper focuses on Mercari US. I first analyze a large-scale field experiment that personalizes the recommendation stage. I find that the personalized recommendation stage increases user engagement (clicks, orders, and revenue) within that stage but reduces the likelihood that customers proceed to the subsequent query stage. As a result, query-stage revenue declines, and total revenue per user from both stages remains unchanged. The experimental results suggest that personalized recommendations have a negative spillover effect on consumer search.

I then design and implement a subsequent field experiment that personalizes the query stage.[4] I develop two versions of a ranking algorithm: a non-personalized version that uses only item-level features (e.g., brand) and contextual features (e.g., inventory of similar items

[4] All users in this experiment receive personalized recommendations at the recommendation stage.

at the time of the query), and a personalized version that additionally incorporates user-level features (e.g., past purchases in the last 30 days). To ensure that differences in outcomes are driven only by personalized features and not by differences in algorithms, I apply the same ranking algorithm, LambdaMART, to both conditions. LambdaMART operates by using many small decision trees to compare item pairs and learn how to order the search results. Deploying LambdaMART-based ranking models on Mercari, I find that personalizing the query stage reduces customers' search costs, as proxied by decreased use of sorting and filtering tools. Users exposed to personalized rankings at the query stage also click on more items and make more purchases, with no effects on the recommendation stage. However, personalization also introduces costs: it increases search latency, leading to longer page load times that, in turn, negatively impact click and purchase behavior.

In both experiments, the effects of personalization are stronger for customers with consistent demand (those who tend to click on similar types of items). In contrast, personalizing the recommendation stage reduces recommendation-stage revenue for customers with dynamic demand (who tend to explore a broader set of products). These findings suggest that a personalized personalization strategy, which applies personalization selectively to customers with consistent demand, may further enhance overall platform revenue.

While my two experiments offer valuable insights, they are insufficient to evaluate the effects of all combinations of multi-stage personalization (e.g., comparing personalization only at the query stage versus at both stages) or to assess alternative strategies such as personalized personalization. To fill these gaps, I develop a structural search model built on Weitzman's (1979) optimal sequential search framework. My structural search model captures individuals' inspection (i.e., clicking) and purchase decisions as they transition from the recommendation stage to the query stage. In the recommendation stage, the customer views limited product information, and decides either to inspect an item by clicking on it to learn more product information or to conduct a query to view additional products at the query stage. Both inspecting an item and conducting a query incur costs. If the customer

conducts a query, they will switch from the recommendation stage to the query stage. At the query stage, the customer can similarly incur a cost to inspect an item. The customer can terminate the process at any stage, either by selecting the outside option or purchasing one of the inspected products.

I exploit experimental variation in customers' propensity to transition into the query stage to estimate my search model. I apply the neural network estimator (NNE) developed by Wei and Jiang (2024), which trains a neural network to learn the mapping from observed data moments to the model's structural parameters. This approach circumvents the need for direct integration over unobservables and offers substantial computational efficiency.

Using the estimated model, I conduct counterfactual simulations to evaluate the impact of various multi-stage personalization strategies. In the first counterfactual simulation, I consider the scenario in which the firm recommends and ranks products using a utility-based framework (Ursu, 2018). Leveraging the same framework at both stages enables me to isolate the effects of personalization from those of the underlying algorithms at each stage. Results show that while personalization generally increases platform revenue, its effectiveness depends on the stage at which it is applied. Personalizing only the query stage generates 2.5% higher revenue than personalizing both stages and 12.1% higher revenue than personalizing only the recommendation stage. The revenue-maximizing personalization strategy remains robust across alternative algorithmic specifications, such as partial personalization that blends individual-level preferences with popularity signals, and under varying transition costs between the recommendation and query stages. In the second counterfactual simulation, I examine the impact of personalized personalization based on customers' preference consistency. I find that applying personalization only to customers with consistent preferences raises platform revenue by 13.6% over no personalization and by 2.7% over applying personalization to all customers.

This paper offers managerial implications. First, as firms increasingly coordinate personalization across multiple stages of the customer journey, they need to account for both within-

stage effects and cross-stage interactions. Given the evidence showing negative spillovers from the recommendation stage to the query stage, managers should balance the incremental benefits against costs when deciding whether, and at which stage, to personalize. My findings suggest that applying personalization to more stages does not necessarily improve outcomes. In fact, personalizing only the query stage maximizes total revenue. Moreover, this approach preserves product diversity: because the recommendation stage offers a broader assortment, leaving it non-personalized ensures customers still encounter a wide range of options. Such a balance mitigates regulatory concerns about filter bubbles, where overly tailored content can limit exposure to new or diverse products (Pariser, 2011). Second, firms can customize personalization at the individual level. For example, they might provide tailored recommendations to customers with consistent preferences, while offering a broader, non-personalized experience to those whose preferences shift more dynamically.

The remainder of the paper is structured as follows. Section 2 situates my work within the existing literature. Section 3 describes the empirical setting and data. Section 4 details two field experiments and shows the experimental results. Section 5 presents the structural search model and outlines the identification and estimation procedures. Section 6 reports the search model estimation results and counterfactual simulations. Section 7 concludes with implications and limitations.

2 Literature Review

This paper relates closely to the literature on personalized recommendation and personalized search ranking. Personalized recommendations have been shown to increase consumer engagement (e.g., Hosanagar et al., 2014) and purchase propensity (e.g., Li et al., 2022). However, they may also reduce consumption diversity and diminish long-term user engagement (Holtz et al., 2020; Chen et al., 2024). Personalized search ranking has been found to improve platform outcomes and consumer welfare, even when platforms take revenue into

account (Donnelly et al., 2024). While prior work focuses on a single stage of the customer journey, this paper is, to the best of my knowledge, the first to examine the combined effects of personalization across both recommendation and query stages. Prior literature has also documented costs associated with personalization: it imposes substantial computational burdens (Yoganarasimhan, 2020). Using my unique search latency data, I show that personalization increases page-load times and subsequently affects customer behavior.

My paper also relates to previous empirical work on the interplay between firm recommendations and consumer search. Song (2021) documents a positive spillover effect: personalized recommendations improve search efficiency in the essential product category and stimulate broader exploration in other categories. Wan et al. (2024) find that recommendations help consumers identify higher-value products, either through lower prices or better preference alignment. In contrast, Yuan et al. (2025) show that when recommendation relevance declines, consumers compensate by increasing their search activity. Fang et al. (2025) demonstrate that the relationship between recommender and nonrecommender searches depends on the number of information cues provided. My paper contributes to the literature by examining multi-stage personalization within a unified customer search framework.

This paper contributes to the growing literature on the customer journey. Prior research has employed a wide range of approaches to modeling the customer journey, including probabilistic machine learning techniques (Padilla et al., 2024), Poisson point process models (Goić et al., 2022), and transformer-based architectures (Lu and Kannan, 2024). My approach is most closely related to work that employs structural search models to analyze consumer decision-making. Existing research has examined how consumers progress from awareness to consideration and choice (Honka et al., 2017; Greminger, 2022), explored how they navigate product search pathways (Zhang et al., 2023), and investigated post-search behaviors such as revisits (Dang et al., 2024). I develop a two-stage sequential search model to understand the impact of multi-stage personalization on consumer decisions.

Furthermore, my work adds to the literature on the empirical search model (Honka et al.,

2019; Honka et al., 2024). There is growing empirical search literature built on the tractable solution offered by Weitzman (1979), such as modeling how consumers use refinement (Chen and Yao, 2017; De los Santos and Koulayev, 2017; Gu and Wang, 2022), incorporating consumer learning (Hodgson and Lewis, 2023; Korganbekova and Zuber, 2023), considering browsing behaviors (Greminger, 2022; Choi and Mela, 2019; Zhang et al., 2023), modeling which product attributes customers search over (Compiani et al., 2024), and so on. Search models typically assume that the decision to conduct a search is exogenous, but this assumption may not hold when the decision is influenced by prior experience, such as exposure to recommended products. I contribute to the literature by endogenizing consumers' decisions to conduct searches. Modeling these decisions is important when firms aim to coordinate consumer active search with passive search such as prior exposures to firm-generated recommendations.

3 Empirical Setting and Data

3.1 Empirical Setting

The empirical setting of this paper is Mercari US. As of 2024, it has 5 million monthly active users, 0.6 million sellers, 11 million new listings per month, and over 100 million app downloads.[5] Users can purchase new or used items from a wide range of categories (e.g., women's fashion, men's fashion, office supplies, home goods, etc.). Mercari emphasizes ease of use through its app-based platform, positioning itself as "a C2C marketplace app that allows anyone with a smartphone to easily sell items they no longer need."[6] This paper focuses on the Mercari app, including the field experiments and data.

In Figure K.1 in the appendix, I illustrate the platform layout by presenting examples of

[5] Sources: `https://about.mercari.com/en/press/news/articles/20230201_mercari10th/`; `https://thesmallbusinessblog.net/mercari-users/`; `https://www.mercari.com/about/`. All accessed January 2025.

[6] Source: `https://careers.mercari.com/en/services/`, accessed June 2025.

Mercari's home recommendation page, search results page, and product page. When users land on the platform through the app, they are first presented with product recommendations on the homepage. These homepage recommendations are pivotal, as 100% of user traffic passes through it. On average, customers spend over one minute at the recommendation stage before entering the query stage.[7] At a high level, the platform recommends products from category-brand pairs based on popularity (i.e., category-brand pairs with the highest cumulative revenue), trendiness (i.e., those with the largest revenue increase in the past weeks), and, when personalization is applied, the user's historical activities. I provide a more detailed description of the platform's recommendation algorithm used in this study in Section 4.1.

Users can conduct query-based searches, which return a ranked list of relevant results based on the submitted query.[8] The order from the recommendation stage to the query stage is mechanically fixed. Mercari employs a three-stage ranking process to generate search results, following a standard industry pipeline documented in the academic literature (e.g., Zhan et al., 2024; Chen et al., 2024). The process consists of: (1) a retrieval stage, where the platform retrieves a broad set of candidate products based on the query, typically ranging from 400 to 10,000 items depending on relevant product availability; (2) a ranking stage, where items are primarily sorted by relevance and recency, with more relevant or recent items receiving a higher score and appearing higher in the list; and (3) a re-ranking stage, where the top 100 items from the previous stage are re-ranked using a machine learning model. Section 4.2 provides further details on how products are re-ranked in this study.

[7] Users also see "similar items" at the bottom of the product detail page after clicking on a product. However, this alternative form of recommendations is not considered in this paper and is left for future research.

[8] Users may also conduct non-query-based searches by choosing product categories. In such cases, item rankings follow the platform's default algorithm and are unaffected by the experiment. As the majority of searches on the platform are query-based, this paper focuses exclusively on query-based search.

3.2 Data Description

For each user session, I observe the full sequence of consumer actions in both the recommendation and query stages, along with their timestamps. These actions include (1) conducting a query, (2) inspecting a product, as measured by the user clicking on the product, (3) making a purchase, and (4) exiting the platform. In both stages, I observe the set of displayed products and their corresponding positions in the list.

I observe each user's historical activity on the platform, including product clicks, purchases, and search queries. For each product, I record attributes such as brand, category, physical condition, listing age, number of photos, price, and seller characteristics. I also track downstream selling outcomes, including whether the product is sold and, if so, the sale date, transaction price, and buyer.

4 Personalization Field Experiments

I present two field experiments: personalizing recommendation stage (Section 4.1) and personalizing query stage (Section 4.2). Section 4.3 summarizes the experimental results and motivates the structural search model.

Since users may self-select into future sessions based on their first session's experience, I focus on each user's first session during the experimental period to mitigate selection bias, following the approach of Fong et al. (2023) and Fong (2024). Aggregated user-level outcomes across all sessions are reported in Tables K.1 and K.2 in the appendix. Both experiments pass the randomization check (see Appendix A.1) and manipulation check (see Appendix A.2). Finally, I rule out the potential stable unit treatment value assumption (SUTVA) violation using several checks in Appendix A.3.

4.1 Personalizing Recommendation Stage

The personalized recommendation stage experiment was conducted for two weeks in Fall 2022. The experiment was implemented on the Mercari mobile app; users accessing the platform via the web continued to receive non-personalized recommendations. My analyses focus on user activity during their first session in the experiment on the Mercari app. In this experiment, 50% of users were assigned to the control condition that received non-personalized (homepage) recommendations, and 50% of users were assigned to the treatment condition that received personalized recommendations. In the non-personalized condition, the platform tracked the categories of items clicked by users and recommended products in those categories based on product popularity and trendiness. Product popularity was measured by the cumulative historical revenue of its category-brand pair, and trendiness was defined as the maximum week-over-week revenue increase for that same pair. The only personal information used was each user's most recently clicked categories, and personalization happened within each category. Hence, customers who had clicked on the same category previously were shown the same set of recommended products. In the personalized condition, the platform used a neural-network architecture. Specifically, it constructed a latent preference profile for each user based on their recent activity, such as search queries and item clicks. Products the user interacted with were simultaneously embedded into the same latent space using product attributes such as product category and title. User and item representations were trained jointly so that items similar to those a user had previously engaged with were positioned closer to their profile. As a result, users were shown different products tailored to their individual historical interactions in each product category.

For new or inactive users (i.e., those without activity in the past 6 months), the platform applied a cold-start (non-personalized, popularity-based) algorithm to generate recommendations. Thus, the estimated effects should be interpreted as intent-to-treat (ITT).

4.1.1 Summary Statistics

During the experimental period, 3,050,146 customers viewed the home recommendations. 19.5% of customers clicked on at least one recommended item on the homepage; among them, the median number of clicks was 1, and the mean was 2.2. 0.9% of customers made a purchase from the recommended products.[9] Customers spent an average of 20 seconds and a mean of 72 seconds at the recommendation stage before initiating search queries. 56.6% of customers initiated a search query. Conditional on viewing search results, customers clicked on a median and mean of 3 and 5.6 items, and 2.8% of them made a purchase from the search results.[10]

4.1.2 Treatment Effects of Personalizing Recommendation Stage

Because users were randomly assigned to either the personalized or non-personalized recommendation condition, the difference in outcomes between these groups identifies the causal impact of personalizing the recommendation stage. I examine how personalizing the recommendation stage affects user engagement and platform outcomes by analyzing the following metrics: the number of clicks, orders, and revenue in both the recommendation and query stages, and whether the user initiates a search query.

Table 1 reports the treatment effects. In the recommendation stage, personalized recommendations increase user engagement: users click on 12.1% more items, place 8.4% more orders, and generate 8.2% more revenue compared to the non-personalized condition. However, there is a reduction in engagement in the subsequent query stage. Users exposed to personalized home recommendations are 4.2% less likely to conduct a search query, click on 4.7% fewer items, place 3.5% fewer search orders, and generate 4.1% less revenue from the query stage. I further establish the substitution effect between product recommendations

[9] A product purchase is attributed to the current session if the customer eventually completes the transaction for that product. This practice is consistent with prior work (e.g., Donnelly et al., 2024; Korganbekova and Zuber, 2023). I adopt this definition of purchase throughout the paper.

[10] I only consider the top 100 items in the search results, as items ranked lower than 100 are rarely clicked.

and consumer search by leveraging experimental variation as an instrumental variable in Appendix B. Applying the framework of Yuan et al. (2025), I find that personalized recommendations lead users to submit less specific search queries, providing further evidence that consumers arrive with an existing preference they aim to fulfill. This demand-fulfillment objective helps explain the observed substitution patterns between stages. Additional details are provided in Appendix G.

Since the query stage presents more relevant products, has a higher purchase rate and contributes a larger share of total revenue, even a modest percentage drop in its revenue offsets the revenue gains from personalizing the recommendation stage. As a result, total orders and total revenue (combining both stages) do not differ significantly between the personalized and non-personalized conditions.

Besides user engagement, I also examine how personalization affects the variety of products shown to customers. I find that personalization at the recommendation stage reduces product variety (see Appendix C). Nonetheless, the recommendation stage still exposes users to a broader range of products than the query stage.

Table 1: Treatment Effects of Personalized Recommendation Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Rec Clicks	0.404	0.453	12.135	29.018	< 0.001
Rec Orders	0.009	0.010	8.380	6.629	< 0.001
Rec Revenue	0.379	0.411	8.222	4.341	< 0.001
I(Initiate a Query)	0.578	0.554	-4.160	-42.344	< 0.001
Query Clicks	3.243	3.090	-4.717	-22.865	< 0.001
Query Orders	0.018	0.017	-3.517	-3.733	< 0.001
Query Revenue	0.576	0.552	-4.055	-2.361	0.018
Total Clicks	3.647	3.543	-2.850	-15.123	< 0.001
Total Orders	0.027	0.027	0.621	0.813	0.416
Total Revenue	0.955	0.963	0.821	0.638	0.524

Notes. This table shows the treatment effects of the personalized recommendation. 1,528,471 (50%) users are in the personalized condition. I report the mean of outcomes in both personalized and non-personalized conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

4.1.3 Heterogeneous Effects of Personalizing Recommendation Stage

Since consumer preferences can evolve over time (Yoon and Simonson, 2008), the extent to which past behavior reflects current intent likely varies across individuals, which in turn affects the performance of personalization. I examine how the effectiveness of the personalized recommendation stage depends on whether consumers exhibit relatively stable or dynamic preferences. To proxy preference consistency, I use the customer clicking pattern in the 30 days prior to the experiment and calculate the ratio of unique category-brand pairs to total items. A higher ratio indicates greater diversity in clicking behavior and, therefore, more dynamic preferences. I classify customers as having dynamic preferences if their ratio falls above the 80th percentile of the distribution.[11]

As shown in Figure 1, personalized recommendations are more effective for customers with consistent preferences, while non-personalized recommendations perform better for customers who tend to click a broader range of products.[12] This heterogeneity presents an opportunity for the platform to tailor their strategies based on preference consistency. For example, the platform might choose to apply personalization only for customers with consistent preferences and maintain broader exposure for customers with more dynamic preferences. Because this selective approach has not been tested in existing experiments, I evaluate its incremental value using counterfactual simulations.

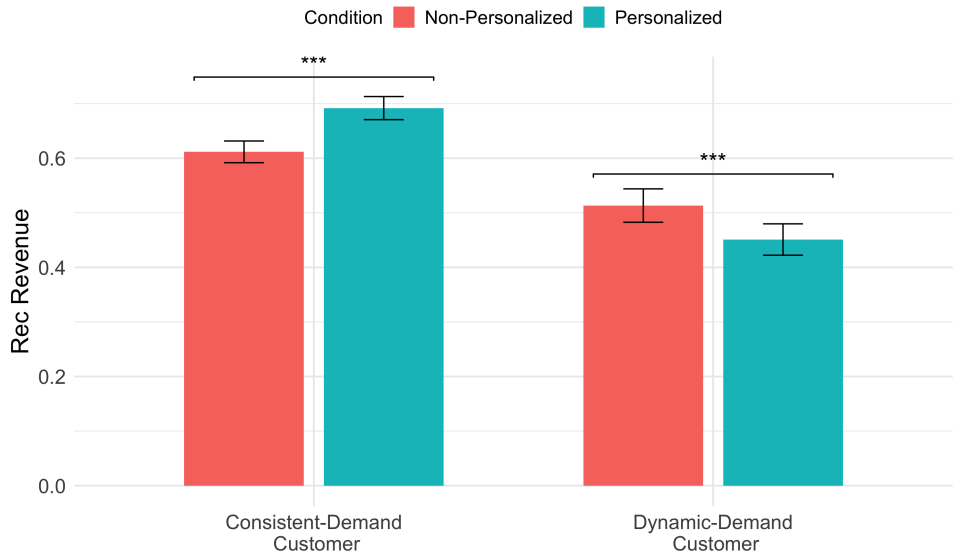
4.2 Personalizing Query Stage

Recall that personalization occurs in the re-ranking stage, where the top 100 products identified by the initial ranking stage are re-ordered. I first describe the ranking models—LambdaMART—used in this re-ranking stage. LambdaMART (Burges, 2010) is a widely used learning-to-rank algorithm that optimizes the ordering of items by minimizing a ranking

[11] I also test alternative cutoffs at the 70th and 75th percentiles and find similar patterns.

[12] These results should be interpreted as correlational rather than causal. However, the observed difference motivates a personalized personalization approach.

Figure 1: Heterogeneous Effects of Personalizing Recommendation Stage



Notes: This figure plots the average revenue at the recommendation stage for customers in the personalized and non-personalized conditions, separately for those with consistent and dynamic preferences. The sample is restricted to customers who clicked more than one item in the 30 days prior to the experiment. Error bars represent 95% confidence intervals.

loss. It combines gradient-boosted decision trees with a pairwise loss function to optimize item ordering based on relevance. LambdaMART has been extensively applied in business contexts (e.g., Microsoft's Bing search engine) and has demonstrated strong empirical performance (Yoganarasimhan, 2020; Kaye, 2024). In my experiment, I use LambdaMART as the ranking model for both personalized and non-personalized conditions. Both models are trained on the same set of product and contextual features, and the personalized model additionally incorporates user-specific features. Item features include attributes (e.g., brand) and past user engagement metrics (total item likes in the past 30 days). Contextual features are the signals that capture the market state at the moment a query is submitted, such as real-time inventory levels. User features include user activities from the past 30, 60, and 365 days, inferred preferences for brand, price, and category, past purchases, various user-level rating and status measures, and so on. Further details on the development of the ranking models are provided in Appendix D. By holding the underlying ranking model constant and varying only the inclusion of personalized features, any observed differences between the per-

sonalized and non-personalized conditions can be attributed only to personalized features rather than algorithmic differences.

The personalized query stage field experiment lasted for 12 days in Fall 2024. I randomly assigned 10% of users to the personalized search condition and another 10% to the non-personalized search condition. The remaining 70% of users received the status quo non-personalized ranking and 10% of users received the personalized version of the status quo ranking (i.e., the same model augmented with individual-level features).[13]

4.2.1 Summary Statistics

I record user activity on the Mercari app during the experiment.[14] The experiment included 360,679 individuals, of whom 187,818 (52.1%) conducted a search query. Among those who entered the query stage, the average number of clicked items was 4.2 (with a median of 3), and 4.5% of individuals made a purchase from the items displayed in the search results. Consumers also refined their search results: 14.5% of customers applied filters (e.g., by item condition or brand), and 3.1% sorted by price.[15]

To validate that the treatment condition indeed produces personalized rankings at the query stage, I examine the top 12 items displayed on the initial screen of the search results. Using price preference as an illustrative case, Figure 2 displays the price distribution of these items for customers inferred to prefer high versus low prices based on their past purchases.[16] The figure suggests that, under the personalized condition, individuals with a

[13] I use LambdaMART with and without user features as my experimental conditions, because the only distinction between these two conditions is the inclusion of user-level inputs. While the remaining conditions also compare personalized and non-personalized rankings, they employ different training approaches, making their results less straightforward to interpret. I use these remaining 80% of users to examine the robustness of personalization effects under an alternative ranking algorithm in Appendix F.

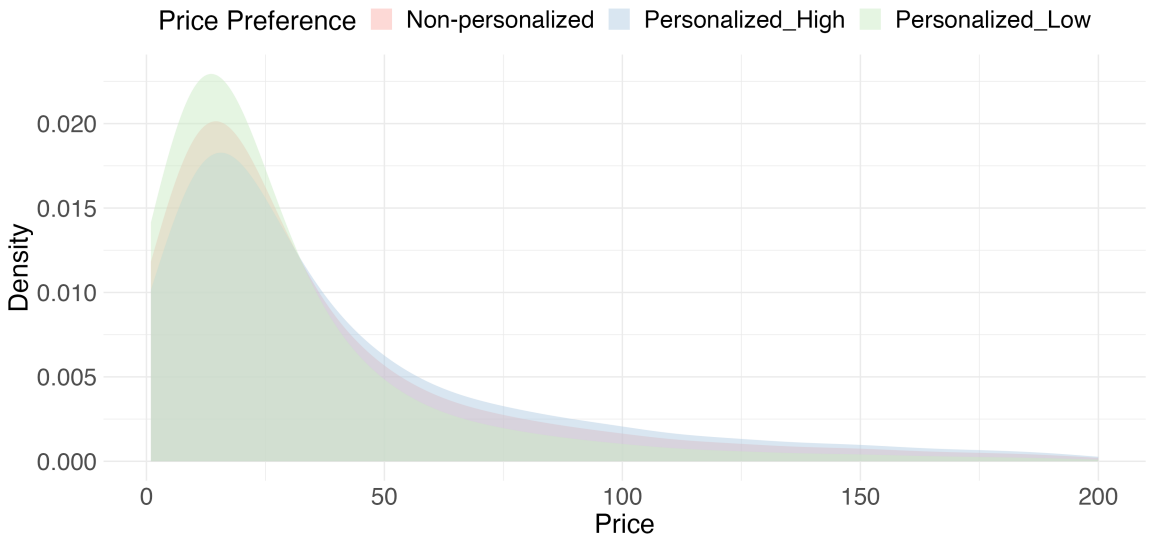
[14] Users were assigned to treatment conditions at the user level rather than the device level. Thus, a user assigned to the personalized condition received personalized rankings on both the app and the web (including mobile web).

[15] I do not consider customers who sorted by recency, as most of these sorting actions were initiated from a saved search, where customers saved a search during a previous platform visit.

[16] For example, a person is classified as preferring high-priced products if the highest prices among their past purchases exceed $50. This threshold is chosen based on historical data.

high-price preference are more likely to be shown higher-priced products, whereas those with a low-price preference are presented with lower-priced options. Moreover, the average item price remains similar across the two experimental conditions. In addition to price preferences, Appendix A.2 shows that customers in the personalized condition are shown a higher percentage of items that match their identified brand preferences at the top of the ranking list.

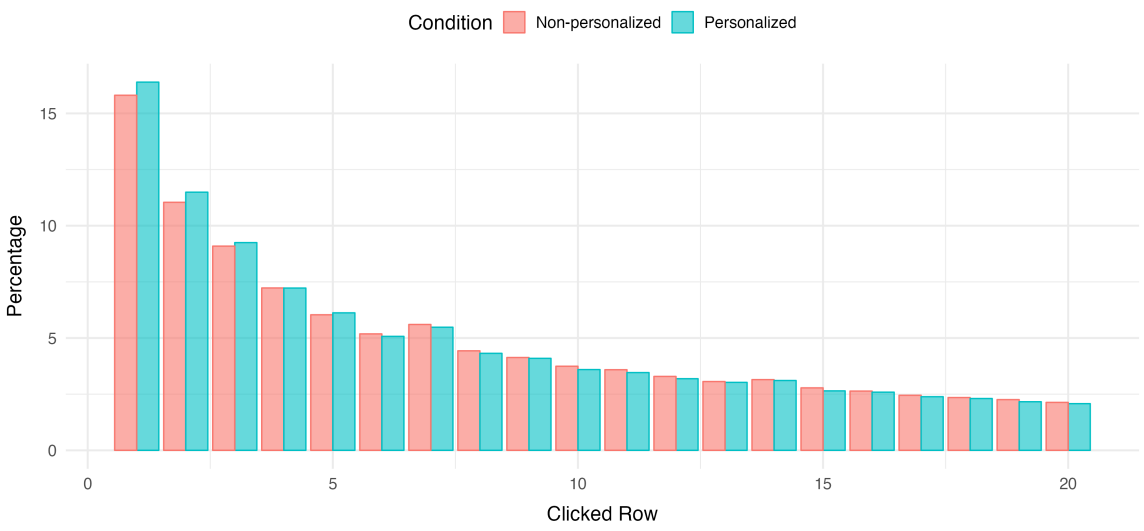
Figure 2: Price Distribution of Products at Query Stage by Customer Price Preference



Notes. This figure shows the price distribution of products displayed on the initial screen at the query stage (top 12 items), segmented by experimental condition ("personalized" vs. "non-personalized") and customers' price preferences inferred from past purchase histories ("high" vs. "low").

Figure 3 presents the distribution of clicks across item positions by experimental condition. Items ranked higher in the list receive more clicks, with this pattern more pronounced under personalized query stage. This suggests that personalization effectively elevates preferred products to more prominent positions.

Figure 3: Distribution of Clicked Positions



Notes. This figure plots the distribution of clicked positions in the ranking across experimental conditions, focusing on the top 20 rows of the search results. Each row contains three items. The x-axis represents the row number of the position, and the y-axis shows the percentage of total clicks.

4.2.2 Treatment Effects of Personalizing Query Stage

I present the treatment effect of personalizing the query stage.[17] To begin with, I proxy consumer search costs by examining their use of sorting and filtering tools. The outcome variables include two binary indicators: whether sorting is applied and whether filters are used. Table 2 shows that customers in the personalized condition are 5.2% less likely to sort search results and 4.3% less likely to apply filters, suggesting a reduction in search costs due to personalization. In addition, I examine how personalizing the query stage influences customer engagement and purchases by measuring item clicks, orders, and revenue. I find in Table 2 that customers in the personalized condition click on 3% more items, make 9.2% more purchases, and generate 10.6% higher revenue.

Table K.3 in the appendix shows that there are no changes in revenue-stage outcomes.

[17] Users were randomly assigned to an experimental condition upon arriving at the platform. By including both users who did and did not submit a search query, I estimate the overall impact of later-stage personalization under an intent-to-treat (ITT) framework.

Table 2: Treatment Effects of Personalizing Query Stage

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Sorted	0.032	0.030	-5.193	-2.863	0.004
Filtered	0.148	0.142	-4.331	-5.475	< 0.001
N Filters	0.192	0.184	-4.239	-4.978	< 0.001
Clicks	2.240	2.307	2.995	4.867	< 0.001
Orders	0.025	0.028	9.165	3.882	< 0.001
Revenue	0.838	0.927	10.622	2.870	0.004

Notes. This table shows the treatment effects of the personalizing query stage. There were 360,679 customers in the experimental conditions, and 180,438 (50%) customers in the personalized condition. I report the mean of outcomes in both personalized and non-personalized conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

When outcomes are aggregated across both stages, total clicks, orders, and revenue are higher in the personalized condition than in the non-personalized condition.

However, personalization imposes costs on customers: by introducing additional computation, personalization can increase search latency. Search latency is a critical concern for online platforms, as even minor delays can significantly degrade user experience and adversely affect key performance indicators.[18] Recognizing the importance of minimizing latency, many platforms have optimized their search algorithms for speed, such as Netflix (Ostuni et al., 2023), Airbnb[19] and Booking[20]. In Appendix E, I present experimental evidence showing that average search latency is higher in the personalized condition (274 milliseconds) compared to the non-personalized condition (259 milliseconds). Moreover, a 1% increase in search latency reduces customer clicks, orders, and revenue by 4.6%, 0.4%,

[18] For example, a Deloitte study highlights that every millisecond counts: a 0.1-second improvement in site speed increases retail conversions by 8.4% and average cart value by 9.2%. Source: Deloitte Consulting LLP (2021), "Milliseconds Make Millions: The Economic Impact of Site Speed," `https://www.deloitte.com/ie/en/services/consulting/research/milliseconds-make-millions.html`, accessed June 2025.

[19] Airbnb Engineering (2023), "Embedding-Based Retrieval for Airbnb Search," `https://medium.com/airbnb-engineering/embedding-based-retrieval-for-airbnb-search-aabebfc85839`, accessed June 2025.

[20] Booking.com Development (2022), "The Engineering Behind Booking.com's Ranking Platform: A System Overview," `https://medium.com/booking-com-development/the-engineering-behind-booking-coms-ranking-platform-a-system-overview-2fb222003ca6`, accessed June 2025.

and 1.2%, respectively.

I also examine how the effectiveness of the personalized query stage varies with customers' preference consistency. Using the same procedure outlined in Section 4.1.3, I present the results in Figure K.2 in the appendix. The findings indicate that personalized rankings have a stronger positive effect on query-stage revenue for customers with more consistent preferences (i.e., those who tended to click on similar items prior to the experiment). In contrast, there is no significant effect for customers with more dynamic preferences. Given the potential latency costs associated with personalization, the non-personalized query stage may work better for these customers.

4.3 Structural Model Motivation

To summarize, the experimental evidence shows that the personalized recommendation stage boosts revenue in that stage but crowds out revenue in the subsequent query stage, leaving total revenue from both stages unchanged. When the query stage, in addition to the recommendation stage, becomes further personalized, customers make more purchases, without compromising revenue from the earlier recommendation stage.

While these experiments provide insight into multi-stage personalization, they have limitations. First, the experiments alone cannot identify the revenue-maximizing stage(s) for applying personalization. These two experiments were conducted in different time periods, so direct comparisons are difficult due to potential temporal trends or other platform-level changes. Moreover, the experiments do not cover all combinations of multi-stage personalization. Specifically, they do not examine the scenario in which only the query stage is personalized. Its impact, relative to personalizing both stages, remains ambiguous depending on the magnitudes of spillovers between stages. Counterfactual simulations enable me to evaluate different personalization combinations within a unified framework.

Second, the recommendation stage and query stage algorithms differ in my field experi-

ments. Although it is standard industry practice to use distinct algorithms at each stage,[21] this variation makes it difficult to disentangle the effect of personalization from the effect of the underlying algorithms. To address this limitation, I perform counterfactual simulations that hold the underlying algorithms constant while varying only the stage at which personalization is applied.

Third, one might be concerned about whether the observed effects in these experiments can be generalized to other recommendation or ranking algorithms. To examine robustness, I test alternative personalization algorithms through additional field experiments. I summarize the key takeaways from these experiments below, with details about the experiments and results provided in Appendix F. First, due to substitution between stages, improvements in recommendation-stage personalization algorithms do not translate into total gains. Second, when the ranking algorithm is held constant, adding individual-level features generally improves platform revenue. Despite these additional experiments, existing field experiments on Mercari cannot test some other algorithmic designs. For instance, Amazon combines individual user preferences with aggregate signals based on "Best Seller" items determined by sales volume. To fill these gaps, I conduct counterfactual simulations to evaluate the incremental value of stage-specific personalization under alternative underlying algorithmic designs. Furthermore, no existing experiment evaluates the impact of personalized personalization, which similarly requires counterfactual simulation for assessment.

Fourth, experiments alone cannot determine whether the results generalize when the platform design changes. For instance, the platform may aim to boost engagement with recommendations or encourage more users to enter the query stage. Such changes could shift the relative contribution of each stage to total revenue, potentially altering which stage(s) should be personalized. To evaluate these scenarios, I rely on counterfactual simulations that vary the frictions governing transitions between stages.

[21] For example, Netflix employs Large Language Models (LLMs) for personalized recommendations and neural networks to rank search results. See Netflix Tech Blog at `https://netflixtechblog.com/foundation-model-for-personalized-recommendation-1a0bd8e02d39`, accessed June 2025.

To enable the aforementioned counterfactual simulations, I develop a structural search model in the next section.

5 Sequential Search Model

I begin by presenting the model setup and outlining the key components of the structural search model: the utility function, the information set and beliefs, and the associated costs. I then formalize the decision rules that govern customer decisions. Finally, I describe the estimation approach and discuss the identification of model parameters.

5.1 Model Setup

An overview of the search model is presented in Figure 4. The model consists of two stages: the recommendation stage and the query stage. In each stage, customers engage in sequential search. I assume that the customer's decision to visit the platform is exogenous (e.g., it is independent of whether the recommendation stage is personalized).[22] Additionally, customers are assumed to have a preexisting unit demand and stable preferences that remain fixed throughout the platform visit (e.g., viewing or clicking products does not change their preferences).[23,24]

In the recommendation stage, the customer is presented with a list of recommended products. They can costlessly observe the visible (i.e., pre-inspection) attributes of these products. In my setting, I consider price, brand, and category as visible attributes.[25] Upon

[22] As personalization is now widespread and many users even expect platforms to provide personalized recommendations, I assume that whether the recommendation stage is personalized does not affect overall traffic to the platform.

[23] As shown in Section 4.1.2, customers visit the platform to fulfill an existing demand, thereby supporting my model assumption.

[24] In my estimation sample, I focus on customers whose purchase intent is focused on a product in the "Video Games and Consoles" category. Details on how I construct the estimation sample are provided in Section 5.4.1.

[25] As shown in Figure K.1 in the appendix, customers see the price and a thumbnail image of each product. I assume that customers can infer the product's category and brand from the image, as it typically includes a brand logo.

viewing the recommendation list, the customer faces three options: (1) inspect an item from the recommended list to access its product page, (2) conduct a search query to switch from the recommendation stage to the query stage, or (3) leave the platform by opting for the outside option (e.g., no purchase or purchase on competing platforms). If the customer inspects an item, they fully uncover its hidden attributes and (post-inspection) match value. Hidden attributes in my setting include the number of photos, the item's physical condition and age, and the seller's past sales. The inspected items enter the customer's consideration set. After inspecting an item, a customer can either terminate the process by purchasing an item from their consideration set or choosing the outside option, or continue the process by inspecting another item or conducting a search query.

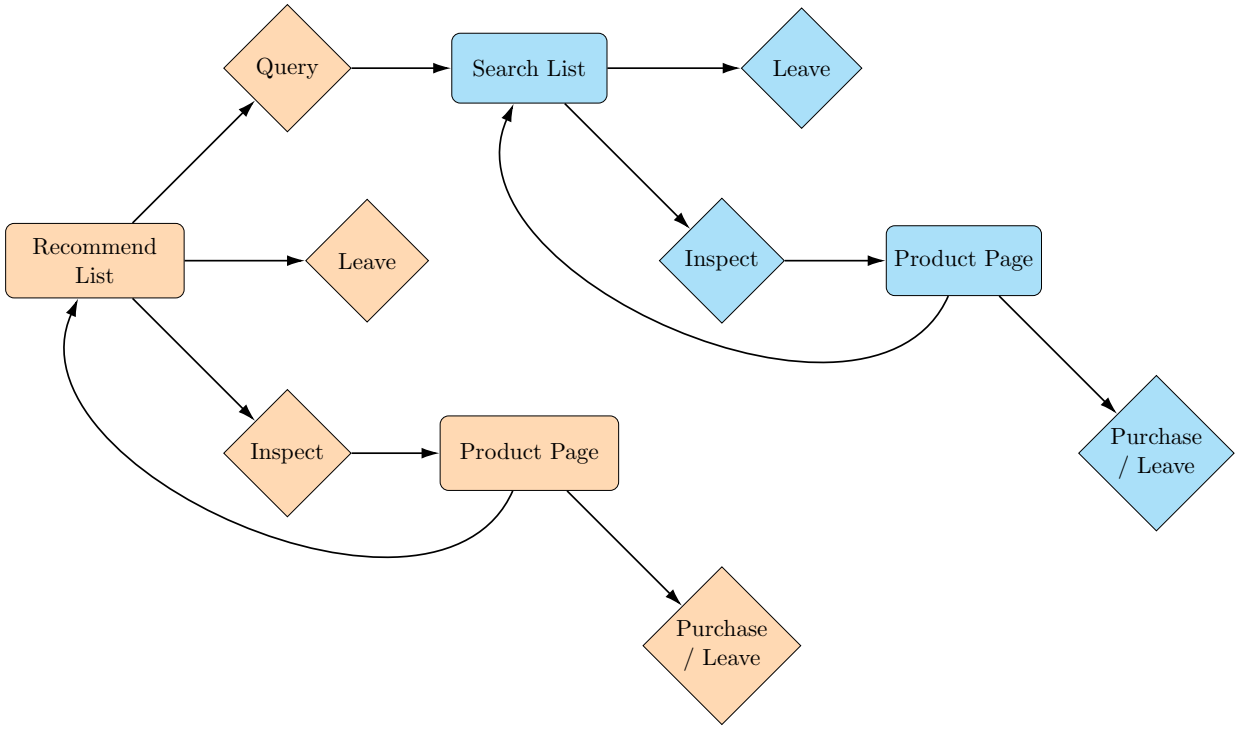
If the customer conducts a search query, they exit the recommendation stage and enter the query stage. In the query stage, they are presented with a list of products relevant to the submitted query, and can costlessly view the visible attributes of these products. The set of visible and hidden attributes is consistent across both stages. After viewing the search results, the customer faces two options: (1) inspect an item from the search result list, or (2) leave the platform by choosing the outside option.[26] Similarly, by inspecting an item in the query stage, the customer learns the hidden attributes and match value of the item. These inspected items expand the customer's consideration set. Customers can only purchase items from their consideration set, because on Mercari, users can only purchase an item from the product page (after inspection).

I assume that it is costless for the customer to navigate back to the recommendation or search list from the product page. Additionally, within each session, the customer is assumed

[26] In practice, customers may also return to the recommendation stage or submit another search query. In this case, the model could be extended to allow for two additional actions in the query stage: returning to the recommendation stage to inspect previously uninspected items, or submitting another query. Both actions mean that customers will enter a new sequential search problem, in which they update cost structures and carry over their current highest realized utility. In this paper, I abstract away from these behaviors to preserve model tractability. In my estimation, I keep only the first query if customers submit multiple queries within a session (16.5% of sessions) and only the first recommendation activity if the user returns to the homepage recommendations (3.5% of sessions).

to have perfect recall, i.e., they remember the utility from previously inspected products or can costlessly revisit them. Customers are also forward-looking. In the model, I do not incorporate refinement actions such as sorting or filtering, because, similar to Korganbekova and Zuber (2023), I find no experimental evidence that sorting or filtering significantly affects purchase behavior (see Table K.4 in the appendix).

Figure 4: Consumer Search Model Overview



Notes. Consumers begin the process at the recommendation stage, where they are shown a list of recommended items. If they choose to conduct a query to enter the subsequent query stage, they will be presented with a ranked list of search results.

5.2 Model Specification

5.2.1 Utility

Customer i derives the following utility from purchasing product j:

$$u_{ij} = \boldsymbol{X_j'}\boldsymbol{\alpha_i} + \boldsymbol{Z_j'}\boldsymbol{\beta_i} + \varepsilon_{ij}^{pre} + \varepsilon_{ij}^{post}. \tag{1}$$

$\boldsymbol{X_j}$ denotes the visible attributes (i.e., product attributes revealed before inspection), and $\boldsymbol{Z_j}$ denotes hidden attributes (i.e., product attributes revealed after inspection). The parameters $\boldsymbol{\alpha_i}$ and $\boldsymbol{\beta_i}$ capture customer i's preferences for visible and hidden attributes respectively. Customer i's product-specific pre-inspection idiosyncratic taste shock ε_{ij}^{pre} is known to the customer before inspection, and post-inspection taste shock ε_{ij}^{post} is known to the customer after inspection. Both taste shocks are unknown to the researcher. Examples of ε_{ij}^{pre} include the i's perceived attractiveness of product j's thumbnail image, and ε_{ij}^{post} include the match between the customer i and the seller of product j. The utility from the outside option u_{i0} is normalized to be ε_{i0}^{post}.

I assume that the idiosyncratic taste shocks (ε_{ij}^{pre} and ε_{i0}^{post}) follow normal distributions, consistent with the commonly adopted assumptions in Weitzman-type search models (e.g., Kim et al., 2010; Chen and Yao, 2017). The standard deviation of the pre-inspection taste shock is set to one as a scale normalization (i.e., $\varepsilon_{ij}^{pre} \sim \mathcal{N}(0,1)$). I assume that the post-inspection taste shock is independently and identically distributed across customers and products, i.e., $\varepsilon_{ij}^{post} \sim \mathcal{N}(0, \sigma^2_{\varepsilon^{post}})$, where its variance $\sigma^2_{\varepsilon^{post}}$ is estimated empirically.[27]

I allow for customer heterogeneity in price sensitivity and hidden attributes through random coefficients, such that $\alpha_i^{price} \sim \mathcal{N}(\alpha^{price}, \Omega_{\alpha^{price}})$ and $\boldsymbol{\beta_i} \sim \mathcal{N}(\boldsymbol{\beta}, \boldsymbol{\Omega_\beta})$, where α^{price} and $\Omega_{\alpha^{price}}$ represent the mean and variance of price sensitivity, and $\boldsymbol{\beta}$ and $\boldsymbol{\Omega_\beta}$ denote the mean and variance of preferences for hidden attributes.

5.2.2 Information Set and Beliefs

I use $\boldsymbol{A_i^{Rec}}$ and $\boldsymbol{A_i^S}$ to denote the list of products seen by customer i in the recommendation stage and query stage, respectively. Customer i's initial information set consists of $\boldsymbol{X_j}$, ε_{ij}^{pre} and pos_{ij} of $\boldsymbol{A_i^{Rec}}$, their preferences $\{\boldsymbol{\alpha_i}, \boldsymbol{\beta_i}\}$, the search query q_i, and the value of the outside option u_{i0}.

[27] See Ursu et al. (2024) for detailed discussions of why fixing the variance of the pre-inspection shock does not lead to a loss of generality, but why both taste shock standard deviations cannot be fixed simultaneously if one aims to monetize the inspection cost.

While in the recommendation stage, the customer does not observe any products in the query stage. Rather, they form a belief (rational expectation) about the utility of conducting a search query q_i, represented by the distribution F_i, such that $u_i^q \sim F_i$. Since users cannot observe u_i^q directly, they rely on noisy signals derived from two environments: (i) the query stage itself, and (ii) the recommendation stage. Formally, I assume that these signals are normally distributed around the expected maximum utility available in each environment. Consistent with prior literature (Koulayev 2014; De los Santos and Koulayev 2017; Gu and Wang 2022), I represent this belief using a single-dimensional parameter—the highest utility among *inspected* products, i.e., $u_i^q = \max\{u_{ij} | j \in \bar{\tilde{\boldsymbol{A}}}_{\boldsymbol{iq}}^{S}\}$. This is because when conducting a query, the customer cares only about the product with the highest utility in the search results. In addition, personalization may affect this maximum utility. Since personalized rankings elevate an individual's preferred products to higher positions and inspection costs change with position, the ranking affects the likelihood that high-utility items are inspected (see Figure 3). To factor in this change in expected utility due to personalization, I compute the maximum utility among the *inspected* products rather than all products.[28]

Specifically, I employ the following procedure to estimate the beliefs of conducting a search query. First, the customer forms beliefs about the products at position *pos* in the search list under the query q, represented by the distribution $H(\tilde{\boldsymbol{A}}^S | q, pos)$. I estimate this belief H using the empirical distribution of products seen by customers based on their query, personalization treatment and segment. If customer i is in the non-personalized condition, then H includes all products (and their positions) seen by customers who are also in the non-personalized condition and submit query q. If customer i is in the personalized condition, I further segment them based on the behavioral inputs used by the personalized ranking algorithm, such as price sensitivity, brand preferences, and activity levels. I then collect all products (and their positions) in the search result list shown to customers who are in

[28] Recall that personalized and non-personalized query-stage rankings contain the same set of products in the top 100 positions for a given query but differ in their ordering. In the absence of inspection costs, the customer would expect to reach the same maximum utility with and without personalization.

the personalized condition, submit query q and are in the same segment. The separation of personalization treatment and segment will capture, for example, the expectation that price-sensitive users are more likely to see lower-priced items ranked higher under personalized ranking (see Figure 2). Additional details on belief estimation are provided in Appendix H.1.

Second, given the high computational cost of calculating the maximum utility among inspected products across all sessions, I follow the literature on belief approximation and use bootstrapping to reduce the computational burden (e.g., Koulayev, 2014; Ghose et al., 2019). In particular, for each customer i, I bootstrap sample products from H based on their query, experimental condition, and segment (if assigned to the personalized condition). Each sample gives me an ordered product list $\tilde{\boldsymbol{A}}^S$, from which I simulate customer i's inspection decisions. The inspected products form their consideration set $\bar{\tilde{\boldsymbol{A}}}_{\boldsymbol{i}}^{\ S}$. Next, I compute the maximum utility among these inspected products, denoted as $\max_{j \in \bar{\tilde{\boldsymbol{A}}}_{\boldsymbol{i}}^{S}}(u_{ij})$. Repeating the above steps gives me the mean and variance of the belief. By assuming the normal distribution of the belief, I can write $u_i^{qS} \sim \mathcal{N}\big(\mu_i^{qS}, (\sigma_i^{qS})^2\big)$, where $\mu_i^{qS} = \mathbb{E}\left[\max_{j \in \bar{\tilde{\boldsymbol{A}}}_{\boldsymbol{i}}^{S}}(u_{ij})\right]$, and $(\sigma_i^{qS})^2 = \mathrm{Var}\Big(\max_{j \in \bar{\tilde{\mathcal{A}}}_i^S} u_{ij}\Big)$.

Similarly, following the same procedure, I can get the belief based on the recommendation stage experience. $u_i^{qR} \sim \mathcal{N}\big(\mu_i^{qR}, (\sigma_i^{qR})^2\big)$, where $\mu_i^{qS} = \mathbb{E}\Big[\max_{j \in \bar{\tilde{\mathcal{A}}}_i^S} u_{ij}\Big]$, and $(\sigma_i^{qR})^2 = \Big[\max_{j \in \bar{\tilde{\boldsymbol{A}}}_{\boldsymbol{i}}^{R}}(u_{ij})\Big]$.

Both u_i^{qS} and u_i^{qR} are unbiased but noisy signals of u_i^q. I denote their respective precisions as $\tau_i^{qS} = \frac{1}{(\sigma_i^{qS})^2}$ and $\tau_i^{qR} = \frac{1}{(\sigma_i^{qR})^2}$. Assuming conditional independence, the customer combines the two signals using Bayes' rule. The resulting posterior belief about the utility of entering the query stage is $u_i^q \sim \mathcal{N}\big(\mu_i^q, (\sigma_i^q)^2\big)$, with mean $\mu_i^q = \frac{\tau_i^{qS}\mu_i^{qS} + \tau_i^{qR}\mu_i^{qR}}{\tau_i^{qS} + \tau_i^{qR}}$ and variance $(\sigma_i^q)^2 = \frac{1}{\tau_i^{qS} + \tau_i^{qR}}$. Intuitively, the user treats each environment as providing a noisy but informative signal of the value of querying. The posterior belief is a precision-weighted average of the two sources, with posterior variance equal to the sum of precisions inverted. Overall, recommendation- and query-stage experiences jointly shape expectations about the value of conducting a query.

Before inspecting a product j, the customer observes its visible attributes $\boldsymbol{X_j}$. The product's visible attributes and position might be informative of its hidden attributes. For example, product j's price might correlate with its physical condition, which is hidden. Not accounting for this correlation might lead to biased estimates of the preferences for the visible attributes. I denote G_i as customer i's conditional beliefs about the utility of product j, such that $u_{ij}|\boldsymbol{X_j} \sim G_i$. I assume G_i follows a normal distribution where its mean and variance depend on the customer's beliefs about the hidden product attributes based on the visible attributes. I model the beliefs about hidden product attributes through a multivariate linear regression, which regresses the hidden attributes on the visible ones: $\boldsymbol{Z_j} = \boldsymbol{\Phi_0} + \boldsymbol{\Phi_1 X_j} + \boldsymbol{\eta}_j$.

In addition, the position of an item in the ranked list could affect customers' beliefs (Fong et al., 2024; Kaye, 2024). Empirical evidence further indicates that the prediction rule varies not only with item position but also across stages.[29] For instance, conditional on the same visible attributes, items displayed in the recommendation stage tend to come from sellers with fewer past sales than items shown in the query stage (see Figure H.2 in the appendix). To capture these stage-position beliefs, I estimate the empirical distribution of product attributes for each stage-position by pooling all items observed at that stage and position across user sessions.[30] Appendix H.2 details the estimation procedure and the results for the prediction rule.

The conditional mean of the belief distribution G_i can then be expressed as $\boldsymbol{X_j'\alpha_i} + \varepsilon_{ij}^{pre} + (\hat{\boldsymbol{\Phi}}_\mathbf{0} + \hat{\boldsymbol{\Phi}}_\mathbf{1}\boldsymbol{X_j})\boldsymbol{\beta_i}$. The conditional variance of G_i is $\boldsymbol{\beta_i}^\top \hat{\Sigma}_\eta \boldsymbol{\beta_i} + \sigma^2_{\phi_{ij}}$, where $\hat{\Sigma}_\eta$ is the estimated variance-covariance matrix of the regression residuals $\boldsymbol{\eta}_j$, and $\sigma^2_{\phi_{ij}}$ is the customer's belief about the variance of the post-inspection idiosyncratic taste shock ε_{ij}^{post}. I assume $\sigma^2_{\phi_{ij}}$ equals

[29] One possible reason for these stage-specific prediction rules is that different ranking algorithms and input data are used at each stage.

[30] One could also empirically estimate the prediction rule separately by personalization treatment and customer segment. However, allowing the empirical distribution of product attributes to vary by personalization condition does not improve model fit. To preserve estimation precision, I therefore apply a common prediction rule across personalization conditions. While the rule for hidden attributes is held constant across customers, the resulting distribution of hidden utility remains heterogeneous due to customer-specific preference coefficients.

its true variance $\sigma^2_{\varepsilon_{ij}^{post}}$.

5.2.3 Costs

Cost of Conducting a Search Query I model the cost of conducting a search query using the following equation:

$$c_i^{query} = exp(\tau_0 + \tau_1 Personalized_i), \tag{2}$$

where $Personalized_i$ is an indicator that equals 1 if customer i is in the personalized query-stage ranking condition, and 0 otherwise. I use this term to capture any potential costs borne by customers as a result of personalization. For example, as presented in Section 4.2.2, personalization increases search latency, which results in longer search result loading times and subsequently reduces product inspection decisions. Similar to Gu and Wang (2022), who assume that customers form rational expectations about loading time costs, I assume that customers form rational expectations about the costs introduced by personalization. The baseline cost of conducting a search query is captured by τ_0, which can also be interpreted as the cost incurred when the query-stage ranking is not personalized. The exponential function assumption of the cost function is consistent with prior literature (e.g., Kim et al., 2010; Ghose et al., 2012; Chen and Yao, 2017) and ensures that the costs are positive.

Inspection Cost Since the layout of product lists in the recommendation and query stages is identical (see Figure K.1 in the appendix), I adopt the same inspection cost structure in both stages. Inspection costs are specified as:

$$c_{ij}^{inspect} = exp(\delta_0 + \delta_1 pos_{ij}), \tag{3}$$

where pos_{ij} is the position of product j in customer i's list. As individuals read and process product information sequentially from top left to bottom right (Rayner, 1998), one can

consider that items in lower positions incur higher time or cognitive costs.[31] This is also supported by the empirical patterns in Figure K.3 in the appendix, which shows that items in lower positions receive fewer clicks and are inspected later in the session. The parameter δ_0 is the baseline inspection cost. I allow this baseline cost to vary between the recommendation and query stages, as, for example, inspection in the query stage may be subject to tighter time constraints due to its later occurrence in the customer journey.

5.3 Optimal Consumer Behavior

The model builds on the classic cost-benefit framework of Weitzman (1979), in which individuals evaluate each action by weighing its expected gains against the costs. Optimal decision rules are characterized by reservation utilities, which define the threshold at which an individual is indifferent between continuing the process and stopping.

The reservation utility of a product j equates the marginal gain from inspecting product j with the marginal cost of doing so. Intuitively, it represents the utility level a product must offer for the customer to be just indifferent between inspecting that product and stopping. For each uninspected product j, the customer computes its reservation utility z_{ij} as:

$$c_{ij}^{inspect} = \int_{z_{ij}}^{\infty} (u_{ij} - z_{ij})\, dG_i(u_{ij}|\boldsymbol{X_j}). \tag{4}$$

Similarly, the reservation utility for conducting a search query q equates its marginal gain with its expected marginal cost. The customer computes the reservation utility z_i^q as:

$$c_i^{query} = \int_{z_i^q}^{\infty} (u_i^q - z_i^q)\, dF_i(u_i^q). \tag{5}$$

Optimal behavior is determined by the relative values of reservation utilities. Under the

[31] Jameei Osgouei et al. (2023) indicate that, when presented with a matrix, two reading patterns yield qualitatively similar results: (1) reading each column from top to bottom before moving to the next column, and (2) reading each row from left to right before proceeding to the next row.

assumption that customers' beliefs follow a normal distribution, I use the solution derived by Kim et al. (2010) and implement a look-up table to recover reservation utilities (see detailed implementation discussion in Ursu et al. (2024)). Let $R(n)$ denotes the index of the selection option with the n-th largest reservation utility $z_{R(n)}$.[32] Let $\boldsymbol{O} = \{\boldsymbol{A}, q\}$ denotes the set of available options, which includes both the set of uninspected products $\boldsymbol{A}$ and the unvisited query q. Let $\bar{\boldsymbol{A}}$ be the set of products that have been inspected across both stages. The customer's current highest realized utility, denoted by u^*, is defined as the maximum utility among the inspected products and the outside option: $u^* = \max\left\{\max_{j\in\bar{\boldsymbol{A}}} u_j,\ u_0\right\}$. I outline the optimal rules as follows.

Selection Rule: The customer ranks all available options (including uninspected products and the query if in the recommendation stage) in decreasing order of reservation utilities, and then sequentially inspects each option in that order.

Stopping Rule: The customer stops the process if the current highest realized utility exceeds the maximum reservation utility of all remaining available options (i.e., uninspected products and the unvisited query if in the recommendation stage). Thus, stopping at step k implies:

$$u^* \geq \max_{n=k+1}^{\boldsymbol{O}} z_{R(n)}. \tag{6}$$

Choice Rule: The customer ultimately purchases the product with the highest realized utility among the inspected products and the outside option. Therefore, purchasing product j implies:

$$u_j = u^*. \tag{7}$$

[32] For notational simplicity, I drop the customer-specific subscript i in what follows.

5.4 Estimation and Identification

5.4.1 Estimation Sample Construction

In the estimation, I focus on consumer sessions that reveal purchase intent for items in the "Video Games and Consoles" category. I choose this category because its products are relatively standardized, meaning that, for example, product attributes like photos provide limited incremental information. Moreover, it ranks among the highest categories in terms of consumer searches, item inspections, and sales volume on Mercari. I define a consumer session as exhibiting purchase intent in the "Video Games and Consoles" category if it meets any of the following criteria: (i) the customer submits a relevant search query; (ii) if no query is submitted, the customer purchases a product from that category in the recommendation stage; or (iii) if neither a query is submitted nor a purchase is made, the customer inspects only items from that category. In cases (ii) and (iii) where no query is conducted, I assume the customer's intended query corresponds to their most recent query in the "Video Games and Consoles" category.[33]

Since products in the recommendation stage could be drawn from multiple categories depending on the customer's prior activity and many receive no inspection, it is difficult to identify category-specific coefficients in the utility function. Thus, instead of including individual product categories, I use a dummy variable to indicate whether product j belongs to the same category as customer i's purchase intent (i.e., "Video Games and Consoles" category in my model estimation).

5.4.2 Model Estimation Approach

The parameters to be estimated $\boldsymbol{\Theta}$ include the preference parameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \Omega_{\alpha^{price}}, \boldsymbol{\Omega_{\beta}}\}$, cost parameters $\{\boldsymbol{\delta}, \boldsymbol{\tau}\}$, and the post-inspection taste shock standard deviation $\sigma_{\varepsilon^{post}}$. I estimate the model using data from the experiment the personalized query stage. The random

[33] Note that for a product in the "Video Games and Consoles" category to appear in the recommendation stage, the customer needs to have previously submitted a related query.

assignment in the experiment introduces exogenous variation in customers' propensity to conduct a search query. The dataset contains, for each customer, the full list of products shown at both stages (with their attributes and positions), the search query submitted, the inspected and purchased items (including the outside option), and the order of inspections. I also observe each customer's personalization condition and segment.

I estimate the model using NNE proposed by Wei and Jiang (2024). The core idea behind NNE is to train a deep learning model to learn the mapping from data to the underlying structural parameters. In many structural econometric models, including the sequential search model, data moments are sufficient for pinning down parameters. This approach offers several advantages over traditional methods such as the simulated method of moments (SMM) and maximum likelihood estimation (MLE). First, NNE eliminates the need to evaluate high-dimensional integrals over unobservables, which are often required in SMM and MLE. Instead, the neural network implicitly learns the mapping from data to parameters. Second, NNE is robust to redundant or noisy moments, as the learning algorithm automatically prioritizes the most informative features for parameter identification. Third, this approach offers computational efficiency: once trained, the estimator can generate point estimates and statistical uncertainty measures with minimal additional computation. Finally, NNE is relatively robust to the choice of neural network architecture and hyperparameters.

I estimate my model parameters using NNE as follows. First, given the products and their positions in my actual estimation sample, I simulate consumer decisions over a grid of candidate parameter values. These simulated decisions include whether to inspect a product, conduct a query, and purchase a product. Second, for each parameter value, I summarize simulated outcomes into a set of data moments (e.g., average purchase rate), computed at the product, session, and individual levels. Third, I train a neural network using the simulated data moments and the corresponding parameter values as the training set. Through the training process, the network learns the inverse mapping: given a set of observed data moments, what parameter values are most likely to have generated them?

Once the network is trained, I compute the same set of data moments from the actual estimation sample and feed them into the neural network. The network then outputs the estimated model parameters and covariance matrix. Further details on NNE are provided in Appendix I.1.

5.4.3 Identification

This section presents the informal identification of parameters. I discuss the identification of position effects δ_1, price parameter α^{price}, preference parameters $\{\boldsymbol{\alpha_i}, \boldsymbol{\beta_i}\}$, cost parameters $\{\tau_0, \tau_1, \delta_0, \delta_1\}$, and standard deviation of post-inspection taste shock $\sigma_{\xi^{post}}$. I perform a Monte Carlo simulation to verify parameter identification and find that the estimation procedure precisely recovers all parameters (see Appendix I.2).

Position Effects $\boldsymbol{\delta_1}$ As detailed in Section 3.1, when a customer conducts a search query, the platform first retrieves a set of relevant items based on that query, then scores and orders them by relevance, and finally re-orders the top 100 results using either a personalized or non-personalized LambdaMART model. This procedure means that the set of top 100 items is determined by query relevance and not by individual or population preferences, but their final ordering reflects those preferences.

At the query stage, I leverage full knowledge of the ranking algorithm and its input features to generate a ranking score for each item. Because position is determined by the relative ordering of these scores within each session, I include the ranking score as a control variable to address position endogeneity. At the recommendation stage, products are selected and ranked based on customers' past histories. Although the proprietary recommendation algorithm is not directly observed by me, internal discussions with Mercari reveal its key input variables, such as historical search queries and item clicks. Similar to De los Santos and Koulayev (2017), I use these input variables to predict position, which serves as a control,

and treat the resulting residual variation as exogenous to individual preferences.[34]

I observe customers' repeated inspection decisions across positions, and these decisions are informative about the customer's sensitivity to position.

Price Parameter α^{price} Estimating the price parameter is prone to endogeneity, as prices may correlate with unobserved product-level taste shocks. To address this, I follow a standard approach using instruments constructed from the product attributes of competing products (see Gandhi and Houde (2019)). I define a "market" in my setting as the set of products that belong to the same category-brand cluster and are listed in the same week.

Preference Parameters $\boldsymbol{\alpha_i, \beta_i}$ As discussed above, product attributes are not endogenous in the query stage. However, they could be endogenous in the recommendation stage, since the algorithm selects and ranks products based on customers' past behaviors. I correct for this endogeneity using a control-function approach that leverages the recommendation algorithm's input variables, following the same approach used above to correct for position effects at the recommendation stage.

I identify customer preferences based on patterns in purchase and inspection. As I observe purchases directly, the identification of the preference parameters is as in a conditional choice model. Similar to how purchase decisions among a set of products reveal mean preference coefficients in a traditional discrete choice model, purchase decisions among inspected products inform the mean preference coefficients in my model. Given the low purchase incidence (3.1% of sessions), I further incorporate customers' inspection decisions. The selection of which product to inspect helps to pin down the preference parameters (e.g., Bronnenberg et al., 2016; Chen and Yao, 2017; Honka et al., 2017). For example, a tendency to inspect more low-priced items indicates larger price sensitivity.

[34] Since the recommendation algorithms were designed to encourage serendipitous exploration during the period from which my estimation sample is drawn, the residual variation can be interpreted as the system's exploratory component, which is exogenous to individual preferences.

The heterogeneity of the preference coefficients across customers can be recovered since I observe a sequence of decisions for each customer, including product inspection, query submission, and purchase. Repeated observations for each customer are informative for customer-specific preference coefficients, while the variation in these coefficients across individuals enables me to identify preference heterogeneity.

Cost Parameters $\boldsymbol{\delta_0, \tau_0, \tau_1}$ Conditional on the consideration set, the purchase decision depends only on consumer preferences and not on inspection costs. Therefore, I can separate inspection costs from preference parameters by using conditional purchase probabilities. The joint variation in inspection and purchase enables the identification of the distribution of consumer preferences. Conditional on those preferences, observed inspection decisions reveal the distribution of inspection costs. I identify the baseline inspection costs δ_0 for both the recommendation and query stages based on the number of product inspections customers perform in each stage.

To identify the effect of personalization on customers' cost of conducting a search query τ_1, I exploit variation in customers' experimental assignment (i.e., whether they are in the personalized or non-personalized condition). The baseline cost of conducting a search query τ_0 is inferred from the frequency with which customers conduct a search query. In my estimation sample, 84.7% of sessions involve a search query while 15.3% do not.

Post-Inspection Taste Shock Standard Deviation $\boldsymbol{\sigma_{\varepsilon^{post}}}$ The variation in the number of items a customer inspects pins down the ratio of the inspection cost to the post-inspection taste shock standard deviation. These two parameters can be separately identified by the parametric form. Since the product position shifts inspection costs but not post-inspection utility, this inspection cost shifter allows me to estimate the standard deviation of the post-inspection taste shock (Yavorsky et al., 2021).

6 Estimation Results and Counterfactual Simulations

6.1 Estimation Results

Table 3 presents the estimation results of the sequential search model. The comparisons between observed and estimated data patterns in Appendix I.3 suggest that my model fits the data well. To express the coefficients in monetary terms, I normalize each estimate by dividing it by the absolute value of the price coefficient. I find that customers derive higher utility from items that are lower priced, have fewer photos, are newly listed (i.e., smaller item age), are in better physical condition, and are sold by more experienced sellers. The estimation results also reveal heterogeneity in these preferences. The positive coefficient of "Video Games and Consoles" category is not surprising as customers prefer products from categories that align with their purchase intent. Relative to unbranded products, customers are willing to pay premia for four major brands: Sony ($0.50), Microsoft ($0.64), Nintendo ($0.41), and Pokemon ($0.45).

In terms of costs, I estimate that the baseline inspection cost is $0.24 in the recommendation stage and $0.18 in the query stage, while the baseline cost of conducting a search query is $0.14. To quantify position effects, I compute the dollar equivalent of the change in inspection cost from a one-unit increase in position. The estimated position effect is $0.05 in the recommendation stage and $0.04 in the query stage. For the initiation screen of 12 products, these inspection costs correspond to approximately $0.6 for the recommendation list and $0.48 for the search list. Personalization introduces frictions in transitioning to the query stage, with an estimated cost equivalent to about $0.16.

6.2 Counterfactual Simulations

As discussed in Section 4.3, some questions cannot be fully answered by field experiments and instead require counterfactual simulations. The first set of simulations explores how personalization at different stages of the customer journey affects total revenue and whether

Table 3: Estimation Results of Structural Search Model

Variables	Mean	SE	Heterogeneity (SD)	SE	Dollarized Value ($)
Preferences					
Constant	-3.3729	0.8535			
Price	-0.2462	0.0723	0.0934	0.0447	
1(Video Game Category)	0.8369	0.2501			0.6543
Brand: Sony	0.6438	0.1554			0.5033
Brand: Microsoft	0.8275	0.1141			0.6469
Brand: Nintendo	0.5197	0.1329			0.4063
Brand: Pokemon	0.5809	0.2229			0.4541
Brand: Others	0.2432	0.0965			0.1901
N Photos	-0.4825	0.0851	0.0284	0.0179	-0.3772
Item Age	-0.7931	0.3176	0.0367	0.0124	-0.4281
Physical Condition	0.1316	0.0305	0.0140	0.0862	0.0926
Seller Past Sales	0.0907	0.6041	0.1403	0.0857	0.0742
Costs					
Inspection Base (Rec)	-1.1647	0.1987			0.2439
Inspection Base (Search)	-1.4585	0.0772			0.1818
Position	0.1987	0.0705			0.0536 / 0.0400
Query Base	-1.6962	0.5616			0.1434
Personalization	0.7652	0.2762			0.1648
N consumers			8,138		
N sessions			17,884		

Notes. The last column reports the monetary value of each variable, calculated by dividing the coefficient estimate by the absolute value of the logged price coefficient. For example, the implied dollarized value of the preference is given by $\frac{\hat{\alpha}}{\exp(\hat{\alpha}^{price})}$ and the baseline cost is calculated as $\frac{\exp(\hat{\delta})}{\exp(\hat{\alpha}^{price})}$. Item age and seller past sales are (log+1)-transformed. The omitted brand is unbranded.

these effects hold under varying levels of transition friction. The second counterfactual simulation investigates the effectiveness of personalized personalization.

6.2.1 Multi-Stage Personalization

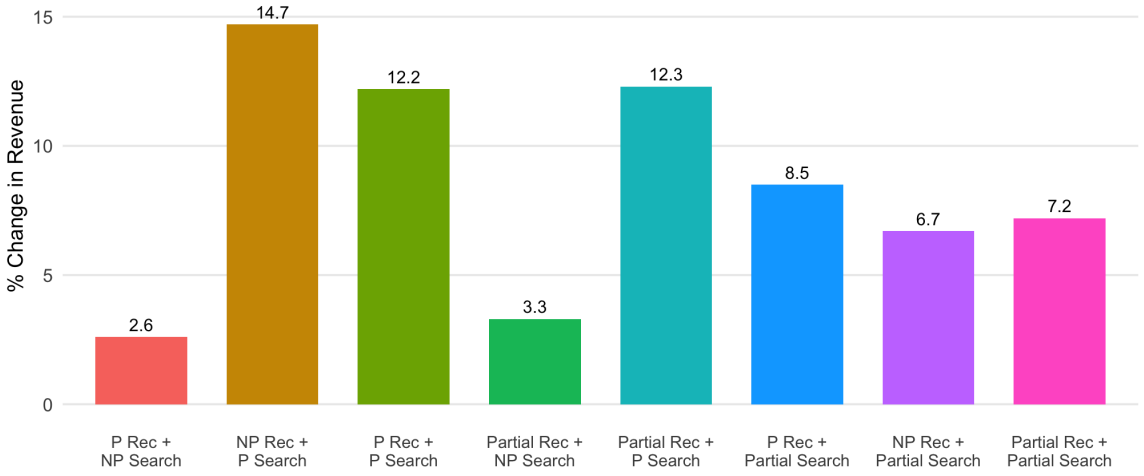
Utility-Based Recommendation- and Query-Stage Ranking Disentangling the effects of multi-stage personalization from those of algorithms requires holding algorithmic design fixed. Hence, I use the same utility-based model to rank both recommendations and search results. The simulations explore a spectrum of personalization levels, including full personalization, popularity-based non-personalization, and intermediate cases that reflect partial personalization. Specifically, my utility-based recommendation and query-stage ranking models integrate a weighted combination of individual and aggregate customer preferences. Appendix J provides details on the utility-based model. To simulate different algorithmic designs, I vary the model by adjusting the weights placed on individual versus average preferences: (1) *no personalization*, in which average utility receives full weight and individual utility receives zero weight; (2) *full personalization*, in which individual utility is given full weight; and (3) *partial personalization*, in which individual and average utilities are equally weighted.[35] I examine the impact of multi-stage personalization on total revenue by comparing the following three classes: (i) no personalization at either stage, (ii) full personalization at the recommendation stage, the query stage, or both, and (iii) partial personalization at one or both stages.

The results, presented in Figure 5, highlight the importance of selecting the appropriate stage for personalization. Personalizing only the early recommendation stage results in a modest improvement (2.6%) in total revenue relative to no personalization at either stage. Extending personalization to both stages does not maximize revenue. Instead, personalizing only the query stage delivers the largest uplift—2.5% more revenue than personalization

[35] I assume that the coefficient τ_1, which captures the effect of personalization on the costs of conducting a query, remains invariant regardless of the degree of personalization at the query stage.

at both stages. Under partial personalization, full query-stage personalization remains the most effective strategy. Although partial personalization slightly underperforms full personalization, it still delivers strong results. For instance, partial personalization in the query stage generates more revenue than fully personalizing only the recommendation stage. This partial personalization approach is particularly valuable when firms seek to preserve product variety for long-term performance (Chen et al., 2024) or when data and resource limitations hinder accurate individual preference prediction.

Figure 5: Total Revenue of Multi-Stage Personalization



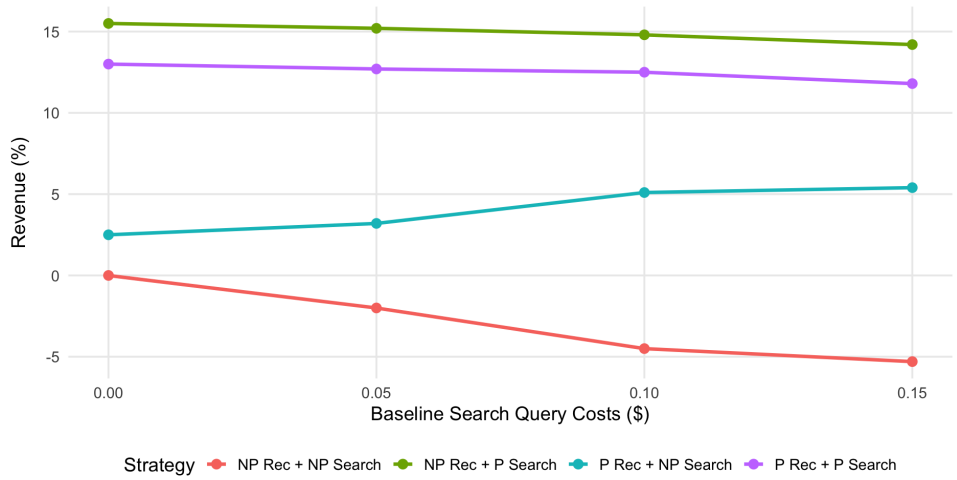
Notes. This figure plots revenue under various combinations of recommendation and query ranking models. In this simulation, I abstract away from heterogeneity in position effects. The x-axis indicates the personalization strategy, and the y-axis shows the percentage change in revenue relative to the baseline, which is the non-personalized condition at both stages. "NP" denotes non-personalized and "P" denotes personalized.

Frictions at Stage Transitions As discussed in Section 4.3, platforms can structure the user experience to be recommendation-centered or search-centered. These structural design choices influence how customers progress through the journey and, in turn, affect the relative value of personalization at each stage. In this counterfactual simulation, I examine whether the previously identified revenue-maximizing strategy—personalizing only the query stage—generalizes across different platform structure designs. Specifically, I vary the baseline

cost of conducting a query. Intuitively, when the cost of transitioning from the recommendation stage to the query stage is low, more users proceed to search, increasing the importance of query ranking and signaling a search-centered design. Conversely, when the query cost is high, users rely more on recommendations, making the design of the recommendation more critical, indicative of a recommendation-centered platform.

Figure 6 plots simulated total revenue under varying levels of transition frictions. First, the multi-stage personalization patterns hold across different platform designs (whether recommendation-centered or search-centered). That is, personalizing only the query stage consistently yields the highest total revenue, followed by personalizing both stages, and then by personalizing only the recommendation stage. As the baseline cost of conducting a query increases, the return to query-stage personalization declines, while the return to recommendation-stage personalization rises.

Figure 6: Revenue of Each Personalization Strategy by Stage Transition Costs



Notes. This figure plots revenue under alternative combinations of recommendation and query ranking models. I abstract away from heterogeneity in position effects in this simulation. The x-axis shows the baseline cost of conducting a query (in US dollars), and the y-axis shows the percentage change in revenue relative to the baseline, which is the non-personalization at both stages when the baseline search query cost is 0.. "NP" refers to non-personalized and "P" to personalized.

6.2.2 Personalized Personalization

The experimental results indicate that personalization delivers significantly greater gains for customers with consistent preferences, while its benefits diminish for those with more dynamic preferences. This finding points to the potential for identifying preference consistency patterns and tailoring the personalization to each customer. To quantify the value of this approach, I simulate a population in which 20% of customers are randomly assigned to a "dynamic preference" segment and 80% to a "stable preference" segment.[36] In the dynamic-preference segment, I generate personalized recommendations and query rankings from their estimated preference parameters, then introduce a 10% random perturbation to mimic temporal shifts in preferences; the consistent-preference segment experiences no such shocks. I compare mass personalization—where both stages are personalized for all customers—with personalized personalization, in which both stages are personalized only for the consistent-preference segment. As shown in Figure 7, targeted personalization raises total individual-level revenue by 2.7% relative to universal personalization and by 13.6% relative to applying no personalization in either stage.

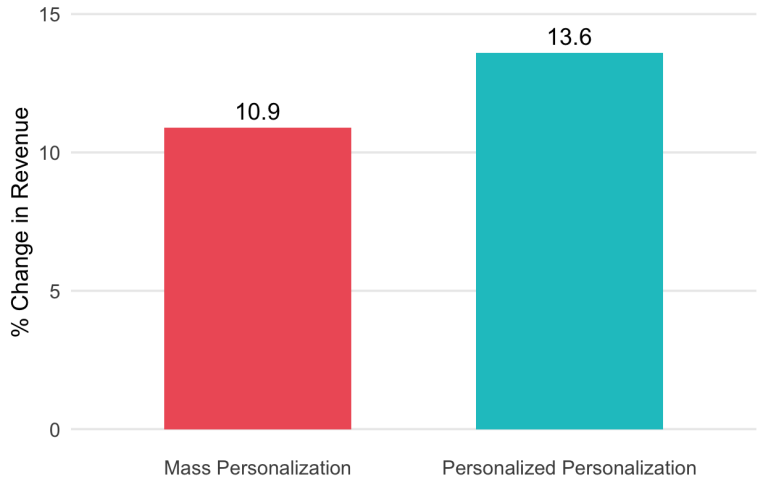
In practice, firms can infer preference consistency from user historical activities, such as purchase or search histories. For example, a customer who consistently buys low-priced or new-condition items will likely benefit most from heavy personalization, whereas customers whose preferences jump across categories or over time may be better served by a popularity-based, non-personalized offerings.

7 Conclusion

Personalization is central to the operations of many digital platforms. For instance, Spotify curates discovery feeds, Netflix tailors thumbnails, and Amazon customizes search results. From homepage recommendations that capture initial engagement to query-stage rankings

[36] This split fits the empirical patterns the best.

Figure 7: Revenue of Personalized Personalization Based on Preference Consistency



Notes. This figure shows the revenue across personalization strategies. The x-axis shows the personalization strategy, and the y-axis shows the percentage change in revenue relative to the baseline condition, i.e., no personalization is applied for any individual at either stage.

that showcase additional relevant products, personalization influences multiple stages of the customer journey. Although many firms make significant investments to optimize each touchpoint, this siloed approach could create tensions when gains from personalization in one stage offset benefits in another. Ignoring important spillovers between stages may lead to suboptimal outcomes. Designing effective multi-stage personalization thus requires a systematic understanding of both within-stage effects and cross-stage spillovers. This paper examines two stages where personalization plays a role—the recommendation stage and the query stage—and investigates how its effects vary depending on which stage(s) it is applied.

Using two field experiments, I find that although personalizing the recommendation stage can boost initial engagement, it can reduce user engagement in the subsequent query stage, ultimately leaving total revenue flat. When both stages are personalized, total revenue increases relative to personalizing only the recommendation stage. These findings suggest a one-way negative spillover from the recommendation stage to the query stage. I also find that the impact of personalization is more pronounced for customers with consistent preferences. In fact, personalizing the recommendation stage can reduce revenue for those with more

dynamic preferences.

To further identify the multi-stage personalization strategy that maximizes total revenue, assess its generalizability and quantify the value of personalized personalization approach, I develop a two-stage sequential search model. Counterfactual simulations reveal that personalizing only at the query stage yields the highest revenue, outperforming both no personalization and personalization at both stages. This revenue-maximizing personalization strategy remains robust to variations in transition costs between stages. In addition, assigning personalization only to users with consistent preferences can further boost platform revenue by 2.7% relative to applying personalization to all customers.

This paper highlights the importance of viewing personalization not as an isolated, one-off intervention, but as a coordinated, multi-stage strategy integrated into the broader consumer decision-making process. While personalization can be beneficial, managers should recognize potential negative spillovers between stages. Increasing the number of personalized touchpoints across the customer journey does not necessarily enhance overall performance, challenging the conventional wisdom of "more personalization is better." Instead, the findings suggest that personalizing only the query stage emerges as the optimal approach, which maximizes total revenue while broadening customer exposure to product variety through the non-personalized recommendation stage.

In addition to optimizing personalization across stages of the customer journey, exploiting heterogeneity across customers can unlock even greater value. Customers differ in their responses to personalized experiences. For example, customers with dynamic preferences not only prefer a more diverse set of products, but are also harder to predict using historical activity data. As a result, assigning non-personalization, popularity-based offerings could perform better for these customers. In contrast, customers with consistent preferences benefit from personalization. Firms can leverage behavioral histories (e.g., browsing, search, or purchase) to identify these preference patterns and optimize their personalization strategies accordingly.

My paper is not without limitations. First, the field experiments in my study last for less than two weeks, and so my findings primarily capture short-term effects of personalization. The long-term impact may differ. For example, sellers might respond strategically by adjusting prices or product quality, which could in turn affect consumer welfare and platform revenue (Kaye, 2024). Second, my study focuses on two key stages of the customer journey, but platforms may personalize other touchpoints as well, such as push notifications or checkout discounts. These additional interventions could be studied alongside the two stages considered in this paper. Furthermore, I use preference consistency to show how personalization can be targeted to a subset of customers. This approach serves as a starting point. Firms with rich customer data can extend the idea by assigning different intensities of personalization across customers or by tailoring personalization at the product-category level. Third, the context of this study is an e-commerce platform, where customers often arrive with well-defined purchase intents. The implications may differ in other settings, such as social media platforms, where user engagement is typically more exploratory or serendipitous. While the algorithms used in this study are commonly adopted in industry and the results are robust across alternative algorithms, if new methods or models emerge that better predict customers' current demand, the optimal design of the personalization stage remains an open question. Future research could study personalization that spans multiple stages over longer time horizons, across different platform types, and under more advanced algorithms.

References

Abraham, Mark and Edelman, David C (2024). "Personalization Done Right The five dimensions to consider and how AI can help". *Harvard Business Review*, 103 (11-12), 104–115.

Arapakis, Ioannis, Park, Souneil, and Pielot, Martin (2021). "Impact of response latency on user behaviour in mobile web search". In "Proceedings of the 2021 Conference on Human Information Interaction and Retrieval", 279–283.

Bleier, Alexander and Eisenbeiss, Maik (2015). "Personalized online advertising effectiveness: The interplay of what, when, and where". *Marketing Science*, 34 (5), 669–688.

Bronnenberg, Bart J, Kim, Jun B, and Mela, Carl F (2016). "Zooming in on choice: How do consumers search for cameras online?" *Marketing Science*, 35 (5), 693–712.

Burges, Christopher JC (2010). "From ranknet to lambdarank to lambdamart: An overview". *Learning*, 11 (23-581), 81.

Chen, Guangying, Chan, Tat, Zhang, Dennis, Liu, Senmao, and Wu, Yuxiang (2024). "The effects of diversity in algorithmic recommendations on digital content consumption: A field experiment". *Available at SSRN 4365121*.

Chen, Yuxin and Yao, Song (2017). "Sequential search with refinement: Model and application with click-stream data". *Management Science*, 63 (12), 4345–4365.

Chen, Zirou, Shi, Mengze, and Zhong, Zemin (Zachary) (2025). "Predictive Accuracy, Consumer Search, and Personalized Recommendation". *Available at SSRN 4298841*.

Choi, Hana and Mela, Carl F (2019). "Monetizing online marketplaces". *Marketing Science*, 38 (6), 948–972.

Compiani, Giovanni, Lewis, Gregory, Peng, Sida, and Wang, Peichun (2024). "Online search and optimal product rankings: An empirical framework". *Marketing Science*, 43 (3), 615–636.

Dang, Chu (Ivy), Ursu, Raluca, and Chintagunta, Pradeep K. (2024). "Going Back to Move Forward? How Search Revisits on a Website We Built, and in Field Data, Inform Us about Search Outcomes". *Available at SSRN 3626451*.

De los Santos, Babur and Koulayev, Sergei (2017). "Optimizing click-through in online rankings with endogenous search refinement". *Marketing Science*, 36 (4), 542–564.

Donnelly, Robert, Kanodia, Ayush, and Morozov, Ilya (2024). "Welfare effects of personalized rankings". *Marketing Science*, 43 (1), 92–113.

Fang, Xing, Kim, SunAh, and Chintagunta, Pradeep K (2025). "Too Many or Too Few? Information Cues in Recommender Systems and Consequences for Search and Purchase Behavior". *Journal of Marketing*.

Fong, Jessica (2024). "Effects of market size and competition in two-sided markets: Evidence from online dating". *Marketing Science*, 43 (5), 971–985.

Fong, Jessica, Manchanda, Puneet, and Song, Yu (2023). "How Effective is Suggested Pricing?: Experimental Evidence from an E-Commerce Platform". *Available at SSRN 4993457*.

Fong, Jessica, Natan, Olivia, and Pantle, Ranmit (2024). "Consumer Inferences from Product Rankings: The Role of Beliefs in Search Behavior". *Available at SSRN*.

Gandhi, Amit and Houde, Jean-François (2019). "Measuring substitution patterns in differentiated-products industries". *NBER Working paper*, (w26375).

Ghose, Anindya, Ipeirotis, Panagiotis G, and Li, Beibei (2012). "Designing ranking systems for hotels on travel search engines by mining user-generated and crowdsourced content". *Marketing Science*, 31 (3), 493–520.

Ghose, Anindya, Ipeirotis, Panagiotis G, and Li, Beibei (2019). "Modeling consumer footprints on search engines: An interplay with social media". *Management Science*, 65 (3), 1363–1385.

Goić, Marcel, Jerath, Kinshuk, and Kalyanam, Kirthi (2022). "The roles of multiple channels in predicting website visits and purchases: Engagers versus closers". *International Journal of Research in Marketing*, 39 (3), 656–677.

Greminger, Rafael P (2022). "Optimal search and discovery". *Management Science*, 68 (5), 3904–3924.

Gu, Chris and Wang, Yike (2022). "Consumer online search with partially revealed information". *Management Science*, 68 (6), 4215–4235.

Hodgson, Charles and Lewis, Gregory (2023). "You can lead a horse to water: Spatial learning and path dependence in consumer search". Technical report, National Bureau of Economic Research.

Holtz, David, Carterette, Ben, Chandar, Praveen, Nazari, Zahra, Cramer, Henriette, and Aral, Sinan (2020). "The engagement-diversity connection: Evidence from a field experiment on spotify". In "Proceedings of the 21st ACM Conference on Economics and Computation", 75–76.

Honka, Elisabeth, Hortaçsu, Ali, and Vitorino, Maria Ana (2017). "Advertising, consumer awareness, and choice: Evidence from the US banking industry". *The RAND Journal of Economics*, 48 (3), 611–646.

Honka, Elisabeth, Hortaçsu, Ali, and Wildenbeest, Matthijs (2019). "Empirical search and consideration sets". In "Handbook of the Economics of Marketing", volume 1, 193–257. Elsevier.

Honka, Elisabeth, Seiler, Stephan, and Ursu, Raluca (2024). "Consumer search: What can we learn from pre-purchase data?" *Journal of Retailing*.

Hosanagar, Kartik, Fleder, Daniel, Lee, Dokyun, and Buja, Andreas (2014). "Will the global village fracture into tribes? Recommender systems and their effects on consumer fragmentation". *Management Science*, 60 (4), 805–823.

Jameei Osgouei, Ata, Ching, Andrew T, Ratchford, Brian T, and Shahrokhi Tehrani, Shervin (2023). "Estimating Position and Social Influence Effects in Online Search: An Empirical Generalized Weitzman Model". *Available at SSRN 4545610*.

Kaye, Aaron P (2024). "The Personalization Paradox: Welfare Effects of Personalized Recommendations in Two-Sided Digital Markets".

Kim, Jun B, Albuquerque, Paulo, and Bronnenberg, Bart J (2010). "Online demand under limited consumer search". *Marketing science*, 29 (6), 1001–1023.

Korganbekova, Malika and Zuber, Cole (2023). "Balancing user privacy and personalization". *Working Paper*.

Koulayev, Sergei (2014). "Search for differentiated products: identification and estimation". *The RAND Journal of Economics*, 45 (3), 553–575.

Lambrecht, Anja and Tucker, Catherine (2013). "When does retargeting work? Information specificity in online advertising". *Journal of Marketing Research*, 50 (5), 561–576.

Li, Hongshuang and Ma, Liye (2020). "Charting the path to purchase using topic models". *Journal of Marketing Research*, 57 (6), 1019–1036.

Li, Xitong, Grahl, Jörn, and Hinz, Oliver (2022). "How do recommender systems lead to consumer purchases? A causal mediation analysis of a field experiment". *Information Systems Research*, 33 (2), 620–637.

Lu, Zipei and Kannan, PK (2024). "Measuring the synergy across customer touchpoints using transformers". *Available at SSRN 4684617*.

MacKenzie, Ian, Meyer, Chris, and Noble, Steve (2013). "How retailers can keep up with consumers". *McKinsey & Company*, 18 (1), 1–10.

Ostuni, Vito, Kofler, Christoph, Nilange, Manjesh, Lamkhede, Sudarshan, and Zylberglejd, Dan (2023). "Search personalization at netflix". In "Companion Proceedings of the ACM Web Conference 2023", 756–758.

Padilla, Nicolas, Ascarza, Eva, and Netzer, Oded (2024). "The customer journey as a source of information". *Quantitative Marketing and Economics*, 1–40.

Pariser, Eli (2011). *The filter bubble: How the new personalized web is changing what we read and how we think*. Penguin.

Rayner, Keith (1998). "Eye movements in reading and information processing: 20 years of research." *Psychological bulletin*, 124 (3), 372.

Simonson, Itamar (2005). "Determinants of customers' responses to customized offers: Conceptual framework and research propositions". *Journal of marketing*, 69 (1), 32–45.

Song, Michelle (2021). "How Do Personalized Recommendations Affect Consumer Exploration: A Field Experiment". *SSRN Electronic Journal*.

Ursu, Raluca, Seiler, Stephan, and Honka, Elisabeth (2024). "The sequential search model: A framework for empirical research". *Quantitative Marketing and Economics*, 1–49.

Ursu, Raluca M (2018). "The power of rankings: Quantifying the effect of rankings on online consumer search and purchase decisions". *Marketing Science*, 37 (4), 530–552.

Wan, Xiang, Kumar, Anuj, and Li, Xitong (2024). "How do product recommendations help consumers search? Evidence from a field experiment". *Management Science*, 70 (9), 5776–5794.

Wei, Yanhao and Jiang, Zhenling (2024). "Estimating parameters of structural models using neural networks". *Marketing Science*.

Weitzman, Martin L. (1979). "Optimal Search for the Best Alternative". *Econometrica*, 47 (3), 641–654.

Yavorsky, Dan, Honka, Elisabeth, and Chen, Keith (2021). "Consumer search in the US auto industry: The role of dealership visits". *Quantitative Marketing and Economics*, 19, 1–52.

Yoganarasimhan, Hema (2020). "Search personalization using machine learning". *Management Science*, 66 (3), 1045–1070.

Yoon, Song-Oh and Simonson, Itamar (2008). "Choice set configuration as a determinant of preference attribution and strength". *Journal of Consumer Research*, 35 (2), 324–336.

Yuan, Zhe, Chen, AJ Yuan, Wang, Yitong, and Sun, Tianshu (2025). "How recommendation affects customer search: A field experiment". *Information Systems Research*, 36 (1), 84–106.

Zhan, Ruohan, Han, Shichao, Hu, Yuchen, and Jiang, Zhenling (2024). "Estimating Treatment Effects under Recommender Interference: A Structured Neural Networks Approach". *arXiv preprint arXiv:2406.14380*.

Zhang, Luna, Ursu, Raluca, Honka, Elisabeth, and Yao, Oliver (2023). "Product discovery and consumer search routes: Evidence from a mobile app". *Available at SSRN*, 4444774.

Appendix

A Randomization Check, Manipulation Check, and Interference

A.1 Randomization Check

For the personalized recommendation experiment, I compare user activity in the 60 days prior to the experiment, which corresponds to the period used to generate personalized recommendations. I conduct a t-test to check the balance between the treatment and control conditions. As shown in Table A.1 Panel (A), I find no systematic differences between the two conditions in recommendation-stage clicks, query-stage clicks, and total orders.

For the personalized search experiment, I analyze user behavior over the 30 days preceding the experiment, including whether the user was active on Mercari, the number of queries they submitted, and their total spending. As shown in Table A.1 Panel (B), there are no systematic differences between the treatment and control conditions.

Table A.1: Randomization Check

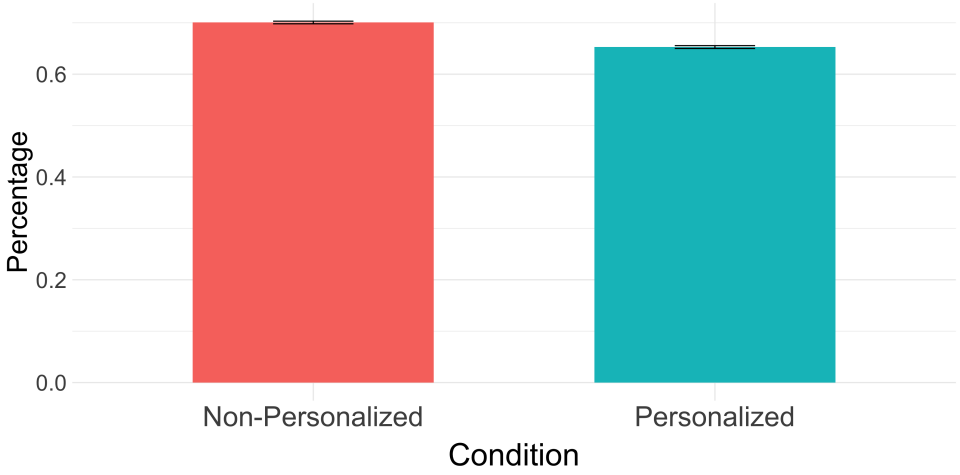
Panel (A): Personalized Recommendation Experiment					
	Non-Personalized		Personalized		
	Mean	SE	Mean	SE	p-value
Recommendation clicks	2.107	0.008	2.107	0.008	0.986
Search clicks	27.784	0.059	27.787	0.059	0.968
Total Orders	0.008	0	0.008	0	0.88
Panel (B): Personalized Search Experiment					
	Non-Personalized		Personalized		
	Mean	SE	Mean	SE	p-value
Whether active	0.799	0.001	0.801	0.001	0.188
Searches	1.907	0.004	1.914	0.004	0.246
Spending	0.775	0.0037	0.773	0.004	0.65

Notes. Panel (A) presents the summary statistics of user activities in the past 60 days before the personalized home recommendation experiment. Panel (B) presents the summary statistics of user activities in the past 30 days before the personalized search experiment. Both searches and spending use $\log(x+1)$ transformation.

A.2 Manipulation Check

Recall that in the personalized recommendation experiment, customers assigned to the non-personalized condition are shown products primarily selected based on popularity. These popular products are likely to belong to category-brand pairs with high historical revenue. To test this, I compute the percentage of items belonging to top-selling category-brand combinations among the top 12 recommended items (i.e., those on the initial screen).[37] In Figure A.1, I report the percentage of items that belong to the top 10% of brands with the highest revenue within each category. As expected, the non-personalized condition features a higher proportion of items from high-revenue category-brand pairs compared to the personalized condition.

Figure A.1: Manipulation Check of Personalized Recommendation



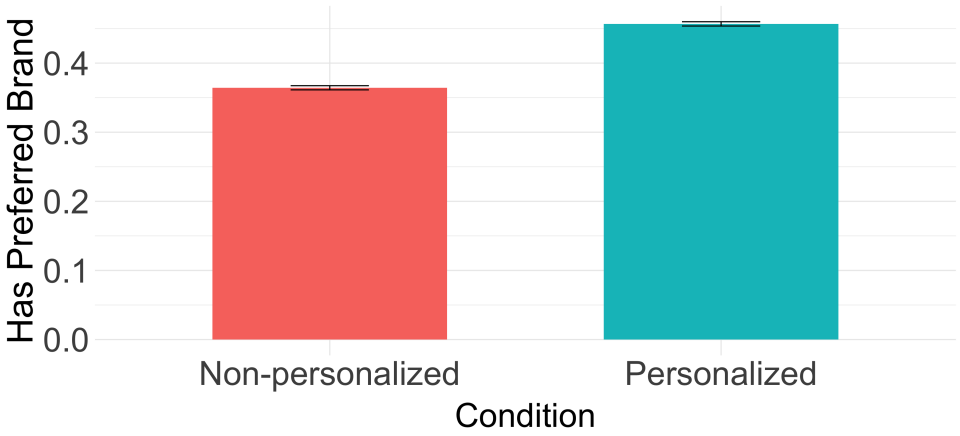
Notes: This figure plots the average value of the percentage of items that belong to the top 10% brands with the highest revenue in a category. Error bars represent 95% confidence intervals.

In the personalized search experiment, I examine whether the ranking aligns with customers' inferred preferences. Section 4.2 presents results for price as an illustrative example.

[37] Because the experiment was conducted over two years ago, not all recommendation impressions have been retained due to data storage limitations. Impressions were only recorded if users interacted with the recommendations (e.g., clicked on a listing or switched tabs), which may not necessarily indicate a click on a specific item. Therefore, this manipulation check is limited to sessions where impressions were recorded. However, the main experimental results from these sessions remain consistent with the full-sample results reported in Section 4.1.2.

Here, I present the analysis of brand preferences inferred from users' past transactions. Figure A.2 reports the average value of an indicator variable equal to one if at least one item among the top 12 search results (i.e., the initial search screen) corresponds to the user's inferred preferred brand. The results show that customers in the personalized condition are more likely to be shown items from their preferred brands compared to those in the non-personalized condition.

Figure A.2: Display of Preferred Brands Across Experimental Conditions



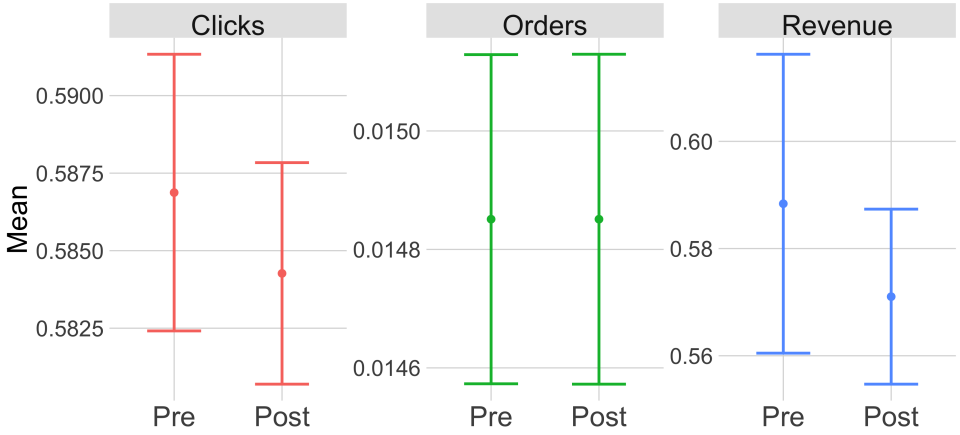
Notes: This figure plots the average value of an indicator variable that equals one if at least one item in the top 12 search results matches the user's inferred brand preference. The analysis focuses on the initial screen (i.e., the first 12 items) of the first search session for each user during the experiment. Error bars represent 95% confidence intervals.

A.3 Interference Discussion

Personalized Recommendation Experiment SUTVA is violated (i.e., there is "interference") when the treatment assignment of one user influences the outcomes of other users in the experiment. To check potential interference, I focus on users in the control condition and compare their activity before and after the experiment. If there is no interference, control users should exhibit consistent patterns in clicks, orders, and revenue across both periods, because they do not experience any change. Specifically, I compare each control user's behavior in the recommendation stage prior to the experiment with their first ses-

sion following the experiment's launch. Figure A.3 shows no significant differences in these outcomes, thereby mitigating concerns about interference.

Figure A.3: Comparisons of Users in Control Condition Pre- and Post-Experiment



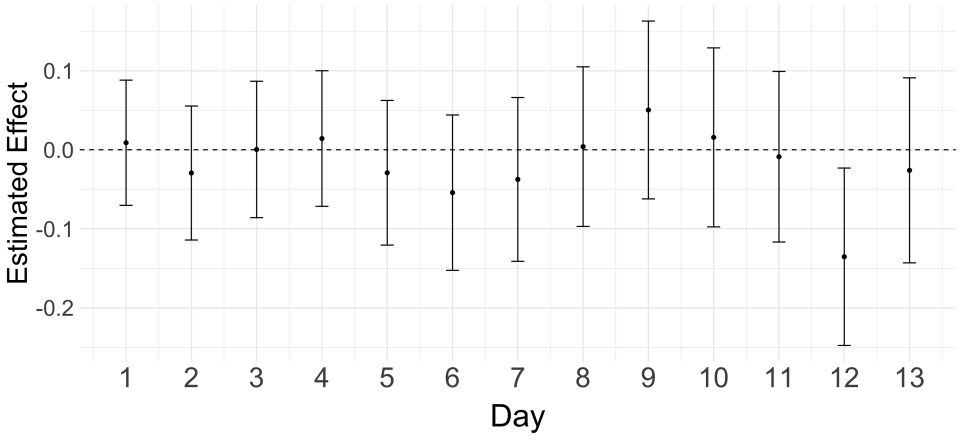
Notes. This figure shows the number of clicks, orders, and total revenue at the recommendation stage for users in the control condition, comparing behavior before and after the experiment. The pre-period captures each user's last activity within the two weeks leading up to the experiment, and the post-period corresponds to their first recommendation engagement afterward. The analysis includes only users who see recommendations in both periods. The error bar represents the 95% confidence interval.

In addition, I examine the treatment effects of personalization over time by analyzing results separately by day. If interference were present, I would expect treatment effects to vary across days, as the likelihood of interference increases with more users being assigned to the experiment (e.g., Fong et al., 2023). However, as shown in Figure A.4, the treatment effects remain stable across days, further alleviating concerns about interference.

Personalized Search Experiment First, in this experiment, only a small share of users were assigned to each experimental condition: 10% to the personalized search condition and 10% to the non-personalized search condition. This low saturation rate reduces the likelihood of substantial interference bias (e.g., Fong, 2024).

Second, following the approach used in the personalized recommendation experiment above, I test for interference by examining users who are consistently exposed to the same ranking algorithm before and after the experiment. Specifically, I focus on the 70% of users

Figure A.4: Treatment Effects of Personalized Recommendation by Day

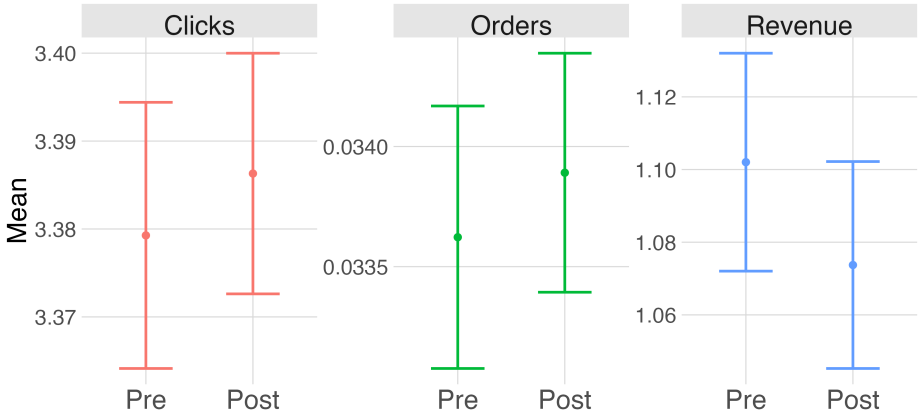


Notes. These figures plots the the estimation results of β^t in the following equation: $Revenue = \beta_0 + \beta_1 I_t + \beta_2 Personalized_t + \sum_t \beta^t I_t \times Personalized_i + \epsilon_{it}$, where I_t indicates the day. The omitted day is the first day (Day = 0). Dependent variables are all logged. Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

assigned to the status-quo non-personalized ranking in both periods. I compare their click, order, and revenue outcomes before and after the experiment. As shown in Figure A.5, these metrics remain stable in both periods, providing evidence against the presence of interference.

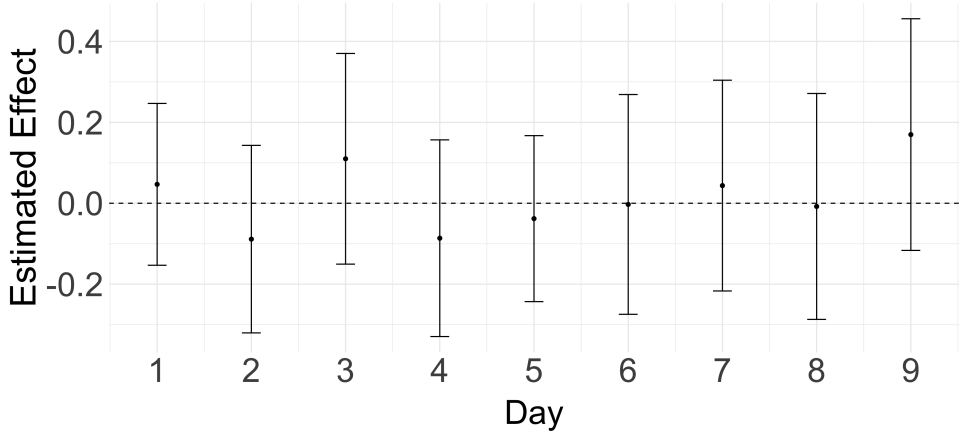
Third, similarly, I estimate the daily treatment effects of personalized query-stage ranking. Figure A.6 plots these estimates for each day during the experiment. None of the effects differ significantly from those on the first day, further minimizing concerns about interference.

Figure A.5: Comparisons of Users in Status-Quo Condition Pre- and Post-Experiment



Notes. This figure presents the number of clicks, orders, and total revenue at the query stage for users in the status-quo condition. The pre-period reflects each user's last activity within the 12 days preceding the experiment (matching the duration of the personalized search field experiment), and the post-period captures their first session. The analysis is restricted to users who conduct a search query in both periods. The error bar represents the 95% confidence interval.

Figure A.6: Treatment Effects of Personalized Search by Day



Notes. These figures plots the the estimation results of β^t in the following equation: $Revenue = \beta_0 + \beta_1 I_t + \beta_2 Personalized_t + \sum_t \beta^t I_t \times Personalized_i + \epsilon_{it}$, where I_t indicates the day. Standard errors are in parentheses, clustered at the individual level. The omitted day is the first day (Day = 0). * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

B Spillovers from Recommendation to Query Stage

I use the experimental condition in the personalized recommendation experiment as an instrumental variable to examine the relationship between the recommendation and query stages. The experimental condition exogenously influences outcomes in the recommendation stage and is uncorrelated with unobserved factors that affect outcomes in the query stage. Furthermore, it impacts the query stage only indirectly, through its effect on the recommendation stage.

I estimate the following two-stage equations:

$$Y_i^{Rec} = \kappa_0 + \kappa_1 Personalized_i + \xi_i, \tag{8}$$

$$Y_i^{Search} = \varphi_0 + \varphi_1 \hat{Y}_i^{Rec} + \epsilon_i, \tag{9}$$

where Y_i^{Rec} and Y_i^{Search} denote customer i's outcomes in the recommendation and query stages, respectively, and $\hat{Y}_i^{Rec}$ is the predicted value from Equation (8). I examine three dependent variables: the number of clicks, the number of orders, and logged revenue. Estimation results are presented in Table B.1. I find that an increase in recommendation-stage outcomes (clicks, orders, and revenue) leads to a decline in the corresponding outcomes in the query stage. This pattern suggests a negative spillover effect from the recommendation stage to the query stage.

Table B.1: Relationship between Recommendation Stage and Query Stage

	Search Clicks (1)	Search Orders (2)	Log(Search Rev) (3)
Constant	4.504***	0.0250***	0.0736***
	(0.0738)	(0.0024)	(0.0074)
Rec Clicks	-3.120***		
	(0.1723)		
Rec Orders		-0.7868***	
		(0.2444)	
Log(Rec Rev)			-0.8287***
			(0.2387)
1-stage F-stats	841.6	43.9	38.3
Observations	3,050,146	3,050,146	3,050,146

Notes. This table shows the estimation results of Equations (8) and Equations (9). Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

C Product Variety

I examine the variety of products presented to customers. I measure product variety using two standard metrics in the literature (e.g., Chen et al., 2024; Holtz et al., 2020): Shannon entropy and the Herfindahl-Hirschman Index (HHI).

Shannon entropy quantifies the dispersion of a distribution. For a discrete random variable Y with support $\{y_1, \ldots, y_n\}$ and associated probabilities $P(y_i)$, entropy is defined as:

$$H(Y) = -\sum_{i=1}^{n} P(y_i) \log_2 P(y_i), \tag{10}$$

where the logarithm is base 2. Entropy reaches its maximum under a uniform distribution, reflecting maximal variety, and decreases as the distribution becomes more concentrated.

HHI captures concentration by summing the squared probabilities:

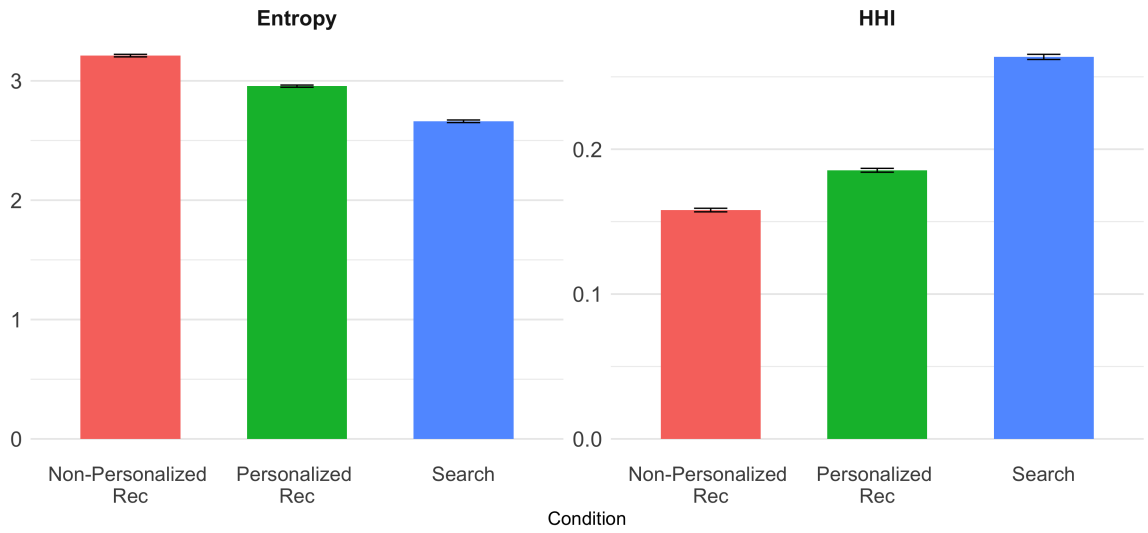
$$HHI(Y) = \sum_{i=1}^{n} \left(P(y_i)\right)^2. \tag{11}$$

Higher HHI values indicate greater dominance by a cluster and thus lower product variety.

Using Equations (10) and (11), I compute Shannon entropy and HHI at the session level, focusing on the top 12 items shown on the initial screen for both stages. I define each brand-physical condition combination as a cluster.

Figure C.1 presents these two metrics for the personalized recommendation stage, the non-personalized recommendation stage, and the query stage. As shown, personalized recommendations reduce product variety, as indicated by lower entropy and higher HHI. Even with personalization, the recommendation stage still exhibits greater variety than the query stage, suggesting that recommendations expose users to a broader range of products.

Figure C.1: Shannon Entropy and HHI



Notes. This figure presents the Shannon entropy and HHI of the top 12 products by session, using data from the personalized recommendation experiment. Higher entropy and lower HHI indicate greater variety.

D Ranking Model at the Query Stage

In the re-ranking stage, I employ LambdaMART (Burges, 2010), a widely used pairwise learning-to-rank algorithm that combines the LambdaRank framework with Multiple Additive Regression Trees (MART). LambdaMART leverages gradient-boosted decision trees and uses a cost function derived from LambdaRank to optimize ranking performance. As one of the most effective learning-to-rank algorithms, LambdaMART has been widely adopted in both academic research and industry applications. In marketing literature, Yoganarasimhan (2020) finds that LambdaMART outperforms alternative ranking models in terms of improvements in Normalized Discounted Cumulative Gain (NDCG) metrics. In industry, LambdaMART powers Microsoft's Bing search engine.[38]

Training Data The training data includes item-level, contextual-level and user-level (for the personalized model) features. Item-level features capture static attributes (e.g., category, brand) and dynamic interaction metrics over the past 30 days (e.g., cumulative likes up to the search date). Contextual-level features are the signals that capture the market state when a query is submitted, such as real-time inventory of relevant products. User-level features include behavioral variables (e.g., number of item views in the past 30 days) and demographic information (e.g., inferred age). The training sample comprises all user search activities on the platform during a two-week period.

LambdaMART uses a ranking score to guide its prediction of item relevance. I construct this score based on observed user interactions, assigning greater weights to more meaningful actions such as purchases. This scoring system captures varying levels of user engagement, allowing the model to prioritize items that reflect stronger user interest. The weights are calibrated using historical user activity data.[39]

[38] Microsoft Research, "RankNet: A Ranking Retrospective," `https://www.microsoft.com/en-us/research/blog/ranknet-a-ranking-retrospective/`, accessed June 2025.

[39] Due to an agreement with Mercari, I cannot disclose the exact scoring function.

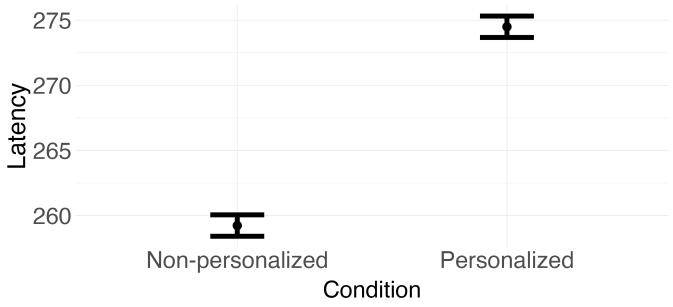
Training Process The model is trained by tuning several hyperparameters to balance flexibility and overfitting. These hyperparameters include regularization terms (which control model complexity), the learning rate (which determines the step size during optimization), and the number of leaves per tree (which affects the model's ability to capture interaction effects). I also tune additional parameters such as the maximum tree depth, the minimum number of data points per leaf, the number of estimators, the bagging fraction, and the feature subsampling ratios. The model is trained to optimize NDCG at the top 100 positions. To evaluate performance, the data are split into 80% for training and 20% for testing.

E Search Latency

The Effect of Personalization on Latency As shown in Figure E.1, in the personalized search experiment, average latency is higher in the personalized condition (274 milliseconds) compared to the non-personalized condition (259 milliseconds). According to the company's search engineering team, this increase is primarily due to the additional time required to retrieve and compute user-specific features, rather than the time needed for model inference.[40]

Since the conclusion of this experiment, the company has significantly expanded its infrastructure investments to improve system performance. These latency results are thus specific to my experimental period. However, they also highlight another cost of personalization—the continual infrastructure investments required to maintain responsiveness at scale.

Figure E.1: The Impact of Personalized Search on Search Latency



Notes: This figure compares search latency under personalized and non-personalized conditions from the personalized query-stage ranking experiment. The error bar represents the 95% confidence interval.

The Effects of Search Latency on Customers From the customer's perspective, increased search latency can lead to longer loading times, potentially degrading the overall shopping experience. To quantify the relationship between backend search latency and observed loading time, I estimate the following regression:

$$Y_k = \iota_0 + \iota_1 latency_k + \epsilon_k, \tag{12}$$

[40] As the personalized recommendation experiment was conducted more than two years prior to my data access, backend latency logs from that period are no longer available.

where $latency_k$ denotes the backend search latency (in milliseconds, log-transformed) for search session k, and Y_k represents the outcome of interest. This analysis uses data from the personalized search experiment. I first consider Y_k as the customer-observed loading time for session k. Since clickstream data is recorded at the second level, observed loading times are rounded to the nearest full second.[41] 37.8% of sessions have an observed loading time of 0 seconds, and 55.8% register a loading time of 1 second.[42] The estimated coefficient on latency, $\hat{\iota_1} = 0.20$ (standard error = 0.05), indicates a statistically significant and positive relationship: longer backend latency is associated with longer observed loading times.

Next, I investigate the impact of search latency on consumer engagement by estimating Equation (12) with three outcome variables: the number of clicks, the number of orders, and logged revenue per session. In Table E.1, a 1% increase in latency is associated with a 4.6% decrease in clicks, a 0.4% decrease in orders, and a 0.1% decrease in revenue. These findings demonstrate the economic significance of minimizing latency in personalized search systems.

Table E.1: Effects of Search Latency on Search Outcomes

	Clicks	Orders	Log(Rev)
	(1)	(2)	(3)
Constant	4.101***	0.0703***	0.2149***
	(0.1420)	(0.0091)	(0.0267)
Log(Latency)	-0.0464*	-0.0038**	-0.0130***
	(0.0258)	(0.0016)	(0.0048)
Observations	101,419	101,419	101,419
R^2	0.00004	0.0001	0.0001

Notes: This table reports the estimated effects of backend search latency (log-transformed) on consumer engagement in the query stage. Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

[41] The latency dataset is not directly linked to individual user sessions, so I match latency to customer search sessions based on timestamp and exclude ambiguous cases (e.g., concurrent searches by different users at the same timestamp).

[42] Sessions with loading times above 3 seconds (fewer than 1%) are excluded as outliers.

F Robustness Checks of Alternative Algorithm Design

F.1 Alternative Recommendation Model

I evaluate an alternative recommendation model, as the "quality" of recommendations may affect consumer decisions. To do so, I draw on a separate field experiment conducted by the company, in which a more advanced personalized recommendation algorithm was tested on the homepage. This new model represents a significant upgrade in several key dimensions. It adopts a transformer-based architecture that incorporates visual product representations (e.g., images) and expands the scope of user engagement signals beyond basic search and click behavior. Additionally, it introduces forecasted serendipity by predicting future user actions and tailoring recommendations with greater contextual relevance. These enhancements improve key performance metrics, including recall, precision, and NDCG.

The experiment evaluating this enhanced algorithm was conducted over a two-week period in Spring 2023. During the experiment, 50% of users were randomly assigned to the status quo personalized recommendation algorithm (the one used as the treatment condition in the experiment described in Section 4.1), while the remaining 50% of users received the more sophisticated recommendation algorithm.

I report the experimental results in Table F.1. I find that customers exposed to a more advanced recommendation system click on 45.5% more items, place 17.4% more orders, and generate 16.2% more revenue within the recommendation stage compared to those in the status quo personalized recommendation condition. However, they are 1.6% less likely to initiate a search query, click on 2% fewer items, place 3.7% fewer orders, and generate 2.8% less revenue in the query stage. The total orders and revenue across both stages remain unchanged.

Table F.1: Treatment Effects of More Advanced Personalized Recommendation

Variable	Personalized	More Personalized	Change (%)	t-statistic	p-value
Rec Clicks	0.216	0.247	14.245	19.799	< 0.001
Rec Orders	0.005	0.006	20.132	9.904	< 0.001
Rec Revenue	0.230	0.284	23.614	5.858	< 0.001
I(Initiate a Search)	0.600	0.592	-1.322	-12.595	< 0.001
Search Clicks	3.427	3.366	-1.796	-8.096	< 0.001
Search Orders	0.038	0.037	-3.823	-5.156	< 0.001
Search Revenue	1.691	1.651	-2.323	-1.721	0.085
Total Clicks	3.643	3.612	-0.846	-3.986	< 0.001
Total Orders	0.043	0.043	-1.056	-1.502	0.133
Total Revenue	1.921	1.936	0.783	0.607	0.544

Notes. This table presents the treatment effects of a more advanced personalized recommendation algorithm. I report the mean values of outcomes under both the control condition (baseline personalization) and the treatment condition (personalization using a more advanced algorithm). The reported difference reflects the mean outcome in the treatment condition minus that in the control condition. The final two columns display the corresponding t-statistics and p-values from two-tailed tests for differences in means.

F.2 Alternative Ranking Model

The effectiveness of personalized ranking also depends in part on the "quality" of the underlying ranking algorithm. For instance, if the baseline non-personalized algorithm is already highly optimized, the marginal benefit of personalization may be limited. In contrast, greater gains may arise when the baseline algorithm has more room for improvement.

I use the status quo ranking model as the base algorithm and develop a personalized counterpart that additionally incorporates user-specific features. Recall that in the personalized search experiment, among the remaining 80% of users not included in the experimental conditions, 70% were randomly assigned to a non-personalized status quo ranking model and 10% were assigned to a personalized version of the same model that integrates user-specific features.

Table F.2 reports the treatment effects of personalization under the status quo ranking algorithm. Personalization leads to statistically significant improvements across all query-stage outcomes: clicks increase by 1.7%, orders by 6.9%, and revenue by 7.3%. These results convey two main takeaways. First, the choice of ranking algorithm itself influences user engagement and platform revenue. For example, the LambdaMART and status quo models

produce different revenue outcomes. Second, holding the algorithm constant, incorporating user-level data can enhance overall revenue.

Table F.2: The Effects of Personalized Query Ranking (Alternative Model)

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Search Clicks	2.464	2.506	1.670	3.209	0.001
Search Orders	0.025	0.027	6.912	3.799	< 0.001
Search Revenue	0.866	0.929	7.334	2.220	0.026
Rec Clicks	0.821	0.824	0.392	0.565	0.572
Rec Orders	0.002	0.002	-2.873	-0.447	0.655
Rec Revenue	0.045	0.050	11.195	0.775	0.438
Total Clicks	3.285	3.329	1.350	3.182	0.001
Total Orders	0.027	0.029	6.324	3.601	< 0.001
Total Revenue	0.911	0.980	7.526	2.337	0.019

Notes. This table presents the treatment effects of personalized query rankings based on the status quo algorithm. I report the mean values of outcomes for the control condition (non-personalized status quo model) and the treatment condition (personalized status quo model). The difference reflects the mean difference between the treatment and control conditions. The final two columns report t-statistics and p-values from two-tailed tests for differences in means.

G Inferring Demand States via Query Specificity

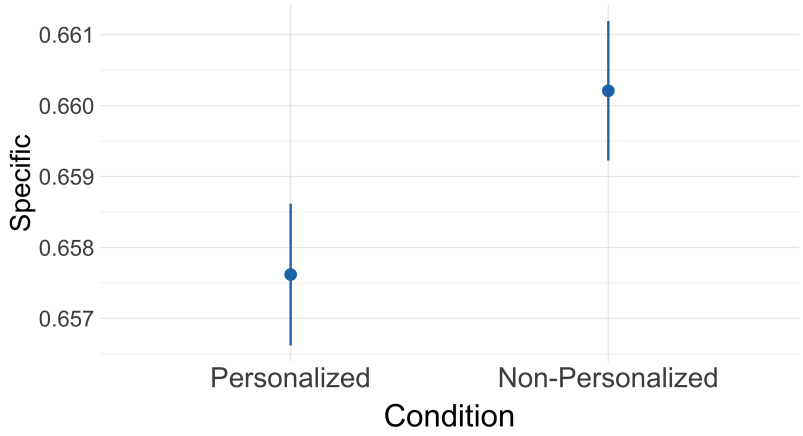
Yuan et al. (2025) identify two distinct customer states when visiting a platform: demand fulfillment and demand formation. Customers in the fulfillment state have well-defined needs and tend to compensate for less relevant recommendations by submitting more specific search queries. In contrast, customers in the demand formation state rely more heavily on recommendations, and reduced recommendation relevance leads them to submit more generic queries. These contrasting patterns offer a lens for inferring whether a customer is in a demand formation or fulfillment state. I adopt this framework to identify customer states.

I examine how personalization in recommendations influences the specificity of users' search queries. I follow the rule-based approach outlined in Yuan et al. (2025) to measure query specificity. Each query is decomposed into two types of components: generic words and decoration words. Generic words refer to terms that describe a product's basic type or category (e.g., *jeans*, *headphones*), while decoration words include descriptive modifiers such as brand (*Lululemon*, *LEGO*), color (*yellow*, *red*), physical condition (*new*), material (*cotton*, *leather*), or size (*plus size*). I compile a comprehensive dictionary of decoration terms derived from the product attributes available on the platform.

A query is classified as *generic* if it contains only generic words, and as *specific* if it includes at least one decoration word. For example, the query *"black UGG boots"* contains a generic term (*boots*) along with decoration words (*UGG* as the brand and *black* as the color), and is categorized as specific.

In Figure G.1, I present the specificity of search queries for users in the personalized and non-personalized conditions. The figure shows that users in the non-personalized condition tend to submit more specific queries. This pattern suggests that, on average, customers arrive with well-defined demand and compensate for less personalized recommendations by refining their search query inputs.

Figure G.1: The Impact of Personalized Recommendations on Search Query Specificity



Notes: This figure presents the search query specificity in the personalized and non-personalized conditions from the personalized recommendations experiment. The error bar represents the 95% confidence interval.

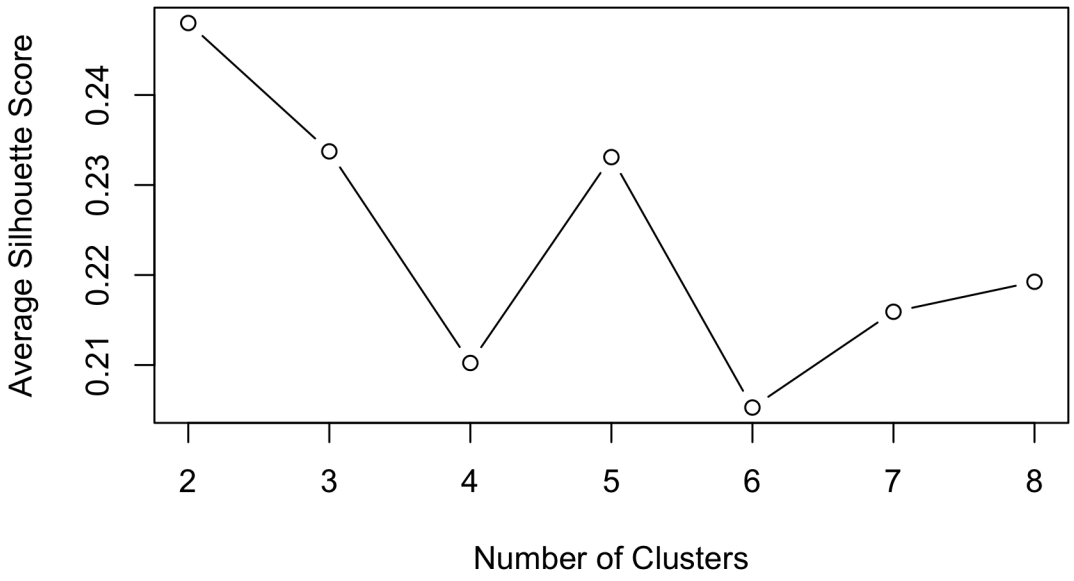
H Belief Estimation

H.1 Conducting a Search Query

To improve the precision of the estimated empirical distribution, I first group similar queries using natural language processing techniques. Specifically, I first embed every query string into a TF-IDF vector space over unigrams and bigrams. I then run k-means clustering on those embeddings. Next, I choose the number of clusters by maximizing the average silhouette score under a cosine-distance metric. The silhouette score measures how similar an object is to its own cluster compared to other clusters. For example, "PS5" and "PlayStation 5" are treated as the same query. These procedures ultimately give me a total of 39 query groups.

I then empirically estimate the distribution of product attributes for each query group across sessions. For customers in the non-personalized condition, expectations are formed based on the empirical distribution of products observed across all non-personalized users. For those in the personalized condition, I segment them based on their historical activities, including preferences (e.g., price level, brand), activity levels (e.g., item clicks in the past 30 days), purchase histories, and inferred demographic age. Customers' expectations are aligned with the empirical distribution of products seen by customers within their respective segments. I again apply k-means clustering to group individuals in the personalized condition based on observed user attributes and use the silhouette scores to assess the quality of the clustering. I iterate the number of clusters from 2 to 8 and find that the silhouette score is maximized when the segment is 2 (Figure H.1). Based on the inferred segments, customers in one group are generally more active, with greater past engagement (e.g., item clicks, searches, and purchases), while those in the other group are less active. For example, the median number of purchased products since account creation is 7 for the less active segment and 49 for the more active segment. Given the relatively small number of customers (2,768 in the personalized condition in my estimation sample), the two-cluster solution seems to be

Figure H.1: Silhouette Score of Clusters



reasonable. I would expect the optimal number of clusters to be larger when considering all customers on the platform.

Finally, I use bootstrapping to sample 48 products according to their query and personalization condition (and segment if in the personalized condition) from the distribution.[43]

H.2 Inspecting a Product

I estimate the following equation to examine the relationship between a product's hidden attributes and its visible attributes:

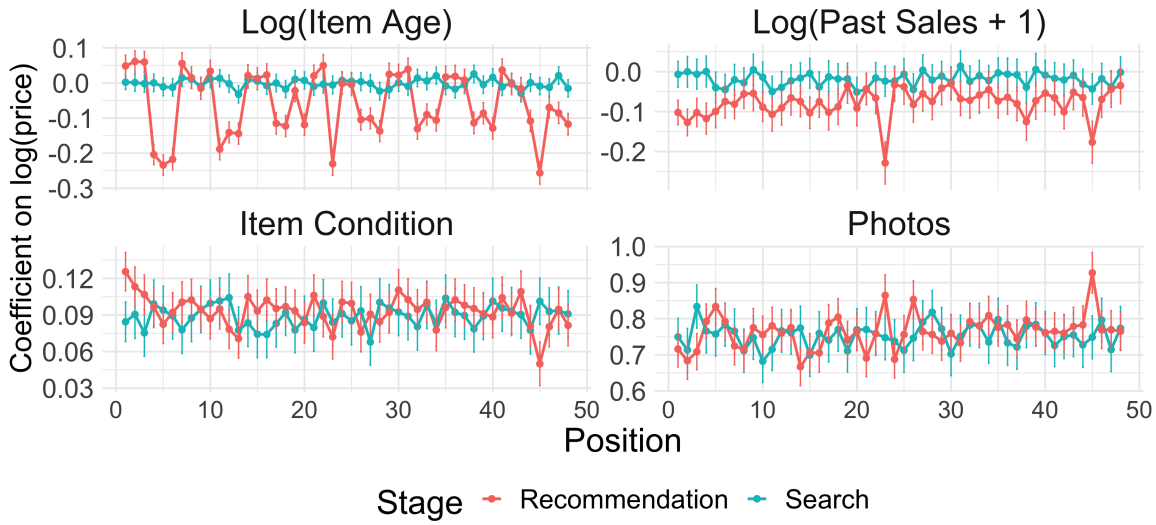
$$Z_j = \phi_0 + \phi_1 \log(price_j) + category_j + brand_j + \eta_j, \tag{13}$$

[43] Top 48 products account for 74% of clicks and 80% of purchase.

where Z_j are the hidden attributes, including the number of photos, item physical condition (measured on a scale from 1 to 5, where 1 indicates poor condition and 5 indicates new), item age and its seller's past sales. The visible attributes include price, category, and brand. The error term η_j is assumed to have a mean of zero.

Figure H.2 illustrates the estimation results by plotting the estimated (logged) price coefficients. The results suggest that higher-priced items tend to include more photos, be in better condition, and be listed by less experienced sellers (i.e., those with fewer past sales). In addition, the coefficients exhibit variation across both stages and positions.

Figure H.2: Beliefs about Hidden Attributes



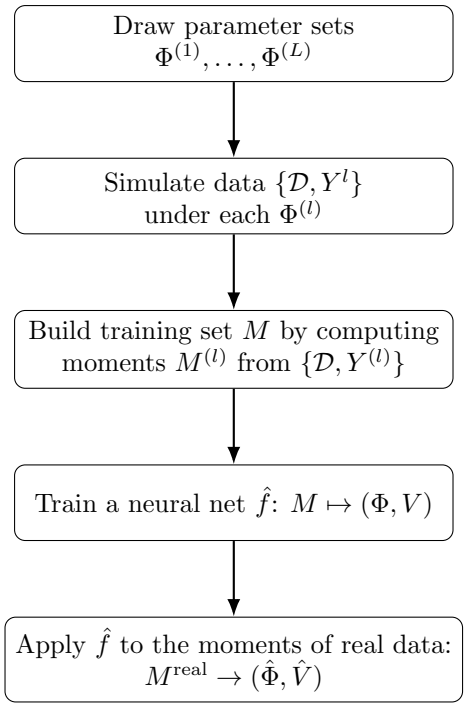
Notes. This figure displays the estimated price coefficients from Equation (13) for the recommendation stage and query stage respectively. Error bars represent 95% confidence intervals.

I Search Model Estimation

I.1 Model Estimation using Neural Networks

I estimate the search model using the NNE approach developed by Wei and Jiang (2024), which leverages a neural network to learn a mapping from datasets to the structural parameters of the econometric model. Figure I.1 illustrates the steps to train a neural network to recover the parameter vector $\boldsymbol{\Theta}$.

Figure I.1: Neural Network Training Illustration



The process can be described as follows.

1. **Draw Parameter Sets:** Generate L independent draws of model parameters $\boldsymbol{\Phi}^{(1)}, \ldots, \boldsymbol{\Phi}^{(L)}$ uniformly from a predefined parameter space. In my application, I set $L = 20{,}000$.

2. **Simulate Data:** For each draw $\boldsymbol{\Phi}^{(l)}$, simulate consumer behavior using the structural search model to generate data $\{\mathcal{D}, Y^{(l)}\}$. The dataset $\mathcal{D}$ includes item attributes $(\boldsymbol{X}, \boldsymbol{Z})$ and cost variables (Pos, $Personalized$). The simulated outcomes $Y^{(l)}$ include inspection, query submission, and purchase.

3. **Compute Moments:** From each simulated dataset $\{\mathcal{D}, Y^{(l)}\}$, compute a vector of data moments $M^{(l)}$ that summarize the key data patterns. I construct the following data moments at the option-level (including items and query), session-level, and individual-level: means and variances of $Y^{(l)}$ and cross-covariances between $\mathcal{D}$ and $Y^{(l)}$. As suggested by Wei and Jiang (2024), NNE is robust to redundant moments, as the neural net can learn from the training set whether a moment contributes to identifying the parameters.

4. **Train a Neural Network:** Collect all moment-parameter pairs $(M^{(l)}, \mathbf{\Phi}^{(l)})_{l=1}^{L}$ into a training set, where M is an input to the neural net and $\mathbf{\Phi}$ is the output. The neural network is trained to learn the inverse mapping from empirical moments M to model parameters $\mathbf{\Phi}$ and covariance matrix $\boldsymbol{V}$. Following Wei and Jiang (2024), I use a cross-entropy loss to teach the network to map from M back to $(\mathbf{\Phi}, \boldsymbol{V})$. The neural network architecture is defined using a sequence of layers. The input layer normalizes the input data using symmetric rescaling. This is followed by a fully connected layer with 64 hidden nodes and a ReLU activation function. The output layer is another fully connected layer, with the number of outputs matching the dimensionality of the output data. The initial learning rate is set to 0.001 and updated using a piecewise schedule, decreasing every 1,000 iterations. The network is trained using the Adam optimizer over a maximum of 500 epochs with a mini-batch size of 500 samples. I use a 90/10 split for training and testing, and model validation is conducted every 500 iterations. As demonstrated by Wei and Jiang (2024), the accuracy of NNE is not sensitive to the specific architecture or hyperparameter choices of the neural network.

5. **Apply to Real Data:** Compute the moments M^{real} on actual data $\{\mathcal{D}, Y^{\text{real}}\}$. Feed M^{real} into the trained network to obtain both the point estimate $\hat{\mathbf{\Phi}}$ and the statistical accuracy for the point estimate $\hat{\boldsymbol{V}}$.

I.2 Monte Carlo Simulation

I generate a dataset consisting of 3,000 customers and each customer has three sessions. In each session, customers follow a two-stage sequential search process: first a recommendation stage, then a query stage. Each stage displays 24 products. Customers are randomly assigned, with equal probability, to either a personalized or non-personalized search condition and are categorized into one of two segments (Segment 1 or Segment 2). For simplicity, product attributes and rankings are randomly generated.

Each product has three visible attributes: price, category and brand, and four hidden attributes: number of photos, product age, item condition, and sellers' past sales. Logged prices are drawn from a normal distribution: $\log(\text{Price}) \sim \mathcal{N}(3.5, 1)$. Category is a dummy variable indicating whether the item belongs to the same category as the customer's purchase intent. The probability that an item is in the relevant category is 0.3 in the recommendation stage and 0.9 in the query stage. There are five brands, each with the following probabilities of being chosen in the recommendation stage: 0.05, 0.15, 0.15, 0.20, and 0.25. In the query stage, the corresponding probabilities are 0.10, 0.02, 0.10, 0.05, and 0.08. The remaining probability mass in each stage corresponds to unbranded items. The number of photos is drawn from a discrete distribution over $\{1, 2, 3, 4, 5\}$ with probabilities $P = (0.10, 0.05, 0.25, 0.40, 0.30)$, respectively. Item condition is also sampled from a discrete distribution over $\{1, 2, 3, 4, 5\}$. Logged product age follows $\mathcal{N}^+(4, 1)$ and sellers past sales (logged) follow $\mathcal{N}^+(0, 1)$; both are normal distributions truncated below at zero.

Consumer utility depends on both visible and hidden product attributes. I follow the procedures described in Section 5.2.2 to estimate customers' beliefs about product utility conditional on visible attributes and beliefs about the utility of conducting a query. Customers also face two types of costs discussed in Section 5.2.3: (1) an inspection cost that varies with a product's position in the ranked list and (2) a query cost that depends on the personalization condition.

Customers begin their decision process in the recommendation stage, choosing whether

to inspect an item, conduct a search query, or exit (i.e., select the outside option). If they conduct a query, they leave the recommendation stage and proceed to the query stage, where they view a new set of 24 products and decide which to inspect. Ultimately, customers choose to purchase the most preferred product among all inspected products across both stages or opt for the outside option.

Table I.1 reports the results of the estimation. Column (1) lists the true parameter values, and Column (2) presents the corresponding estimates from the model. Overall, the estimated parameters align closely with the true values.

I.3 Model Fit

To evaluate model fit, I compare predicted and observed customer outcomes using parameter-free moments: the average number of inspections per session, the percentage of sessions with a search query, and the percentage of sessions with a purchase, each measured at both the session and individual levels. Table I.2 shows that the model captures these moments well.

Table I.2: Model Fit

Moments	Predicted	Observed
Session level		
Inspections	1.614	1.681
Query Submission Percentage	0.847	0.843
Orders	0.031	0.031
Individual level		
Inspections	4.192	4.376
Orders	0.067	0.070

Table I.1: Monte Carlo Simulation Results

Variables	(1) True	(2) Estimated
Preferences		
Mean Preferences		
Constant	-3.0	-3.026 (0.776)
Price	-0.4	-0.443 (0.091)
Same Category	0.8	0.833 (0.109)
Brand 1	0.4	0.434 (0.148)
Brand 2	0.4	0.449 (0.107)
Brand 3	0.5	0.555 (0.165)
Brand 4	0.3	0.349 (0.115)
Brand 5	0.1	0.141 (0.139)
N Photos	0.3	0.292 (0.090)
Item Age	-0.2	-0.238 (0.085)
Physical Condition	0.3	0.293 (0.092)
Seller Past Sales	0.3	0.296 (0.108)
Inspection Base (Recommendation)	-3.0	-3.260 (0.281)
Position	1.0	0.964 (0.137)
Inspection Base (Search)	-4.0	-3.841 (0.256)
Query Base	-1.5	-1.487 (0.244)
Personalization	0.8	0.879 (0.126)
Preferences Heterogeneity (SD)		
Price	0.2	0.200 (0.058)
N Photos	0.1	0.095 (0.058)
Item Age	0.1	0.100 (0.057)
Physical Condition	0.1	0.094 (0.058)
Seller Past Sales	0.1	0.101 (0.058)
Costs		
Inspection Base (Recommendation)	-3.0	-3.260 (0.281)
Position	1.0	0.964 (0.137)
Inspection Base (Search)	-4.0	-3.941 (0.256)
Query Base	-1.5	-1.587 (0.244)
Personalization	0.8	0.829 (0.126)

Notes. The estimation is based on 3,000 customers, each with 3 sessions (9,000 sessions in total). Standard errors are in parentheses.

J Utility-Based Recommendation and Query-Stage Ranking Models

Recommendation Model The platform recommends and ranks items to customers in decreasing order of the match index ν_{ij}^R:

$$\nu_{ij}^R = \omega^R u_{ij} + (1 - \omega^R)\bar{u}_{ij}, \tag{14}$$

where $u_{ij} = \boldsymbol{X_j'\alpha_i} + \boldsymbol{Z_j'\beta_i}$ represents the utility that customer i derives from product j, and $\bar{u}_{ij} = \boldsymbol{X_j'\bar{\alpha}} + \boldsymbol{Z_j'\bar{\beta}}$ denotes the average utility of product j based on mean preference coefficients across customers. The parameter ω^R captures the weight placed on individual-level utility, while $1 - \omega^R$ reflects the weight placed on average utility. A higher value of ω^R indicates greater emphasis on individual preferences, whereas a lower value implies that the platform prioritizes average customer preferences, which can be interpreted as a proxy for product popularity. In this framework, fully personalized recommendation corresponds to $\omega^R = 1$, non-personalized recommendation to $\omega^R = 0$, and partialy personalized recommendation to $\omega^R = 0.5$.

Query Ranking Model To construct query rankings for user i, the platform sorts items in decreasing order of the match index ν_{ij}^S:

$$\nu_{ij}^S = \omega^S \cdot u_{ij} + (1 - \omega^S) \cdot \bar{u}_{ij}, \tag{15}$$

where $u_{ij} = \boldsymbol{X_j'\alpha_i} + \boldsymbol{Z_j'\beta_i}$ denotes the utility that user i derives from product j, and $\bar{u}_{ij} = \boldsymbol{X_j'\bar{\alpha}} + \boldsymbol{Z_j'\bar{\beta}}$ represents the average utility of product j across users. The parameter ω^S captures the degree of personalization in the ranking at the query stage. Similarly, a fully personalized query-stage ranking corresponds to $\omega^S = 1$, a non-personalized ranking to $\omega^S = 0$, and a partially personalized ranking to $\omega^S = 0.5$.

K Additional Tables and Figures

Table K.1: User-Level Treatment Effects of Personalization Recommendation

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Rec Clicks	0.421	0.490	16.545	52.351	< 0.001
Rec Orders	0.011	0.012	6.044	7.914	< 0.001
Rec Revenue	0.537	0.565	5.044	3.758	< 0.001
Query Clicks	3.100	2.956	-4.632	-27.764	< 0.001
Query Orders	0.0224	0.0216	-3.665	-6.674	< 0.001
Query Revenue	0.789	0.750	-5.024	-5.315	< 0.001
Total Clicks	3.517	3.443	-2.115	-13.802	< 0.001
Total Orders	0.033	0.033	-0.322	-0.687	0.492
Total Revenue	1.306	1.294	-0.962	-1.159	0.246

Notes: The table shows the treatment effects of personalized recommendation on outcomes at the user level. I take the mean outcomes across sessions in the experiment for each user. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table K.2: User-Level Treatment Effects of Personalization Query-Stage Ranking

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Rec Clicks	0.791	0.793	0.255	0.320	0.749
Rec Orders	0.002	0.002	0.963	0.163	0.871
Rec Revenue	0.042	0.042	-0.582	-0.058	0.954
Query Clicks	2.242	2.303	2.708	5.132	< 0.001
Query Orders	0.023	0.025	9.222	4.775	< 0.001
Query Revenue	0.812	0.873	7.582	2.264	0.024
Total Clicks	3.032	3.094	2.048	4.595	< 0.001
Total Orders	0.025	0.027	8.680	4.674	< 0.001
Total Revenue	0.854	0.915	7.148	2.212	0.027

Notes: The table shows the treatment effects of personalized query stage on outcomes at the user level. I take the mean outcomes across sessions in the experiment for each user. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table K.3: Treatment Effects of Personalized Search on Recommendation Outcomes and Total Outcomes

Variable	Non-Personalized	Personalized	Change (%)	t-statistic	p-value
Rec Clicks	0.846	0.844	-0.300	-0.241	0.81
Rec Orders	0.002	0.002	-1.271	-0.140	0.888
Rec Revenue	0.045	0.045	0.968	0.060	0.952
Total Clicks	3.086	3.175	2.874	5.173	< 0.001
Total Orders	0.027	0.030	9.549	4.144	< 0.001
Total Revenue	0.883	0.982	11.231	3.098	0.002

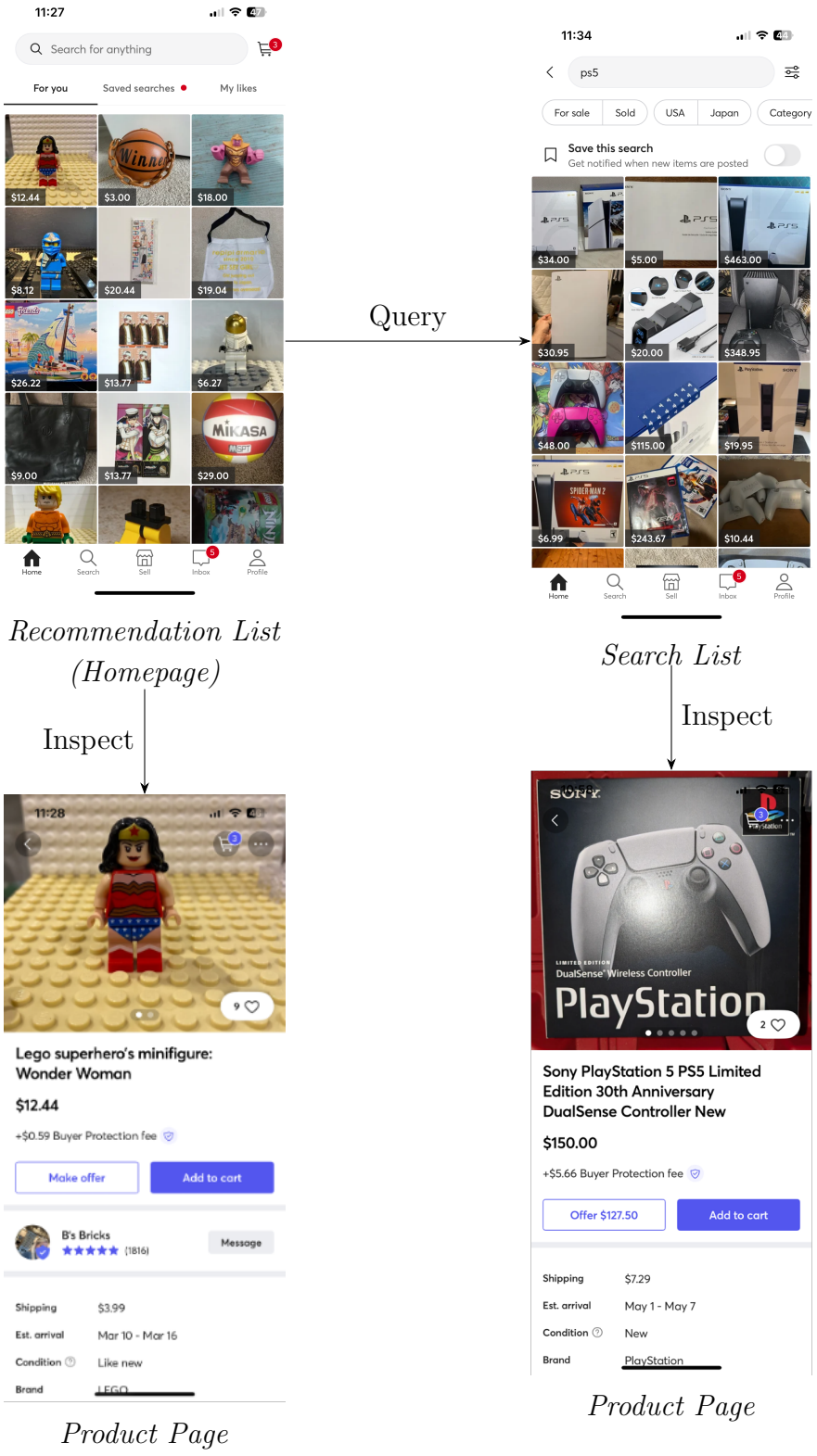
Notes: This table shows the treatment effects of the personalized recommendation. I report the mean of outcomes in both personalized and non-personalized conditions. The difference is measured using the outcomes in the personalized condition minus the outcomes in the non-personalized condition. The last two columns report t-statistics and p-values from the corresponding two-tailed tests for differences in means.

Table K.4: The Effects of Refinement on Customer Outcomes

	Orders	
	(1)	(2)
Constant	0.0459***	0.0477***
	(0.0008)	(0.0009)
Sorted	0.0443***	
	(0.0071)	
Personalized	0.0031***	0.0029**
	(0.0011)	(0.0012)
Sorted × Personalized	0.0028	
	(0.0093)	
Filtered		-0.0024
		(0.0024)
Filtered × Personalized		0.0018
		(0.0033)
Observations	187,818	187,818
R^2	0.0011	0.00005

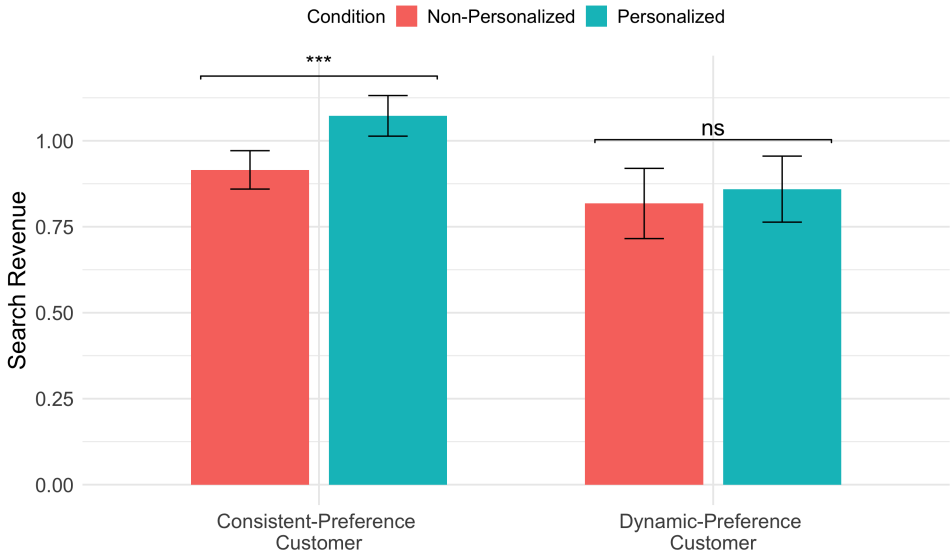
Notes: This table shows the effects of applying sorting and filters on the customer's purchase decisions. Standard errors are in parentheses, clustered at the individual level. * $p < 0.1$, ** $p < 0.5$, *** $p < 0.01$.

Figure K.1: An Illustration of Platform Layout



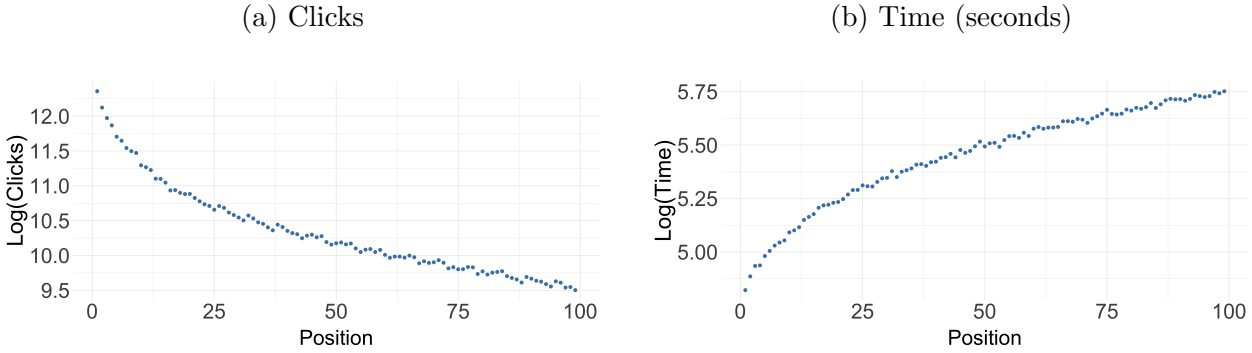
Notes. This figure shows an example of product display on Mercari in the recommendation stage and query stage. Customers see a product's image and price in the recommendation or search list. They can then inspect a product by clicking on it, which takes them to the product page.

Figure K.2: Heterogeneous Effects of Personalizing Query Stage



Notes: This figure plots the average revenue at the query stage for customers in the personalized and non-personalized conditions, separately for those with consistent and dynamic demand. The sample is restricted to customers who clicked more than one item in the 30 days prior to the personalized query stage experiment. Error bars represent 95% confidence intervals.

Figure K.3: Clicks and Time by Item Position



Notes. Panel (a) presents the logged number of clicks by search-result position; panel (b) presents the logged average time (in seconds) from the start of the search session to the click.