Trade-offs in Leveraging External Data Capabilities: A Field Experiment in Search

*

October 2024

Abstract

Firms increasingly leverage external entities' data capabilities to unlock improvements in their offerings, but measuring the impact of such capabilities is challenging. Collaborating with the search team at a technology company, we analyzed a large-scale field experiment where we randomized access to an external, leading search engine's autocomplete API for more than 2 million users over 108 days. We measure the causal effects of removing API access on two performance metrics of the focal company's search product: (a) clickthrough rate (CTR) on search suggestions and (2) CTR on the Search Engine Results Page. We find that, on average, compared to the baseline with API access, removing API access reduces the search-suggestion CTR by 4.6%. Further, exploiting the experimental variation, we use an instrumental variables approach to establish that a 10% increase (decrease) in CTR on search suggestions leads to a 1.85% increase (decrease) in CTR on top-slot search results. However, the negative effect of removing API access becomes less negative over time with the effect magnitude in the longer term being half what we would have obtained with a short-term experiment. We provide suggestive mechanism evidence of the longer-term effect: the focal company's reliance on the leading search engine's data capability tapers off the accumulation of internal data and then limits the improvement of its autocomplete predictions. This research informs managers of a critical trade-off in leveraging external data capabilities and sheds light on regulations such as the Digital Markets Act that mandate data sharing by large digital platforms.

*We thank Manuela Collis, Hong Deng, Anindya Ghose, John Lalor, Keyan Li, Shijie Lu, Alex Moehring, Abhishek Nagaraj, Christian Peukert, Mohammad Rahman, Yoonseock Son, Steve Tadelis, Sonny Tambe, Yun (Alicia) Wang, Joy Wu, Li Zheng, Shuang Zheng and participants at Notre Dame (Marketing), CMU (Tepper), University of Pittsburgh, UC Irvine (Merage), Nanyang Business School, SWUFE, Soochow University, UT Austin (McCombs), Georgia Tech (Scheller), Purdue (Daniels), Cornell (Dyson), UCSD, IU (Kelley), UIBE, Shenzhen University, NYU AI in Strategic Management Workshop, NSF Convergence Workshop on Human-AI Frontier, NBER Digital Economics Spring Meeting, China India Insights Conference, Conference on Information Systems and Technology, Marketing Dynamics Conference, Workshop on Information Systems and Economics, Hi! PARIS Workshop on AI and Digital Economy, TSE Online Platform Seminar Series, Columbia/Wharton Management, Analytics, and Data Conference, 11th Operations and Supply Chain Workshop, NBER Economics of Artificial Intelligence Conference, INFORMS Annual Meeting, 34th POMS Conference, and Marketing Science Conference for helpful suggestions.

1 Introduction

Large online firms collect an abundance of data to build their data-enabled capabilities, enhance their offerings, and shore up revenues (Duranton et al., 2021; Sun et al., 2024). For example, leading search engines devote significant resources to the development of query autocomplete (Cai et al., 2016), which generates relevant search suggestions in response to search queries before users reach the search engine results page (SERP) (Sullivan, 2018; Gulli, 2013). Increasingly, smaller players use Application Programming Interfaces (APIs) to gain access to the market leader's data capabilities[1] to unlock improvements in their offerings (Benzell et al., 2023; Xue et al., 2019; Li and Kettinger, 2021). In the context of search, Google Autocomplete API enables publishers and developers to retrieve search suggestions from Google, and Bing Autosuggest API enables smaller players such as DuckDuckGo to source search suggestions from Bing. Despite their prevalence in search markets, there is limited causal evidence of the impact of external data capabilities on the recipients' product performance. Measuring the impact of external data capabilities in search markets is also important from a policy perspective. For example, the Digital Markets Act requires leading search engines to provide smaller players with access to privacy-preserving search results with the aim of leveling the playing field.[2]

Theoretically, leveraging external data capabilities introduces a potential intertemporal trade-off for smaller players in search markets. On the one hand, queries can be ambiguous, broad, or heterogeneous (Sanderson, 2008), so smaller players often utilize external, established sources to generate predictions in response to those queries (Kopliku et al., 2014). As a result, access to external data capabilities can enhance performance with little risk for smaller players, especially at the early stage of product development (Rahmandad, 2012). On the other hand, over time, excessive reliance on external data capabilities might taper the accumulation of internal data, limiting smaller players' development of internal data capabilities in the long run (Gupta et al., 2006; Laverty, 1996).

In this paper, we analyze a large-scale field experiment to measure the impact of the *removal* of

[1] A note on terminology. We use general terms such as "external data capabilities" and "the market leader's data capabilities" and specific terms such as "the market leader's candidate items" interchangeably throughout the paper.

[2] For an overview of the Digital Markets Act, see: https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/digital-markets-act-ensuring-fair-and-open-digital-markets_en#documents.

access to a leading search engine's (i.e., the market leader's) autocomplete API on the performance of our partner company's search product (i.e., a smaller player; hereafter, "the company"). We further explore the heterogeneity and dynamics in user responses to the removal of API access. The primary outcome of interest is clickthrough rate (CTR), a key performance indicator of search success, online advertising effectiveness, and online behavior at large. The company made its initial foray into the search market by launching search suggestions, an important initial application of generative artificial intelligence models (Kucharavy et al., 2023; Park and Chiba, 2017; Serban et al., 2016). In line with industry standards, search suggestions are generated by the underlying algorithm that uses multiple sources to predict what users want to click in response to their queries. As a smaller player in the search market, the company had leveraged the market leader's autocomplete API to retrieve candidate suggestions to generate its search suggestions. We partnered with the search product team to experiment with access to the market leader's autocomplete API while using the same algorithm to generate search suggestions. This partnership provides an ideal context where we can measure the causal effects of access to the market leader's autocomplete API.

In this field experiment, more than 2.3 million users were randomly assigned to one of two conditions over 108 days (about 16 weeks). In the control condition (the status quo), the company's ranking algorithm ranks search-suggestion candidates retrieved from the market leader's autocomplete API along with its own search-suggestion candidates and generates a final list of search suggestions in response to user-submitted queries. These candidates retrieved from the market leader's autocomplete API is what we refer to as "the market leader's data capability". The queries to the market leader's autocomplete API are not personalized because the company does not provide user-specific personal data (e.g., search histories) in those API calls. In the treatment condition, we remove access to the API: that is, the company's ranking algorithm does not have access to the market leader's search-suggestion candidates and ranks only its own candidates. This research design has three notable features. First, the intervention exogenously changes the supply of candidates at the ranking stage of the search-suggestion generating process, while holding the ranking algorithm and user interface constant. Hence, the random assignment among millions of users enables us to precisely estimate the causal effects of access to the market leader's data

capability. Second, maintaining a consistent treatment over 108 days enables us to estimate the longer-term effects of access to the market leader's data capability. Third, this design is closely related to the broader agenda of the Digital Markets Act, which requires privacy-preserving data-sharing agreements touted to have significant benefits for companies in the marketplace.

We report three findings. First, on average, removing access to the market leader's autocomplete API leads to a 4.6% decline in CTR on search suggestions. We also document who is more (less) responsive to the intervention. For example, the negative impact of removing the market leader's autocomplete API is more pronounced for heavy users (v.s. light users), suggesting that access to the market leader's API is more effective in helping user retention by streamlining heavy users' experience than in ameliorating the cold-start problem for light users.

Second, leveraging the random assignment as an instrumental variable, we establish the link between clicking on search suggestions and clicking on the result at the top of the SERP (i.e., top-slot search result). Specifically, the elasticity of CTR on the top-slot search result with respect to CTR on search suggestions is 0.185. That is, a 10% increase in CTR on search suggestions leads to a 1.85% increase in CTR on the top-slot search result. These results demonstrate, on average, the downstream impact of the market leader's data capability on user engagement on the SERP.

Third, the magnitude of treatment effect estimates drops over the 16-week experimental period. It starts at about a 8.1%–9% decline in the first 3 weeks and becomes less negative at 3.6%–4.5% in the last 3–4 weeks. These estimates suggest that, had we run a short-term experiment, we could have overestimated the value of the market leader's data capability (the negative effect of API removal) by a factor of 2. To shed light on the underlying mechanism, we find that the magnitude of treatment effects is smaller as queries accumulated more internal data (searches) over time. That is, in the absence of the market leader's API, there appears to be a gradual improvement in the company's autocomplete search suggestions because of internal data accumulation. Over time, such an improvement gradually mitigates the negative effect of removing the market leader's API access. We further verify that this data pattern is not driven by (1) diminishing returns to external candidates, (2) user adaptation, (3) dynamic self-selection, or (4) unobserved spillover effects.

Our findings offer practical implications for managers and policymakers. First, we substantiate

a critical intertemporal trade-off in leveraging external data capabilities. On the one hand, over the entire experimental period, removing access to the market leader's API reduces CTR on both search suggestions and SERP.[3] On the other hand, exploring the dynamics in the treatment effects suggests that extensive reliance on the market leader's data capability tends to limit internal data accumulation over time, which could impede improvements in autocomplete predictions in the longer term. Thus, managers need to develop strategies that bridge the gap between quick wins and lasting success. For example, in the second field experiment, we show that lowering the rank of the market leader's candidate items (rather than removing the market leader's API access altogether) may enable the company to benefit from the market leader's data capability while potentially preserving internal data accumulation.

Second, our findings inform regulations such as the Digital Markets Act, which requires large digital platforms/gatekeepers to share depersonalized data with smaller players in search. Indeed, there are notable economic benefits of privacy-preserving data sharing for smaller players in the search market, especially in the initial stages of product development. Yet policymakers need to be mindful of smaller players' self-development in the longer term. In addition, access to gatekeepers' data capabilities is not a panacea for the lopsided market structure given the degree of market concentration in the search market (e.g., Google has more than 80% market share in the U.S.). Notably, recent work by Allcott et al. (2024) suggests that alternative policies targeted at switching defaults may have a more significant impact on market structure in the context of web search. Given that research in this area is in its infancy, policymakers could take a multi-faceted approach that systematically evaluates the role of defaults, data capabilities, interoperability among others.

Our paper contributes to four strands of academic literature. First, we contribute to the literature that aims to quantify the value (of different dimensions) of data for search engines and related online products. Yoganarasimhan (2020) analyzes how utilization of user-level data can help in the personalization of search and quantifies significant returns to personal data. Chiou and Tucker (2017) analyze the efficacy of search recommendations when companies such as Yahoo! and

[3]More broadly, our research substantiates the potential impact of restricting access to large platforms' API, as was the case with Google, Reddit, and Yahoo. For more information, see: https://developers.google.com/search/blog/2015/07/update-on-autocomplete-api; https://www.reddit.com/r/rootgame/comments/14jmfzx/reddit_is_killing_thirdparty_apps_and_itself/ and Havakhor et al. (2024).

Microsoft reduced the amount of individual-level data retention to 90 days and found no change in a user's CTR. Schaefer and Sapi (2023) analyze the role of within and across user learning for algorithms used by search engines to increase engagement. Klein et al. (2022) demonstrate the need for large players to share user-generated data to improve the performance of competing search products. Zhao et al. (2023) document that limiting the use of personal data, the implementation of General Data Protection Regulation increases consumers' efforts in general and product search. These papers are motivated by privacy regulations and the role of data in conferring market power to search engines. We augment this strand of research by analyzing how access to external data capabilities, a combination of data and inferences based on that data, impacts search product performance using a large-scale, longer-term field experiment. We demonstrate that depersonalized search-suggestion candidates via a leading search engine's API could help search products grow successfully, at least in the short run, but there could be a trade-off between the short-run benefit and the longer-term development of the focal product. Finally, we take a middle path and analyze a privacy-preserving situation where depersonalized data inputs from external sources could be used for product development. More generally, we speak to regulations associated with mandated data sharing across search engines in a privacy-preserving manner.

Second, we contribute to the literature that analyzes different strategies for platform growth. Benzell et al. (2023) use aggregate data on online platforms to demonstrate how a firm can grow by opening itself up to third-party complementors using APIs. Relatedly, Peukert et al. (2024) quantify the value of personal data (or lack thereof) for an algorithm relative to human experts. Sun et al. (2024) simulate a privacy regulation through a field experiment on Alibaba to quantify the value of individual-level data for platform users. They find significant negative effects on engagement and purchase when personal data is not used for product recommendations. Relative to these papers, and to Sun et al. (2024) in particular, we analyze a different yet economically meaningful context of search, which, as summarized in Online Appendix Table A1, is theoretically distinct from e-commerce in terms of their recommender systems and the potential use for external data capabilities. Moreover, whereas Sun et al. (2024) study the value of personal data in a business-to-consumer setting, we study the extent to which access to the market leader's data API influences

smaller players in search (i.e., a business-to-business data-sharing mechanism). Next, relative to extant studies with a short-term experiment, our 108-day experiment enables us to uncover a trade-off where external data capabilities can help product performance initially, but over-reliance in the longer term could inhibit focal prediction development based on internal data.

Third, we contribute to a strand of literature that analyzes the impact of utilizing several forms of third-party data on firm outcomes. Beraja et al. (2023) show that access to government data, in the form of surveillance videos, leads to AI innovation in the facial recognition industry by private companies in China. Similarly, Nagaraj (2022) finds that access to publicly available Landsat, a U.S. National Aeronautics and Space Administration satellite mapping program, leads to more gold discoveries especially by new entrants. In the case of online platforms, Chan et al. (2022) show that access to employer-verified employment data via Equifax benefits both auto loan borrowers and lenders. Havakhor et al. (2024) utilize the shutdown of Yahoo Finance API to show that access to high-volume historical and real-time price data causes retail investors to trade excessively with worse outcomes. Wernerfelt et al. (2024) leverage a field experiment to show that a loss of off-platform cookie data increases the customer acquisition cost for advertisers on Meta. Using our experimental setup of data sharing through the market leader's autocomplete API, we can cleanly identify dynamics in the treatment effects within a policy-relevant search context. More generally, we focus on a widely used, external source of data capabilities (i.e., API access), providing implications for other companies that aim to utilize such an external source for their product growth.

Finally, we contribute to a nascent strand of literature that focuses on the efficacy of text-based search on e-commerce platforms. Zheng et al. (2023) leverage a field experiment on a food delivery platform to understand the impact of related product suggestions in the search bar interface on the amount and diversity of consumption. Fang et al. (2024) analyze the impact of textual refinement and visual cues in the search bar on short- and long-term consumption patterns. Similar to these papers, our paper analyzes the impact of an intervention targeted at facilitating the search process, and our estimates are in the same ballpark as these papers. A key differentiator is that we examine the role of a business-to-business data sharing mechanism via access to the leading search engine's API, a lever of strategic and policy relevance (Benzell et al., 2023; Li and Kettinger, 2021).

2 Empirical Setting

In this section, we describe the empirical setting with a focus on (1) search-suggestion generating process through the lens of the algorithmic funnel, (2) the role of the market leader's autocomplete API in shaping such a process, and (3) our choice of target metrics that measure the performance of search suggestions.

We partnered with a large technology company in China that develops a mobile app for users to gather, consume, and share information. The app offers various in-app products such as news feed, search engine, and video and eBook streaming, each with its dedicated product interface. We focus on search suggestions, the main search product with approximately 100 million monthly active users (100% mobile) as of September 2021. By design, search suggestions function as query autocomplete, which is similar to the autocomplete feature offered by Google Chrome. Search suggestions are considered as an early-stage application of generative artificial intelligence models. Generative models produce system responses that are autonomously generated word-by-word, which open up the possibility for realistic, flexible interactions (Serban et al., 2016). Figure 1 provides an illustration of the app interface. When users enter a specific query term into the search bar (e.g., covid), they typically see a list of ten search suggestions in the form of phrases and/or sentences in response to the query (e.g., "What does a covid-19 Ag mean?"). A user can choose to click a matched suggestion, press the search button without adopting search suggestions, delete this query and enter a new one, or quit the session.

Companies invest significant resources in search suggestions and related contexts of predictive text completion (Agrawal et al., 2018). Search suggestions streamline the search process, bridging the gap between users' search intent and content consumption. Optimizing search suggestions is important because our partner company intends to monetize this process. For example, search suggestions may accelerate the process of finding desired results at the top of the search engine results page (SERP), and clicks on such search results generate substantial revenue (Schaefer and Sapi, 2023; Ursu, 2018). Moreover, the search team's longer-term objective is to monetize search suggestions with built-in brand logos and websites (e.g., JD.com), and clicks on these search sugges-

tions would guide the search process to the SERP or could even take users straight to the websites without landing on SERP.[4] Indeed, search box optimization is an increasingly popular strategy for additional visibility early in the search process (Zaif, 2023).

Figure 1: An Illustration of Search Suggestions

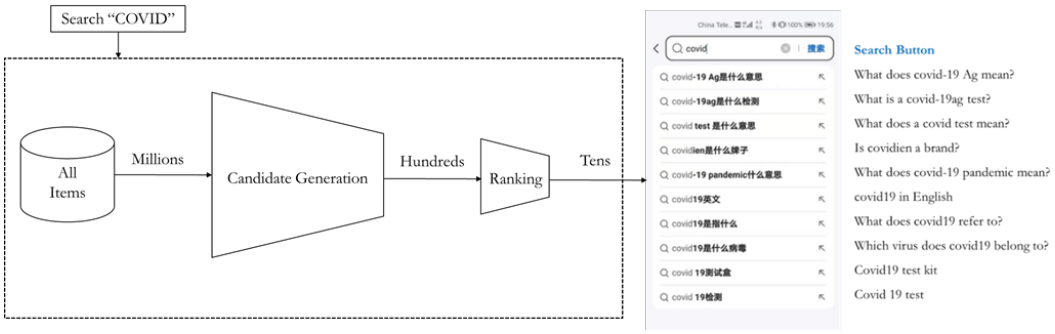


How does the company generate search suggestions in response to user-submitted queries? At a high level, the company uses a proprietary algorithmic funnel, a three-stage architecture similar to a recommender system (Covington et al., 2016): (1) item generation, (2) candidate generation, and (3) ranking. Figure 2 provides an illustration. At the *item generation* stage, the company first builds raw items based on (a) internal sources in the news feed (e.g., articles and videos) and query terms from users' active search histories and (b) external sources such as public trending news. Next, the company uses natural language processing techniques (e.g., keyword extraction, text summarization) based on large language models to transform raw items into phrases and sentences. As a result, this large item base contains millions of phrases and sentences that are filtered at the

[4]While our experiment was running, the search team started deliberating about the concept of direct reach. Direct reach products would appear as brand logos or information guiding the search process to the SERP and could even lead the user straight to the websites without landing on SERP. More broadly, as of 2024, direct reach products are prevalent in China among general search engines (e.g., Baidu, Sogou) and specialized search engines on social media platforms (e.g., ByteDance, Xiaohongshu). Notably, unlike sponsored ads on the SERP, direct reach products are not associated with disclosure labels. Moreover, public documents suggest that search engines enter into bilateral contracts with companies for direct reach products in search suggestions, where rates are determined based on traffic thresholds. This process is different from the real-time auctions for sponsored search results.

next stage. At the *candidate generation* stage, a subset of candidate items relevant to the submitted query are retrieved from the item base based on their popularity on the platform (e.g., common and trending searches). As a standard practice in such a context, this stage does not utilize personal data because of the need to filter through millions of data points (Mitra and Craswell, 2015). As a result, hundreds of candidate phrases and sentences are selected to enter the ranking stage. At the *ranking* stage, there are several steps of retrieving a larger number of features and the use of a pre-trained algorithm. Eventually, a ranking algorithm scores each candidate item according to its predicted click-through rate, which is a function of user (e.g., location, search histories) and query (e.g., topic, freshness) features. The highest-scoring items are presented in a ranked order in the final list to users. The literature shows that sophisticated algorithms are utilized to handle a larger feature set at the ranking stage and rank fewer items. This contrasts with the candidate generation stage that uses generic models and considers a larger set of items focusing on efficiently pruning duplicate and irrelevant items (Covington et al., 2016; Nandy et al., 2021).

Figure 2: An Overview of the Algorithmic Funnel



However, a key challenge of developing a new search product, such as search suggestions, is the potential lack of candidate items with respect to quantity and quality. For example, at the initial development stage, the company's algorithmic funnel may generate a relatively shorter list of search suggestions that could match users' search intent. To address this problem, the company had started to leverage access to the autocomplete API of a leading search engine in China (i.e., the market leader's API). The company's hypothesis is that because of its large user base and capability

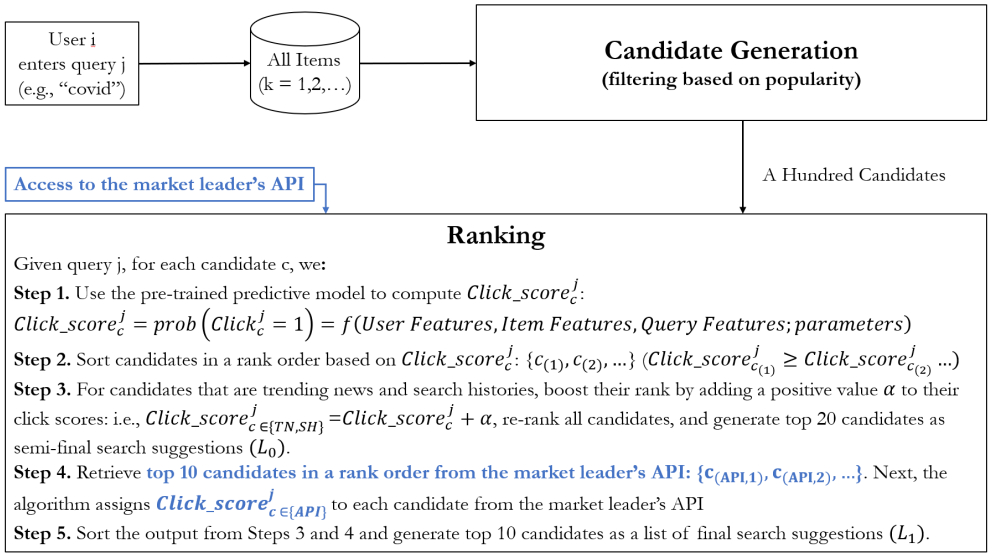
to present relevant, timely results to of millions of users at scale, the market leader may (1) provide the company with access to more precise candidate items with respect to their popularity and freshness, and (2) expand the scope of the company's candidate items with respect to their topic coverage, both of which can be used in the ranking stage to generate a list of search suggestions.

How does the market leader's API work? At a high level, the company and the market leader form a business-to-business contractual relationship where the company acquires a license to the market leader's API. To initiate the data request, the company provides the market leader's API with its user-submitted queries in real time, which are used by the market leader's API to return a set of candidate items. Importantly, to preserve privacy, the focal company does not provide any personal information (unless explicitly volunteered by the user through their query) such as search histories. Put it differently, the candidate items received from the market leader's API represent predictions of aggregate, trending searches based on the market leader's data and the inferences based on its candidate generation algorithm. It is pertinent to note that neither we as researchers nor the company observe any details of how the market leader generates its candidate items or updates its candidate generation algorithm (i.e., a black box). Hence, we cannot separate the impact of data from that of the inferences and refer these candidate items as data capabilities, a concept highlighted in the literature and policy discussions (Ubaldi, 2013; Zeleti and Ojo, 2017). And then, at the ranking stage, the algorithm uses (1) candidates items from the market leader's API as an additional input along with (2) its own candidate items to generate a list of search suggestions in a ranked order. The company and market leader have an agreement where the company pays service fees to the market leader at an undisclosed rate for each query it makes through the API (see Figure A1 in the Online Appendix).

Concretely, building on Figure 2 and the discussion above, Figure 3 decomposes the entire process where the company's ranking algorithm meets the market leader's API. To fix ideas, consider the user i who enters the search query j into the search box (e.g, covid). First, the company's candidate generation model filters through millions of items and retrieves about 100 relevant candidate items from its item base (items indexed by k=1,2,3,...,K), associated with query j based on their popularity. Second, the ranking algorithm scores each candidate item based on user (e.g., location,

search histories), query (e.g., topic, freshness), item features, and interactions among such features and sorts them according to their predicted click score $Click_Score_c^j$ (i.e., Steps 1–3 under **Ranking**). Third, given query j, another set of depersonalized candidate items is retrieved from the market leader's API in a ranked order ($c_{API(1)}$, $c_{API(2)}$...). Fourth, for each candidate item from the market leader's API, the ranking algorithm assigns a score $Click_Score_{c,API}^j$ (i.e., Step 4 under **Ranking**). Lastly, the ranking algorithm re-ranks the mix of candidate items from two sources to generate a list of highest-scoring items as search suggestions (i.e., Step 5 under **Ranking**). Hence, access to the market leader's API changes the supply of candidate items entering into the ranking stage. The overall architecture used by the company mirrors existing industry standards.

Figure 3: Search-Suggestion Generating Process: Decomposing the Algorithmic Funnel



Notes: The rank order was determined based on the click score, a continuous measure based on a variety of input features, so it is unlikely that there were ties between two search suggestions with respect to their click scores. During our experiment, alpha was determined based on the team's problem-solving heuristics and expertise to boost the rank of trending news and search histories.

Notably, this setup reflects a specific form of data sharing agreements between companies and provides a context to analyze the economic value of data sharing. First, at the time of the field experiment, the market leader's API was indeed available to other entities in the market. Second,

candidate items retrieved from the market leader's API mirror how Google custom search API provides developers with response data that can be incorporated (Alrashed et al., 2020; Zaveri et al., 2017). Third, this context provides an example of how gatekeepers could provide smaller players or startups with "access on fair, reasonable and non-discriminatory terms to ranking, query, click and view data," if mandated within the framework of the Digital Markets Act. The details of the Digital Markets Act and similar regulations are still being finalized, so our analysis can be viewed as an early look at the practicalities and potential benefits of data sharing by gatekeepers.

How do we evaluate the performance of search suggestions? Our conversations with the search product team revealed that the target metric is click-through rate (CTR), a widely-used key performance indicator of search success in online markets. CTR is important from the company's perspective because customer satisfaction with the search product depends on the search engine's ability to serve relevant results (Yoganarasimhan, 2020). In the context of search suggestions, CTR is measured as the ratio of number of clicks to search suggestions in a list to the number of exposures. Specifically, when a user starts typing a keyword into the search bar, this user is exposed to a list of search suggestions in this session (i.e., one exposure). If a user clicks to any search suggestion in this list, such an action will be counted as one click. To ensure that our results are not sensitive to the variation in exposures, we conduct robustness checks using alternative measures such as the number of clicks and the probability of any click. We can measure CTR for each user on a daily basis or over a longer period (e.g., week). To capture the overall user activities, rather than a user's specific search session, we focus on the *aggregate* CTR for each user: the ratio of the total number of clicks to total number of exposures over the entire experiment (Yang et al., 2024). Moreover, since clicks on the SERP represent a key metric of search quality (Schaefer and Sapi, 2023), we also examine CTR on search results as a downstream outcome of interest.

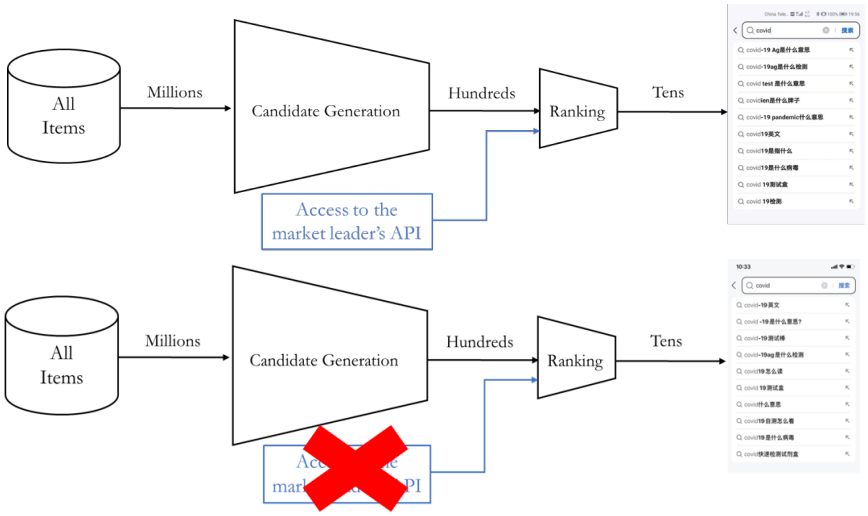
However, measuring the causal effects of the market leader's data capability (candidate items retrieved from the market leader's autocomplete API) on target metrics is challenging without exogenous variation on this dimension. To circumvent this challenge, we conducted a large-scale field experiment where we manipulated access to the leading search engine's autocomplete API while holding all other aspects (e.g., algorithms, user interface) equal.

3 Field Experiment

3.1 Experimental Design

The randomization was implemented at the user level in real time. That is, as soon as a user started typing keywords into the search bar in a given time point during the experiment, she was randomly assigned to one of two conditions.[5] Once a user was assigned to a condition, this user stayed in the same condition until the experiment ended. We maintained a consistent treatment assignment over 108 days from May 17, 2021 to September 1, 2021. Figure 4 visualizes our design.

Figure 4: Experimental Design



Control (N=1,194,619): In response to user-submitted queries, search suggestions are generated by the company's proprietary algorithm funnel, including item generation, candidate generation, and ranking. In the control condition, at the ranking stage, the ranking algorithm scores candidate items from two sources: including (1) those retrieved at the candidate generation stage and (2)

[5]The company ensured that users in this experiment did not overlap with the users in any other experiment conducted simultaneously on the platform. Like other major technology companies such as LinkedIn, Google, and Microsoft, the company uses the standard design of overlapping experiments to run experiments simultaneously and efficiently (Figure 2b in Tang et al. (2010)) and hash-based assignment (rather than the use of ephemeral random numbers) to ensure no correlation between the assignments with multiple experiments running (Xu et al., 2015).

those supplied by the market leader's API. Next, the ranking algorithm generates the highest scoring items in a ranked list. As a result, users in this condition see a sizable proportion of search suggestions supplied by the market leader even though the number of search suggestions from the market leader might differ across users depending on queries, demographics, and search histories.

Treatment (N=1,195,625): In response to user-submitted queries, search suggestions are generated by the same proprietary algorithm funnel without access to the market leader's API at the ranking stage. The ranking algorithm (the same as the one used in the control condition) scores only candidate items retrieved from the candidate generation stage and generates the highest scoring items in a ranked list (i.e., no Step 4 in Figure 3). As a result, users in this condition never see search suggestions supplied by the market leader.[6] Notably, our treatment does not cause any change in the user interface: that is, there is no disclosure to the user of whether a search suggestion comes from the market leader's autocomplete API.

3.2 Data

The primary data set for analyses is at the user-level (the unit of randomization) where we observe a unique user identifier, treatment status, number of exposures to search suggestions, number of clicks to search suggestions, and search button usage. In addition, we observe pre-experimental characteristics, such as demographics (e.g., gender, city of residence), mobile operating system (e.g., Android), and activity level (e.g., active days in the past 30 days). About 82% of users have used the search bar prior to the experiment in this data set. The dependent variable, aggregate CTR, is computed for each user as the ratio of the total number of clicks to total number of exposures over 108 days. We conduct several checks to ensure that the random assignment is successful. Table 1 shows that the mean difference in observables across conditions is neither economically nor statistically significant. Figures A2–A4 in the Online Appendix show that users were equally likely to be assigned to either condition over the course of our experiment, regardless of whether they are new users on any given day (e.g., if a user had not used the search bar since the beginning of the

[6] It is pertinent to note that any updates to the training data input during the experimental period remains separate for treatment and control conditions. Similar to data-diverted experiments proposed in Holtz et al. (2023), this was explicitly done to prevent Stable Unit Treatment Value Assumption (SUTVA) violations.

experiment and did so for the first time on day t, she is considered a new user on day t).

Table 1: Randomization Checks

User Characteristics	Control	Treatment	p value
Male	0.5034	0.5037	0.6724
	(0.0005)	(0.0005)	
Larger Cities	0.5053	0.5048	0.4254
	(0.0005)	(0.0005)	
Smaller Cities	0.4340	0.4346	0.3485
	(0.0005)	(0.0005)	
Mobile Operating System: Apple iOS	0.1067	0.1066	0.7358
	(0.0003)	(0.0003)	
Mobile Operating System: Android	0.8368	0.8369	0.7944
	(0.0003)	(0.0003)	
Active days in the past 30 days (search activities)	100	99.8259	0.4257
	(0.1547)	(0.1544)	
Query views in the past 30 days (search activities)	100	99.3835	0.1263
	(0.2878)	(0.2825)	

Notes: This table shows the balance along several observable dimensions between users in the treatment condition and those in the control condition. The second column and third column provide the mean of each variable with the standard error in parentheses. Following the hierarchical classification of Chinese cities, larger cities include tier 1 to 4 cities (e.g., tier 1: largest cities such as Beijing), whereas smaller cities refer to tier 5 cities and below. p-value is obtained based on a two-sided t-test on the equality of means with unequal variances. For confidentiality purposes, values reported in the last two rows were normalized so that the variable means in the control condition are 100.

We further supplement this user-level data set (**Primary Data**) with two additional data sets. First, we collect user-level metrics such as the number of searches conducted on the search engine results page (SERP), the number of clicks on the top-slot search results, and the number of clicks on the other-slot search results (**SERP Data**). Second, we collect detailed log records of a random sample of our experimental subjects (11,630 treated users and 11,913 control users) starting from June 26, 2021 (**Granular Data**). We were not able to collect log-level data for the entire experiment sample nor data for the random sample prior to June 26, 2021, due to the company's data retention policy. In each log record in **Granular Data**, we observe the query term entered by each user, content category associated with the term, the ordered list of search suggestions provided, and whether and what a user clicked. On average, there are 8.80 and 8.68 search suggestions presented to users in the control and treatment conditions, respectively.

4 Empirical Analyses

4.1 Empirical Framework

We use a potential outcome framework to specify our model. For illustration purposes, we consider an experiment with N users who are randomly assigned to one of two conditions: that is, treatment (e.g., API removal) or control condition. For a set of independent and identically distributed users $i = 1,...,n$, we observe the outcome of interest Y_i (e.g., CTR aggregated over 108 days); treatment assignment T_i; and a vector of user characteristics Z_i (e.g., demographics, past search activities). For each user i, there are two potential outcomes: if a user is assigned to the treatment condition, we observe the outcome $Y_i = Y_{i1}$, and if the user is assigned to the control condition, we observe $Y_i = Y_{i0}$. In theory, the average treatment effect (ATE) is $E[Y_{i1} - Y_{i0}]$, can be used to assess whether the treatment causes changes in Y_i. Alternatively, we can estimate the following regression:

$$Y_i = \alpha + \beta \times T_i + \epsilon_i, \tag{1}$$

where β captures the causal effect of the removal of the market leader's API on the outcome of interest (e.g., CTR aggregated over 108 days). Because of the successful randomization, we do not expect the controls Z_i to affect the estimate of β. Therefore, we estimate the regression without control variables as the baseline results and use heteroskedasticity-robust standard errors.

Another important consideration is that β itself does not provide a sense of the effect magnitude. Hence, we report the *lift* estimates to facilitate the interpretation of estimates as the magnitude and comparison across experiments (Goli et al., 2024; Gordon et al., 2023): the incremental CTR among treated users relative to control users as a percentage of CTR among control users ($\frac{\overline{Y_1} - \overline{Y_0}}{\overline{Y_0}}$ or $\frac{\hat{\beta}}{\hat{\alpha}}$). A negative (positive) value of the estimated lift indicates a decrease (an increase) in CTR among treated users relative to control users. Importantly, the lift is a ratio of two random variables, making it a random variable. Therefore, we use the Delta method to derive approximations for the mean and variance estimates of the lift (Casella and Berger, 2002; Deng et al., 2018).

4.2 Baseline Effects

4.2.1 Average Treatment Effect and Treatment Effects by User Characteristics

We start with estimating the average treatment effect. As shown in Figure 5, there is a statistically significant and negative treatment effect over 108 days of the experiment: on average, API removal decreases the search-suggestion CTR by 4.6% in the treatment condition relative to the control condition (i.e., the average lift is –4.6%).[7] The large sample size gives us a precise estimate as the width of the 95% confidence interval is less than 10% of the absolute value of the point estimate. Notably, the magnitude of the effect is economically significant in the context of digital experimentation in search. Relative to other academic studies, the magnitude is in line with the 1–2% effect in Zheng et al. (2023) for the impact of related product suggestions in the search bar interface on relevant metrics for an online food delivery platform. Similarly, Fang et al. (2024) find that textual and visual refinement of text search leads to a 1.3% increase in purchases over 24 weeks on an e-commerce platform, while Yang et al. (2024) find that incorporating advertising information in ranking search listings leads to an additional 0.15%–0.57% increase in key metrics of interest.[8]

Next, we explore whether treatment effects vary with pre-experimental user characteristics to dig into who is more (less) responsive to the treatment of API removal. The first dimension relates to gender and city of residence as possible sources of heterogeneity. We see that (1) treatment effects are economically and statistically indistinguishable for female and male users, and (2) the treatment effect is significantly larger among users in larger cities than those in small cities. From a managerial perspective, these findings inform the company's growth and segmentation strategy for a nascent online product based on readily available demographic variables.

The second dimension is to examine whether users with different activity levels are more or less responsive to API removal. A priori, it is unclear whether external candidate items will help

[7] In Online Appendix D, we have presented our results in the form of tables, each of which is a companion to the corresponding figure (e.g., Table D1 corresponds to Figure 5, Table D2 corresponds to Figure 6, etc.).

[8] As a relevant benchmark, 70%–80% of experiments in digital markets yield statistically and economically insignificant results (Kohavi et al., 2013). See comments by Ronny Kohavi, former VP of Analytics and Experimentation at Bing, here https://www.facebook.com/watch/?v=2368597899925946. It is also important to note that there are some dimensions, such as benefits of saving time, which were not capture through our analysis. Aggregated over a significant user base and time, the cost savings of such a streamlined search could be substantial.

solve the cold-start problem for light or provide a more streamlined experience for heavy, or both. To examine this, we use the number of search-related active days in the past month prior to the experiment to differentiate light users from heavy users. We construct an indicator variable, heavy user, which is equal to 1 when a user's number of search-relate active days is strictly above the median and zero otherwise. Figure 5 shows that the magnitude of the negative effect of API removal among heavy users is significantly larger than that among light users. This pattern is highly consistent when we use the number of query views in the past month prior to the experiment as an alternative. A plausible explanation is that because heavy users are more experienced, they may set a higher expectation of product quality and are more sensitive to changes in the quality of search suggestions due to the absence of the market leader's data capability.[9] In this regard, our results complement Sun et al. (2024) where they find that the use of personal data in the recommendation benefits light users more when data volume and customer resilience coexist in their context.

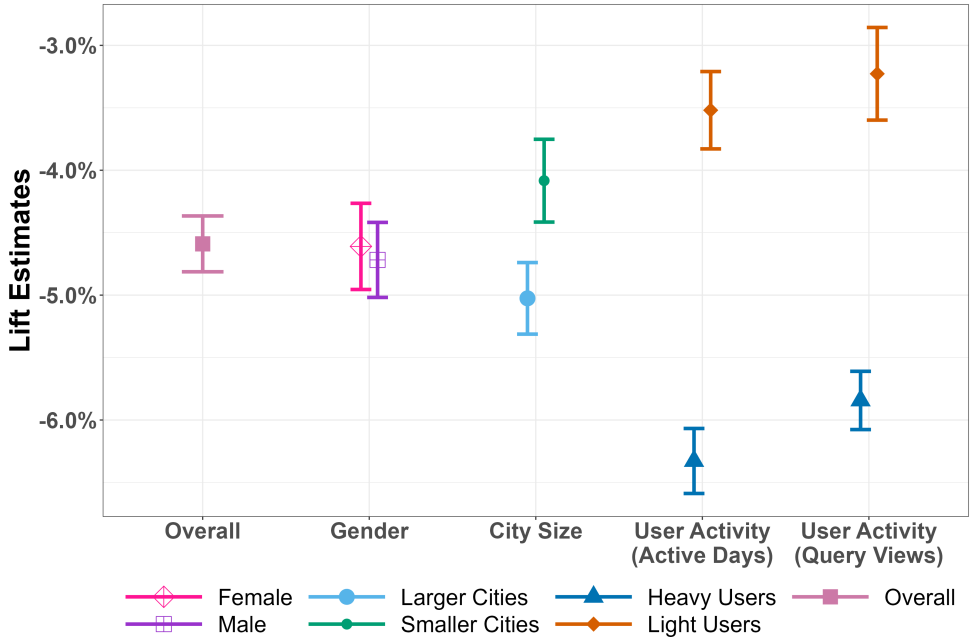
And then, we look at another target metric, search button usage, to explore whether a decrease in CTR on search suggestions is substituted by the utilization of an alternative search function. In particular, if the search suggestions become less useful for users, they might use the search button as an alternative way to navigate to the SERP. On average, we find that API removal significantly increases the usage of the search button by 5.7%. This positive effect on search button usage suggests a positive demand spillover to the search button as a substitute for search suggestions. The magnitude of the change suggests an almost one-to-one substitution between these actions.

Lastly, we carry out a variety of checks to test for the robustness of our baseline estimates (Table A3, Online Appendix). First, we estimate the treatment effect using only the first-day observation of each user in the experiment (i.e., the first day of the experiment when a user interacts with search suggestions). Second, we estimate a linear regression adjusting for observed user characteristics to potentially improve the precision of the estimate, and check the stability of results among the sample where all user characteristics are observed. Columns (1) and (2) show that the results are qualitatively similar to our baseline estimates. Third, our results are robust to alternative

[9] In Online Appendix B, we further explore how treatment effects vary by the popularity of a content category and find that the negative treatment effect is driven by queries in popular categories. It suggests that due to the size of its user base, the market leader's autocomplete API provides the company with access to useful candidate items with respect to their popularity based on real searches on the market leader's platform.

operationalizations of the dependent variable, including the click dummy (Column (3)) and the logarithm of one plus the number of clicks (Column (4)). Lastly, we estimate a linear regression with moderators rather than subsample analysis and find consistent results (Column (5)). In summary, these checks provide an additional degree of confidence in our baseline results.

Figure 5: Average Treatment Effect and Heterogeneous Treatment Effects by User Characteristics



Notes: Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. CTR is computed as the ratio of the total number of clicks to total number of exposures over the entire experiment (108 days). The regression is specified in Equation 1. Overall refers to the average treatment effect based on the full sample (N=2,390,244), while heterogeneous treatment effects are based on subsamples by user characteristics (gender, city size, user activity). Error bars represent 95% confidence intervals of lift estimates, which are calculated using the Delta method.

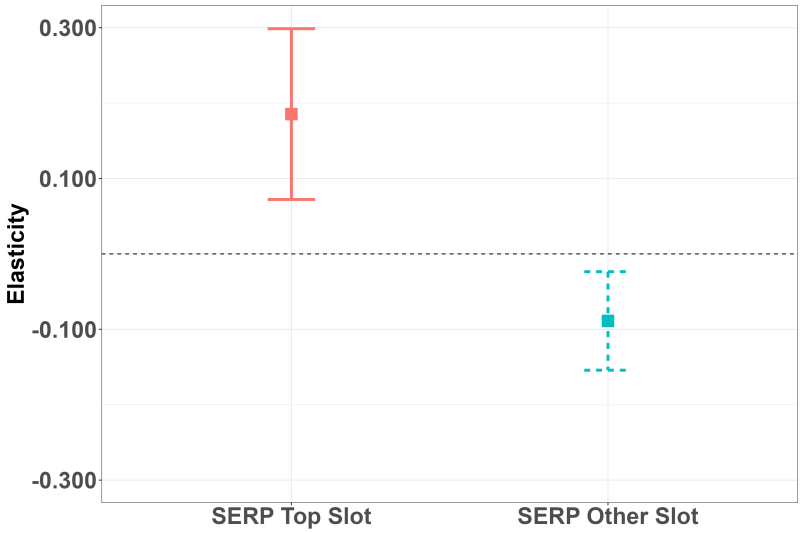
4.2.2 Downstream Impact on the Search Engine Results Page (SERP)

To gauge the downstream impact, we establish an empirical link between clicking on search suggestions and clicking on results on the search engine results page (SERP). That is, we translate the decrease in clicking on search suggestions into a decline in clicking on search results. Specifically, we focus on top-slot search results because they capture customer satisfaction with the search product

and the relevance of top-slot clicks is well established in the literature in terms of revenue implications for the company (e.g., Schaefer and Sapi (2023); Ursu (2018); Yoganarasimhan (2020)). Using **SERP Data**, we operationalize the CTR on search results using (a) CTR on top-slot search result and (b) CTR on other-slot search results.

API removal exogenously shifts CTR on search suggestions (the first-stage F-statistic is 506.85; Table D2, Online Appendix), and it does not belong directly in the production function of CTR on search results. Thus, using API removal as the instrumental variable (IV) for CTR on search suggestions, we can estimate the causal relationship between CTR on search suggestions and CTR on the top-slot search result via two-stage least-squares. We log transformed all variables to enable the interpretation of coefficient as elasticity. Figure 6 presents the estimated elasticity.

Figure 6: Elasticity of CTR on Search Results with respect to CTR on Search Suggestions



Notes: N=1,653,659. Error bars represent 95% confidence intervals of lift estimates, which are calculated using the Delta method. For estimation details, see Table D2.

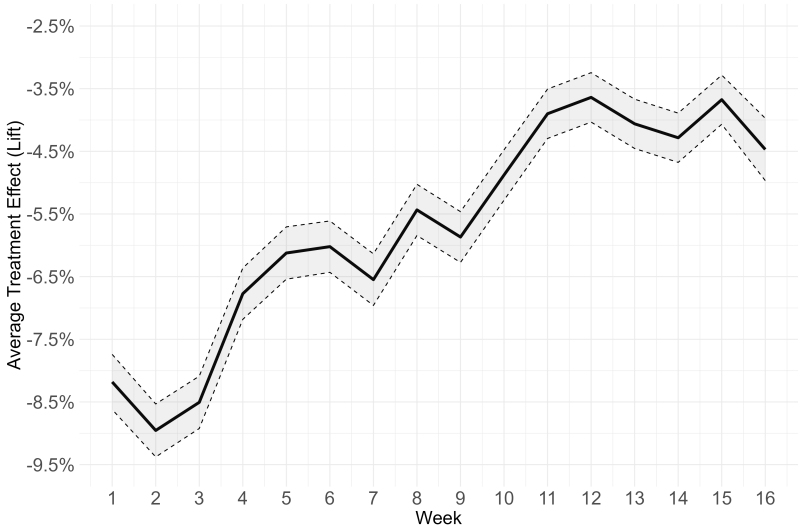
Specifically, the elasticity of CTR on the top-slot search result with respect to CTR on search suggestions is 0.185. That is, a 10% increase (decrease) in CTR on search suggestions leads to a 1.85% increase (decrease) in the CTR on the top-slot search result. This implies that clicking on search suggestions has a downstream impact on clicking on top-slot search results. In Online

Appendix C, we conduct a back-of-the-envelope calculation to shed light on the economic implications of access (or lack thereof) to external data capabilities. Next, if the CTR on top-slot search results increases, is it a market expansion effect or a substitution effect from other slots? Figure 6 shows a substitution effect. This implies that search suggestions streamline the search process by facilitating relevant top-slot search results, reducing attention on the lower slots.

4.3 Long(er)-Term Effects

A novel aspect of our design is that a consistent 16-week treatment assignment allows us to estimate the longer-term effects. Following Goli et al. (2018), we estimate a regression for each week as if that week were the final one. This gives us 16 separate regression estimates. Figure 7 shows how the lift estimates vary over time, with the solid (dotted) line representing the point estimates (the 95% confidence intervals). The absolute values of the estimates start at about a 8.1%–9%, with those decreasing to 3.6%–4.5% in the last few weeks. Thus, had we run a short-term experiment, we could have overestimated the value of the market leader's data capability by a factor of 2.

Figure 7: Longer-Term Effects



Notes: Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. The figure plots weekly lift estimates based on linear regression for the entire sample until the end of that week. Error bands represent 95% confidence intervals of lift estimates, which are calculated using the Delta method. For details, see Tables D3 and D4.

4.3.1 Do Our Results Reflect Diminishing Returns to External Candidate Items?

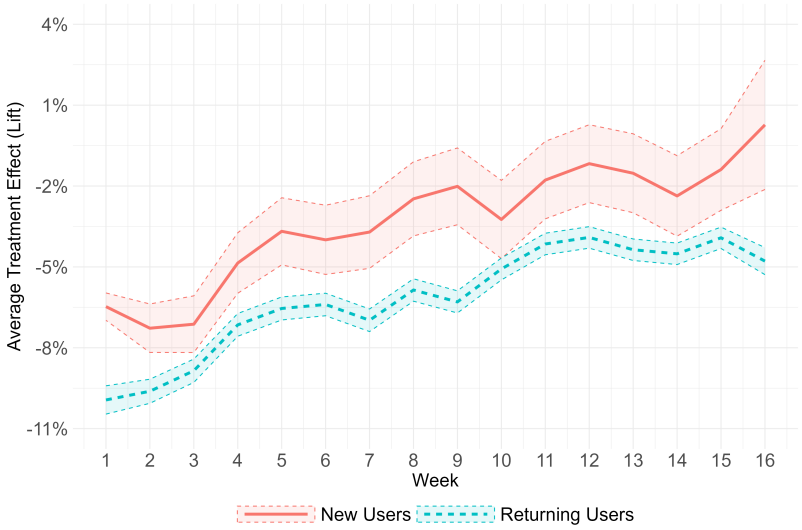
A possible explanation is that the decline in the magnitude over time could reflect diminishing returns to external candidate items (Peukert et al., 2024). That is, treatment effects could be less negative over time if the API-generated candidate items become less valuable over time. To examine this possibility, we focus on the trend in CTR among the users in the control condition where the candidate items are supplied by both the focal firm and the market leader. If the observed longer-term effect were to be driven by diminishing returns to external candidate items, the trend in CTR in the control condition would show a concave pattern. In Figure A6 in the Online Appendix, we conduct a calibration analysis to test the curvature of the weekly average CTRs among the users in the control condition (Committee, 1994; Johansson, 1979; Little, 1979). This calibration exercise cannot reject the null of a linear trend; hence, these results do not support a concave trend indicating diminishing returns to candidates items from the external API.

4.3.2 Do Our Results Reflect User Adaptation?

Another possible explanation is that users have a strong initial negative response to the changes in quality of search suggestions due to API removal and then gradually get used to such changes over time—user adaptation. This adaptation might explain the upward trend in the treatment effect estimates. However, this is less plausible in our setting because our treatment does not involve the disclosure of API items to the user. Moreover, we plot treatment effect estimates over time across two subgroups: new users vs. returning (experienced) users. We define new users on a weekly basis: During the 16-week experimental period, new users at week t are those who have never used the search bar since the start of the experiment and used the search bar for the first time at week t. Our hypothesis is that new users had few interactions with the product, so adaptation should be minimal for new users. Therefore, examining treatment effects over time among new users should help us understand if our effects are driven by user adaptation over time. Figure 8 shows that the treatment effects become less negative over the course of the experiment, even among new users

(solid line). Thus, our estimates are unlikely to be driven by user adaptation to API removal.[10]

Figure 8: Longer-Term Treatment Effects on CTR by User Type: New vs. Returning Users



Notes: Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. The figure plots weekly lift estimates based on linear regressions for sub-samples of new and returning users. Error bands represent 95% confidence intervals of lift estimates, which are calculated using the Delta method. For details, see Tables D3 and D4.

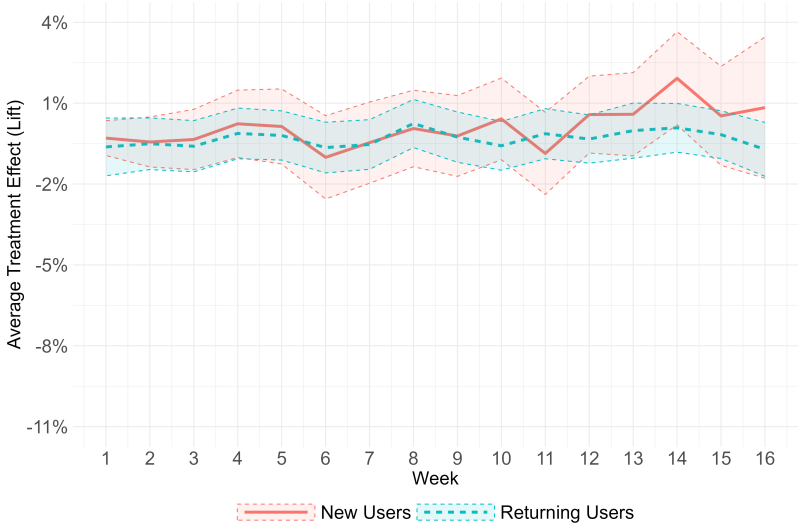
4.3.3 Do Our Results Reflect Dynamic Self-selection?

The third possible explanation is that our results are driven by dynamic self-selection of users. For example, our definition of new users could make the sample of new users became smaller over time, causing the temporal changes in the sample of new users. Similarly, if returning users were becoming inactive regarding their usage, the estimates would capture the treatment effect due to temporal changes in the sample of returning users. First, as shown in Figure A3 in the Online Appendix, our randomization ensures that the proportion of new users assigned to each condition is balanced over time. Second, we examine whether the treatment induces a significant change in query volume. We plot the weekly estimates of the impact of API removal on query volume among new and returning users. Figure 9 shows that API removal does not cause a significant change

[10] We find a similar trend for returning (experienced) users, but this trend could reflect a combination of user adaptation and internal data accumulation. We examine the latter in Section 4.3.5.

in query volume among treated users relative to control users in any week for both subgroups (confidence intervals consistently contain zero). Together, these results suggest that the decrease in the magnitude of lift estimates is unlikely to be driven by dynamic self-selection of users.[11]

Figure 9: Longer-Term Treatment Effects on Query Volume: New vs. Returning Users



Notes: Lift refers to the incremental number of queries entered by treated users relative to control users as a percentage of the number of queries by control users. Error bands represent 95% confidence intervals of lift estimates, which are calculated using the Delta method. For details, see Tables D5 and D6.

Figures 8 and 9 together suggest an interesting takeaway: whereas the market leader's API generates initial efficiency advantage since it effectively helps users reach the website they are looking for, it does not seem to reduce the number of times users search. That is, our results suggest that leveraging external data capabilities impacts the intensive margin behavior (clicks conditional on search) but not the extensive margin (the search volume).

4.3.4 Do Our Results Reflect Violations of SUTVA?

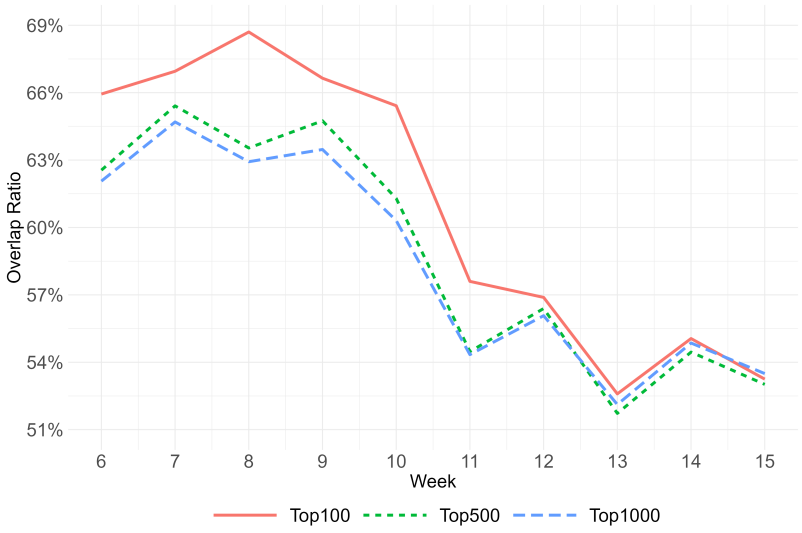
The fourth possible explanation is that the decline in the magnitude of the longer-term effects could be driven by a spillover across the treatment and control conditions. Over time, user-item-algorithm

[11] We further alleviate the concern of changes in the sample composition in the Online Appendix. First, Figure A4 shows the number of unique daily active users across two conditions track each other over time. Second, the average effects on query volume are not significant for returning users (p=0.36) and new users (p=0.62). Third, Figure A5 verifies that over time, API removal does not cause a significant change in the query volume for the entire sample.

interactions in the treatment (control) condition could affect the algorithm and subsequent search suggestions to the counterparts. As a result, this spillover might make search suggestions more similar across two conditions, gradually reducing the performance gap between the two conditions.

To examine this explanation, we provide institutional and empirical evidence. First, the algorithm was held constant across two conditions, and the system in each condition was updated based on only data produced by users in that condition. Similar to the purpose of "data-diverted experiments" proposed in Holtz et al. (2023), the goal of these system decisions was to prevent SUTVA violations. Second, we use **Granular Data** (detailed log records of 11,630 treated users and 11,913 control users) to examine the overlap between search suggestions in the two conditions. If there were a spillover, the overlap would have increased for queries entered by both users in the treatment condition and those in the control contion (i.e., common query). For example, if there were a spillover, when users entered the common query "coffee", the list of search suggestions in the treatment condition would be more similar to that in the control condition over time.

Figure 10: Overlap Ratio of Search Suggestions in Response to Common Queries



In Figure 10, we observe that over time, the overlap ratio of search suggestions in the two conditions decreases for the top 100, top 500, and top 1000 common queries, suggesting that the decrease in the magnitude of lift estimates is unlikely to be driven by the spillover between the

treatment and control conditions. This exercise also lends credence to the steps taken by the search team to keep data systems separate for the two conditions. Moreover, this diminishing overlap seems to indicate that the system could provide better search suggestions over time to close the performance gap across the two conditions, potentially due to the accumulation of internal data—the mechanism we explore next.

4.3.5 Do Our Results Imply Better Prediction due to Internal Data Accumulation?

So far, we have provided evidence to rule out several alternative explanations for the observed longer-term effects. Our thesis is that the decline in the magnitude over time could be driven by the gradual improvement in the company's search suggestions fueled by the accumulation of internal data in the absence of the market leader's data capability. Specifically, in the absence of the market leader's candidate items, the company's search suggestions could be improved because of additional internal data generated through user activities related to internal candidate items. Therefore, in the absence of access to the market leader's API, the relative improvement in the company's autocomplete predictions, based on its internal data, may gradually compensate for the negative impact of API removal. If this were the mechanism, it could significantly impact firm strategy in leveraging external data capabilities. In particular, it might suggest a fundamental trade-off between short-run benefits and longer-run development of its internal data capabilities.

Yet can the company use external candidate items to develop its internal data capabilities? In line with industry standards, although the focal company's algorithm has pre trained, offline pipelines to transform internal candidate items into features, it only has access to the output from the market leader's API and thus does not have pre-computing features for API items to be used for training the algorithm in real-time. As a result, this causes an inference and data-sparsity problem for external API-based output (Brinkmann, 2022; Sarwar, 2001; Stoica et al., 2017). Moreover, it is unclear whether such API-related data-sharing agreements allow for using the API output for training the focal algorithm. Indeed, a recent example is OpenAI API usage agreement, which "prohibits using output from the API to develop a competing product" and "prohibits reverse engineering the source code, model parameters and algorithm" (OpenAI, 2024). Similar concerns

of reverse engineering and the use of information to build competing products was one of the reasons Google restricted access to its autocomplete API.

Specifically, we examine whether the negative effect of API removal could be weakened if the queries accumulate more searches. The rationale is that if there were indeed improved predictions due to the accumulation of internal data in the treatment condition in the absence of the market leader's API, the negative impact of API removal would be mitigated for queries that accumulated more internal data (e.g., search histories). To dive deeper into this mechanism, we constructed the data set at the user-query-day level based on the **Granular Data**. This data structure allows us to include query-fixed effects and day-fixed effects (to account for time-invariant unobserved differences across queries and common temporal shocks) and clustered the standard errors at the query level (to account for serial correlation within a query over time). In Column (1) of Table 2, we first replicate the baseline effect in this user-query-day level dataset of the random sample. That is, the API removal leads to a decrease in CTR on search suggestions. In Column (2) of Table 2, we provide evidence to support the hypothesis of improved prediction due to internal data accumulation within the same user. We defined a time-varying indicator variable, **Repeated Queries** (=1 if query j has been searched by user i prior to day t, 0 otherwise).

We can see that this interaction term is positive and significant, implying that the negative impact of API removal is smaller for terms queried by the same user before. Furthermore, in Column (3), we analyze whether the negative impact of API removal is smaller as the number of times a query term has been input by other users apart from the focal user. We define a time-varying cumulative measure, **Query Histories**: that is, the logarithm of the frequency of query j that has been searched by other users apart from user i prior to day t. Again, we see that the interaction term is positive and significant, suggesting that the negative impact of API removal is weakened as queries establish their histories via user searches over time. These results, along with Figure 10 where search suggestions across the treatment and control conditions were becoming different over time, suggest the prevalence of within-user and across-user learning based on internal data in the absence of the market leader's API (Hagiu and Wright, 2023; Schaefer and Sapi, 2023).[12]

[12]Notably, unlike analyses in Online Appendix B, a query with more internal searches over time does not necessarily imply that the category where this query belongs is popular, since there are numerous queries in a given category.

Table 2: Evidence on the Role of Internal Data Accumulation

Variables	(1) CTR	(2) CTR	(3) CTR
API Removal	-0.0030***	-0.0171***	-0.0044***
	(0.0005)	(0.0040)	(0.0009)
API Removal × Repeated Query		0.0145***	
		(0.0041)	
API Removal × Query Histories			0.0004**
			(0.0002)
Unit of analysis	User-Query-Day	User-Query-Day	User-Query-Day
Query fixed effects	✓	✓	✓
Day fixed effects	✓	✓	✓
R^2	0.2289	0.2289	0.2289
Observations	1,636,900	1,636,900	1,636,900

* $p<0.1$, ** $p<0.05$, ***$p<0.01$. Standard errors in parentheses are clustered at the query level. Intercepts are omitted. Repeated Query is equal to 1 if the query j has been searched by user i prior to day t, 0 otherwise. Query Histories is defined as the logarithm of the frequency of query j that has been searched by other users apart from user i prior to day t. Data represent detailed log records of a random sample of 11,630 treated users and 11,913 control users since June 26, 2021.

Collectively, there are several takeaways from this section. First, a short-term evaluation would have made us overstate the impact of the market leader's data capability. Second, longer-term effects are unlikely to be driven by (1) diminishing returns to external candidate items, (2) user adaptation, (3) dynamic self-selection, or (4) unobserved spillover. Third, we provide evidence that suggests a plausible mechanism of the longer-term effects: the focal company's reliance on the leading search engine's data capability tends to limit the improvement of its own autocomplete predictions based on the accumulation of internal data. These findings highlight a critical trade-off between short-term benefits and potential long-term gain.

5 An Extension with Field Experiment 2

In the previous sections, we have demonstrated the nuanced effects of the market leader's data capability on the company's search product performance. In this section, we leverage another

one-day field experiment in July 2021 to achieve three objectives: (1) replicate the main results from the first field experiment, (2) validate the manipulation of search suggestions provided by the market leader's API, and (3) assess the relative impact of manipulating the supply of candidates into the ranking algorithm versus directly manipulating the rank of external candidates in the same experimental setup.

5.1 Experimental Design

Similar to field experiment 1, the randomization in this experiment was implemented at the user level (see the randomization checks in the Online Appendix, Table A2). A total of 250,281 users were randomly assigned to one of three conditions:

Control (N=83,500): This condition is similar to the control condition in field experiment 1. Search suggestions are generated by the proprietary algorithm funnel. At the ranking stage, the ranking algorithm scores candidate items from two sources: (1) those retrieved at the candidate generation stage and (2) those supplied by the market leader's API. The ranking algorithm scores such candidate items and generates highest-scoring search suggestions in a ranked order.

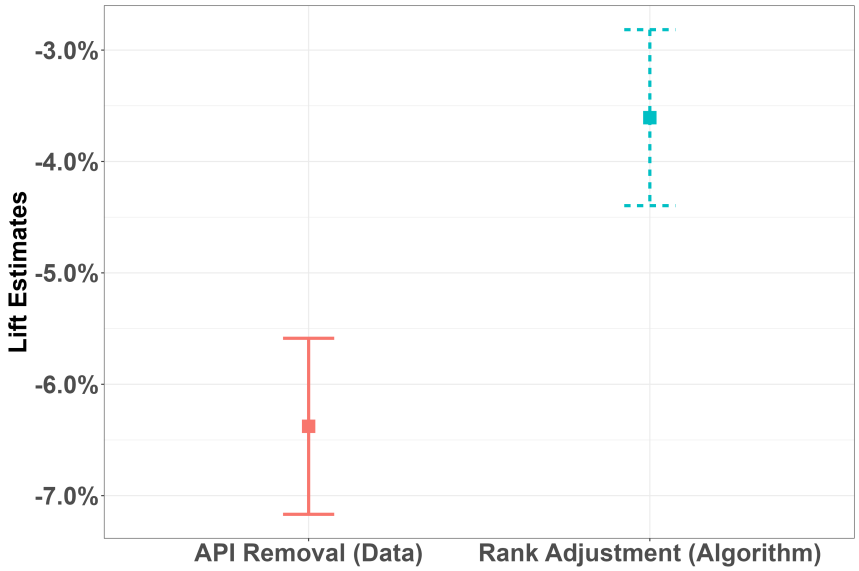
Rank Adjustment (N=83,517): Identical to the control condition, at the ranking stage, the ranking algorithm scores candidate items from two sources: (1) those retrieved at the candidate generation stage and (2) those supplied by the market leader's API. In contrast to the control condition, the ranking algorithm lowered (boosted) the rank of the market leader's candidate items (the company's own candidate items) and then generated the final search suggestions. As a result, relative to those in the control condition, users in this condition are less likely to see search suggestions supplied by the market leader in the final list.

Removal of Access to the Market Leader's API (N=83,264): This condition is similar to the treatment condition in field experiment 1. The ranking algorithm (the same as the one used in the control condition) scores only candidate items retrieved from the candidate generation stage and generates highest-scoring search suggestions in a ranked order. As a result, users in this condition never see search suggestions supplied by the market leader.

5.2 Results

Figure 11 shows that the removal of the market leader's API leads to a decrease in CTR by 6.4%. This estimate is between the aggregate estimate (i.e., –4.6%) and first-week estimate (i.e., –8.2%) from the main experiment. In addition, the estimate of the main field experiment for week 10 is about 4.9%. Hence, we are able to broadly replicate the main effect reported earlier.

Figure 11: The Relative Impact of API Removal and Rank Adjustment



Notes: N= 250,281. Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. Error bars represent 95% confidence intervals of lift estimates, which are calculated using the Delta method.

Turning to rank adjustment, we find that pushing down the ranking of search suggestions from the market leader leads to a decrease in CTR by 3.6%. This estimate supports that manipulating the rank directly has a significant impact on product quality. In practice, such an adjustment broadly relates to Google's adjustment in its ranking algorithms. Specifically, Google's white paper notes that "where our algorithms detect that a user's query relates to a "Your Money or Your Life (YMYL)" pages topic, we will give more weight in our ranking systems to factors like our understanding of the authoritativeness, expertise, or trustworthiness of the pages we present in response (Google, 2019)."

Taken together, these estimates provide suggestive evidence that in our context, manipulating the rank of the market leader's candidate items induces a smaller impact on CTR relative to the removal of access to the market leader's candidate items for the ranking algorithm. Another takeaway is that these estimates shed light on the strength of the manipulation across different conditions. Conceptually, the manipulation of pushing down the rank of search suggestions from the market leader should be weaker than completely removing the market leader's candidate items. Therefore, the magnitude of the estimates (6.4% versus 3.6%) increases our confidence in the success of the manipulation. Finally, given the intertemporal trade-off between short-term gain and longer-term development, a practical takeaway of this experiment is that adjusting the rank of the market leader's candidate items could be a "balanced" strategy that could allow a company to benefit from API access but also collect more individual-level data due to more clicks on internal candidates.

6 Discussion

We leverage a field experiment where we exogenously removed access to the market leader's data capability in the search-suggestion generating process. We find that, without access to the market leader's API, users click on search suggestions 4.6% less relative to those users in the condition with such access Moreover, we find that a 10% increase (decrease) in CTR on search suggestions leads to a 1.85% increase (decrease) in CTR on the top-slot link on SERP. The length of the large-scale experiment enables us to document significant dynamics in how users interact with the product when there is no access to external data capabilities. In particular, we find that the negative effect of removing API access is less negative over time. Our mechanism analyses suggest that such dynamics are likely to be driven by the improvement in the company's autocomplete predictions through accumulating more internal data. Finally, using a second (short-term) experiment, we manipulated (i) access to the market leader's API and (ii) the rank of search-suggestion candidates to ensure that it is indeed the market leader's data capability that lead to the effects we measure.

Our paper has notable managerial implications, highlighting a nuanced story for companies looking to leverage external data capabilities. The results suggest that leveraging external data

capabilities can provide an economically meaningful return for a company, especially in the early stages of new product development. For search products, in particular, we show that access to depersonalized search results can increase engagement not only on the focal product, but also downstream on the SERP. This is important because several search products are launched periodically (e.g., Cliqz) that could look to this potential strategy. However, we also highlight a trade-off between the short-run gains and longer-run product development: that is, reliance on external data capabilities over the longer term can limit the focal firm's organic development using internal data.

We believe our results also shed light on policy issues that are currently being debated. While it might be consequential for a company in isolation, given the degree of concentration in search markets in different countries (e.g., Google in the US and Baidu in China), access to the market leader's data capability may not be able to significantly alter the market structure by reducing the barriers to entry sufficiently. Indeed, recent papers such as Allcott et al. (2024) demonstrate that changing the default search engine can have a significantly larger effect than what we find in our setting. That noted, sharing data capabilities across search engines could be part of a multi-pronged approach adopted by regulators to tackle market concentration by using different levers.

Our study has limitations. Like other studies with field experiments, we can only look at one setting. Hence, our estimates require assumptions to extrapolate to other contexts. Given that our field experiment varies external candidate items at only one stage of the algorithmic funnel, it would be prudent to replicate the general tenor of our findings in other contexts. This would, of course, depend on whether the opportunity exists to leverage such data-sharing agreements. Our study is also a partial equilibrium analysis, and further research could consider more general equilibrium dimensions. For example, there could be strategic responses from other (competing) platforms in the face of such data partnerships. Analyzing how the platform ecosystem evolves with such data focused partnerships or through regulations would be a fruitful next step. Finally, monetization strategies in search are evolving with companies' experimentation with thumbnails, featured snippets, and embedded links. While users of search engines might be getting used to these features, their responses could change as monetization occurs at different stages of the search funnel. Follow-up research could account for this while studying these topics.

References

Agrawal, A., Gans, J., and Goldfarb, A. (2018). *Prediction machines: the simple economics of artificial intelligence*. Harvard Business Press. https://dl.acm.org/doi/book/10.5555/3239975.

Allcott, H., Castillo, J. C., Gentzkow, M., Musolff, L., and Salz, T. (2024). Sources of market power in web search evidence from a field experiment. Technical report, National Bureau of Economic Research. https://lmusolff.github.io/papers/SearchMarket_NBER.pdf.

Alrashed, T., Almahmoud, J., Zhang, A. X., and Karger, D. R. (2020). Scrapir: making web data apis accessible to end users. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12. https://dl.acm.org/doi/10.1145/3313831.3376691.

Angrist, J. D. and Pischke, J.-S. (2009). *Mostly harmless econometrics: An empiricist's companion*. Princeton University Press. https://www.degruyter.com/document/doi/10.1515/9781400829828/html.

Benzell, S. G., Hersh, J., and Van Alstyne, M. (2023). How apis create growth by inverting the firm. *Management Science*. https://pubsonline.informs.org/doi/full/10.1287/mnsc.2023.4968.

Beraja, M., Yang, D. Y., and Yuchtman, N. (2023). Data-intensive innovation and the state: evidence from ai firms in china. *The Review of Economic Studies*, 90(4):1701–1723. https://academic.oup.com/restud/article/90/4/1701/6665906.

Brinkmann, D. (2022). Why real-time data pipelines are so hard. *MLOps Community*. https://mlops.community/why-real-time-data-pipelines-are-so-hard/.

Cai, F., De Rijke, M., et al. (2016). A survey of query auto completion in information retrieval. *Foundations and Trends® in Information Retrieval*, 10(4):273–363. https://www.nowpublishers.com/article/Details/INR-055.

Casella, G. and Berger, R. L. (2002). *Statistical inference*. Duxbury Press: Pacific Grove, CA, second edition. https://openlibrary.org/books/OL3943908M/Statistical_Inference.

Chan, T., Hamdi, N., Hui, X., and Jiang, Z. (2022). The value of verified employment data for consumer lending: Evidence from equifax. *Marketing Science*, 41(4):795–814. https://pubsonline.informs.org/doi/pdf/10.1287/mksc.2021.1335.

Chiou, L. and Tucker, C. (2017). Search engines and data retention: Implications for privacy and antitrust. Technical report, National Bureau of Economic Research. https://www.nber.org/papers/w23815.

Committee, A. M. (1994). Is my calibration linear? *Analyst*, 119(11):2363–2366. https://pubs.rsc.org/en/content/articlelanding/1994/an/an9941902363.

Covington, P., Adams, J., and Sargin, E. (2016). Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 191–198. https://dl.acm.org/doi/10.1145/2959100.2959190.

Deng, A., Knoblich, U., and Lu, J. (2018). Applying the delta method in metric analytics: A practical guide with novel ideas. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 233–242. https://www.microsoft.com/en-us/research/publication/applying-the-delta-method-in-metric-analytics-a-practical-guide-with-novel-ideas/.

Duranton, S., Gourévitch, A., Baltassis, E., Khendek, Y., Quarta, L., Fernández, M., and Rubio, M. M. (2021). Is your company gaining momentum in data? *Boston Consulting Group*. https://www.bcg.com/publications/2021/companies-data-capabilities-progress.

Fang, L., Chen, Y., Chiara, F., Yuan, Z., and Wang, Y. (2024). Platform information provision and consumer search: A field experiment. Technical report, National Bureau of Economic Research. https://www.nber.org/papers/w32099.

Goli, A., Huang, J., Reiley, D., and Riabov, N. (2018). Measuring consumer sensitivity to audio advertising: A field experiment on pandora internet radio. *Available at SSRN 3166676*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3166676.

Goli, A., Reiley, D. H., and Zhang, H. (2024). Personalizing ad load to optimize subscription and ad revenues: Product strategies constructed from experiments on pandora. *Marketing Science*. https://pubsonline.informs.org/doi/full/10.1287/mksc.2022.0357.

Google (2019). How google fights disinformation. *White Paper*. https://safety.google/intl/en_uk/stories/fighting-misinformation-online/.

Gordon, B. R., Moakler, R., and Zettelmeyer, F. (2023). Close enough? a large-scale exploration of non-experimental approaches to advertising measurement. *Marketing Science*, 42(4):768–793. https://pubsonline.informs.org/doi/full/10.1287/mksc.2022.1413.

Gulli, A. (2013). A Deeper Look at Autosuggest. *Bing Blog*, https://blogs.bing.com/search/March-2013/A-Deeper-Look-at-Autosuggest. .

Gupta, A. K., Smith, K. G., and Shalley, C. E. (2006). The interplay between exploration and exploitation. *Academy of Management Journal*, 49(4):693–706. https://journals.aom.org/doi/10.5465/amj.2006.22083026.

Hagiu, A. and Wright, J. (2023). Data-enabled learning, network effects, and competitive advantage. *The RAND Journal of Economics*, 54(4):638–667. https://onlinelibrary.wiley.com/doi/full/10.1111/1756-2171.12453.

Havakhor, T., Rahman, M. S., Zhang, T., and Zhu, C. (2024). Tech-enabled financial data access, retail investors, and gambling-like behavior in the stock market. *Management Science*. https://pubsonline.informs.org/doi/full/10.1287/mnsc.2021.01379.

Holtz, D., Brennan, J., and Pouget-Abadie, J. (2023). A study of "symbiosis bias" in a/b tests of recommendation algorithms. *arXiv preprint arXiv:2309.07107*. https://arxiv.org/abs/2309.07107.

Johansson, J. K. (1979). Advertising and the s-curve: A new approach. *Journal of Marketing Research*, 16(3):346–354. https://journals.sagepub.com/doi/abs/10.1177/002224377901600307?journalCode=mrja.

Klein, T. J., Kurmangaliyeva, M., Prüfer, J., Prüfer, P., and Park, N. N. (2022). How important are user-generated data for search result quality? experimental evidence. Technical report, CEPR Discussion Paper DP17934. https://repec.cepr.org/repec/cpr/ceprdp/DP17934.pdf.

Kohavi, R., Deng, A., Frasca, B., Walker, T., Xu, Y., and Pohlmann, N. (2013). Online controlled experiments at large scale. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1168–1176. https://dl.acm.org/doi/10.1145/2487575.2488217.

Kopliku, A., Pinel-Sauvagnat, K., and Boughanem, M. (2014). Aggregated search: A new information retrieval paradigm. *ACM Computing Surveys (CSUR)*, 46(3):1–31. https://dl.acm.org/doi/10.1145/2523817.

Kucharavy, A., Schillaci, Z., Maréchal, L., Würsch, M., Dolamic, L., Sabonnadiere, R., David, D. P., Mermoud, A., and Lenders, V. (2023). Fundamentals of generative large language models and perspectives in cyber-defense. *arXiv preprint arXiv:2303.12132*. https://arxiv.org/abs/2303.12132.

Laverty, K. J. (1996). Economic "short-termism": The debate, the unresolved issues, and the implications for management practice and research. *Academy of Management Review*, 21(3):825–860. https://journals.aom.org/doi/abs/10.5465/amr.1996.9702100316.

Li, H. and Kettinger, W. B. J. (2021). The building blocks of software platforms: understanding the past to forge the future. *Journal of the Association for Information Systems, Forthcoming*, 22(6):1524–1555. https://aisel.aisnet.org/jais/vol22/iss6/9/.

Little, J. D. (1979). Aggregate advertising models: The state of the art. *Operations Research*, 27(4):629–667. https://pubsonline.informs.org/doi/abs/10.1287/opre.27.4.629.

Mitra, B. and Craswell, N. (2015). Query auto-completion for rare prefixes. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1755–1758. https://dl.acm.org/doi/10.1145/2806416.2806599.

Nagaraj, A. (2022). The private impact of public data: Landsat satellite maps increased gold discoveries and encouraged entry. *Management Science*, 68(1):564–582. https://pubsonline.informs.org/doi/abs/10.1287/mnsc.2020.3878.

Nandy, P., Venugopalan, D., Lo, C., and Chatterjee, S. (2021). A/b testing for recommender systems in a two-sided marketplace. In *Proceedings of the 35th Neural Information Processing Systems*, pages 6466–6477. https://dl.acm.org/doi/abs/10.5555/3540261.3540756.

OpenAI (2024). Terms of Use. *OpenAI Blog*. https://openai.com/policies/terms-of-use/.

Park, D. H. and Chiba, R. (2017). A neural language model for query auto-completion. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1189–1192. https://dl.acm.org/doi/10.1145/3077136.3080758.

Peukert, C., Sen, A., and Claussen, J. (2024). The editor and the algorithm: Recommendation technology in online news. *Management Science*, 70(9):5816–5831. https://pubsonline.informs.org/doi/full/10.1287/mnsc.2023.4954.

Rahmandad, H. (2012). Impact of growth opportunities and competition on firm-level capability development trade-offs. *Organization Science*, 23(1):138–154. https://pubsonline.informs.org/doi/full/10.1287/orsc.1100.0628.

Sanderson, M. (2008). Ambiguous queries: test collections need more sense. In *Proceedings of the 31st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 499–506. https://dl.acm.org/doi/10.1145/1390334.1390420.

Sarwar, B. M. (2001). *Sparsity, scalability, and distribution in recommender systems*. University of Minnesota. https://dl.acm.org/doi/book/10.5555/932549.

Schaefer, M. and Sapi, G. (2023). Complementarities in learning from data: Insights from general search. *Information Economics and Policy*, 65:101063. https://www.sciencedirect.com/science/article/pii/S0167624523000483.

Serban, I., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, volume 30. https://dl.acm.org/doi/10.5555/3016387.3016435.

Stoica, I., Song, D., Popa, R. A., Patterson, D., Mahoney, M. W., Katz, R., Joseph, A. D., Jordan, M., Hellerstein, J. M., Gonzalez, J. E., et al. (2017). A berkeley view of systems challenges for ai. *arXiv preprint arXiv:1712.05855*. https://www2.eecs.berkeley.edu/Pubs/TechRpts/2017/EECS-2017-159.html.

Sullivan, D. (2018). How Google autocomplete works in Search. *Google Blog*, https://blog.google/products/search/how-google-autocomplete-works-search/. .

Sun, T., Yuan, Z., Li, C., Zhang, K., and Xu, J. (2024). The value of personal data in internet commerce: A high-stakes field experiment on data regulation policy. *Management Science*, 70(4):2645–2660. https://pubsonline.informs.org/doi/full/10.1287/mnsc.2023.4828.

Tang, D., Agarwal, A., O'Brien, D., and Meyer, M. (2010). Overlapping experiment infrastructure: More, better, faster experimentation. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 17–26. https://dl.acm.org/doi/abs/10.1145/1835804.1835810.

Ubaldi, B. (2013). Open government data: Towards empirical analysis of open government data initiatives. Technical report. https://www.oecd-ilibrary.org/governance/open-government-data_5k46bj4f03s7-en.

Ursu, R. M. (2018). The power of rankings: Quantifying the effect of rankings on online consumer search and purchase decisions. *Marketing Science*, 37(4):530–552. https://pubsonline.informs.org/doi/10.1287/mksc.2017.1072.

Wernerfelt, N., Tuchman, A., Shapiro, B., and Moakler, R. (2024). Estimating the value of offsite data to advertisers on meta. *Marketing Science*. https://pubsonline.informs.org/doi/10.1287/mksc.2023.0274.

Xu, Y., Chen, N., Fernandez, A., Sinno, O., and Bhasin, A. (2015). From infrastructure to culture: A/b testing challenges in large scale social networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2227–2236. https://dl.acm.org/doi/10.1145/2783258.2788602.

Xue, L., Song, P., Rai, A., Zhang, C. G., and Zhao, X. (2019). Implications of application programming interfaces for third-party new app development and copycatting. *Production and Operations Management*, pages 1887–1902. https://onlinelibrary.wiley.com/doi/full/10.1111/poms.13021.

Yang, J., Sahni, N. S., Nair, H. S., and Xiong, X. (2024). Advertising as information for ranking e-commerce search listings. *Marketing Science*, 43(2):360–377. https://pubsonline.informs.org/doi/full/10.1287/mksc.2021.0292.

Yoganarasimhan, H. (2020). Search personalization using machine learning. *Management Science*, 66(3):1045–1070. https://pubsonline.informs.org/doi/10.1287/mnsc.2018.3255.

Zaif, Z. (2023). Search box optimization. *The Medium*. https://googlerankingexpert.medium.com/search-box-optimization-ff883b2fb56e.

Zaveri, A., Dastgheib, S., Wu, C., Whetzel, T., Verborgh, R., Avillach, P., Korodi, G., Terryn, R., Jagodnik, K., Assis, P., et al. (2017). smartapi: towards a more intelligent network of web apis. In *The Semantic Web: 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28–June 1, 2017, Proceedings, Part II 14*, pages 154–169. Springer. https://dl.acm.org/doi/abs/10.1007/978-3-319-58451-5_11.

Zeleti, F. A. and Ojo, A. (2017). Open data value capability architecture. *Information Systems Frontiers*, 19(2):337–360. https://link.springer.com/article/10.1007/s10796-016-9711-5.

Zhang, X. and Feng, J. (2011). Cyclical bid adjustments in search-engine advertising. *Management Science*, 57(9):1703–1719. https://pubsonline.informs.org/doi/abs/10.1287/mnsc.1110.1408.

Zhao, Y., Yildirim, P., and Chintagunta, P. K. (2023). Privacy regulations and online search friction: Evidence from gdpr. *Marketing Science Institute Working Paper*. https://www.msi.org/working-paper/privacy-regulations-and-online-search-friction-evidence-from-gdpr/.

Zheng, S., Tong, S., Kwon, H. E., Burtch, G., and Li, X. (2023). Recommending what to search: Sales volume and consumption diversity effects of a query recommender system. *Available at SSRN 4667778*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4667778.

7 Online Appendix

Appendix A Additional Tables and Figures

A.1 Tables

A.1.1 Comparison between Search and Recommendation

Table A1: Search Engine vs. Homepage Recommendations

	Search Engine	Homepage Recommendations
Basis	Rank items based on degree of match with explicit search query	Rank items based on implicitly collected engagement data
Explicitness of ranking signals	High	Low
Use of personalization	Low/Moderate	High

A.1.2 Randomization Checks: Field Experiment 2

Table A2: Randomization Checks: Field Experiment 2

User Characteristics	Control (C)	Algorithm (T1)	Data (T2)	*p.* value Diff (T1,C)	*p.* value Diff (T2,C)
Male	0.5714	0.5671	0.5671	0.0729	0.0755
	(0.0017)	(0.0017)	(0.0017)		
Larger Cities	0.6327	0.6312	0.6334	0.5228	0.7714
	(0.0017)	(0.0017)	(0.0017)		
Smaller Cities	0.3292	0.3281	0.3269	0.6152	0.2983
	(0.0016)	(0.0016)	(0.0016)		
Active days in the past 30 days (search activities)	100	100.0533	99.8655	0.8609	0.6571
	(0.2144)	(0.2156)	(0.2141)		
Query views in the past 30 days (search activities)	100	100.4362	100.3986	0.5468	0.5775
	(0.5058)	(0.5179)	(0.5061)		

Notes: This table shows the balance along several observable dimensions between users in the treatment condition and those in the control condition. Following the hierarchical classification of Chinese cities, larger cities include tier 1 to 4 cities (e.g., tier 1: largest cities such as Beijing), whereas smaller cities refer to tier 5 cities and below. p value is obtained based on a two-sided t-test on the equality of means with unequal variances. For confidentiality purposes, numbers in the last two rows were normalized so that the variable means in the control condition are 100.

A.1.3 Robustness Checks

Table A3: Robustness Checks

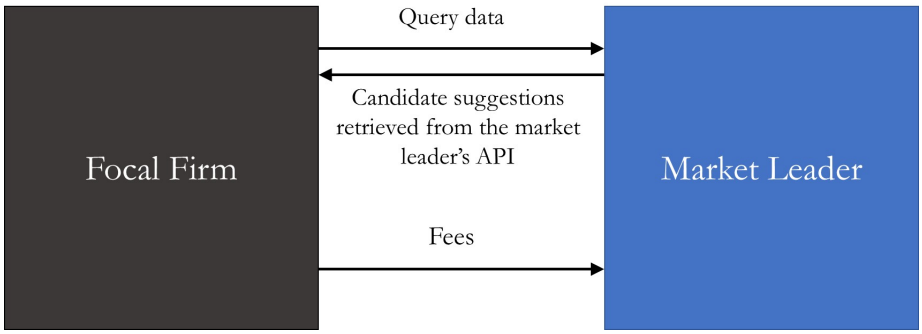
	(1) Lift of CTR	(2) Lift of CTR	(3) Click Probability	(4) ln(Clicks)	(5) CTR
API Removal	-0.0459***	-0.0539***	-0.0103***	-0.0329***	-0.0124***
	(0.0014)	(0.0013)	(0.0009)	(0.0018)	(0.0010)
API Removal×Female					-0.0016
					(0.0010)
API Removal×Smaller Cities					0.0013
					(0.0010)
API Removal×New User					0.0068***
					(0.0024)
API Removal×Active Days					-0.0085***
					(0.0013)
API Removal×Query Views					-0.0104***
					(0.0013)
Control variables	No	Yes	No	No	No
R^2	0.0004	0.0126	0.0001	0.0001	0.0126
Observations	2,388,377	1,932,886	2,390,244	2,390,244	1,932,886

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. In Column (1), CTR is computed using the first day of each user in the sample. In Columns (2) to (5), CTR is computed across the entire experiment (108 days). In Column (2), controls variables include gender, city size, and user activity (query views, active days). Whereas we use a linear probability model in Column (3), we use the logarithm of (1+clicks) as the dependent variable in Column (4). Column (5) includes all interaction terms of control variables with treatment status.

A.2 Figures

A.2.1 Mechanics of the Market Leader's API

Figure A1: How Does the Market Leader's API Work?



A.2.2 Randomization Checks

Figure A2: Proportion of Users Assigned to the Treatment Condition Over Time

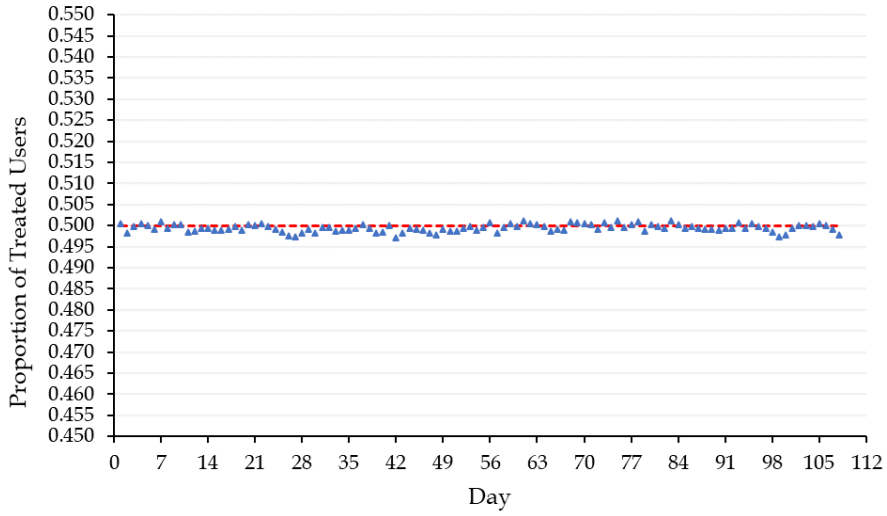


Figure A3: Proportion of New Users Assigned to the Treatment Condition Over Time

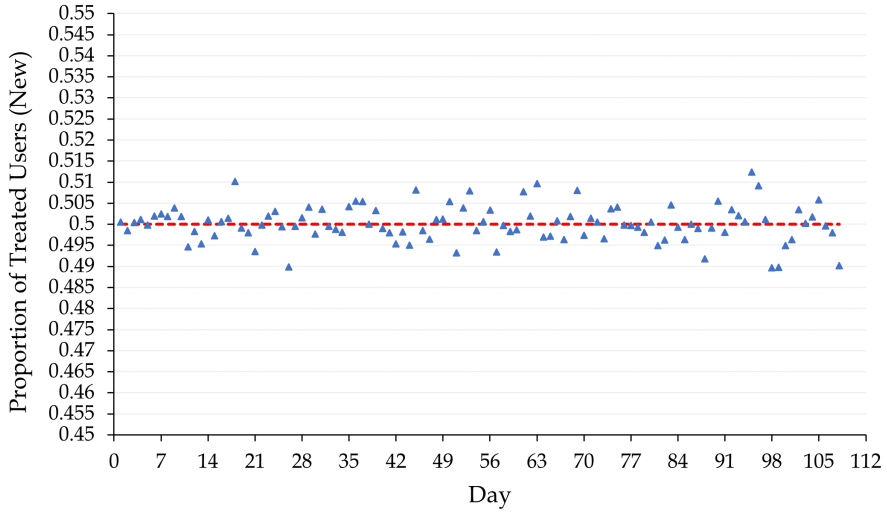
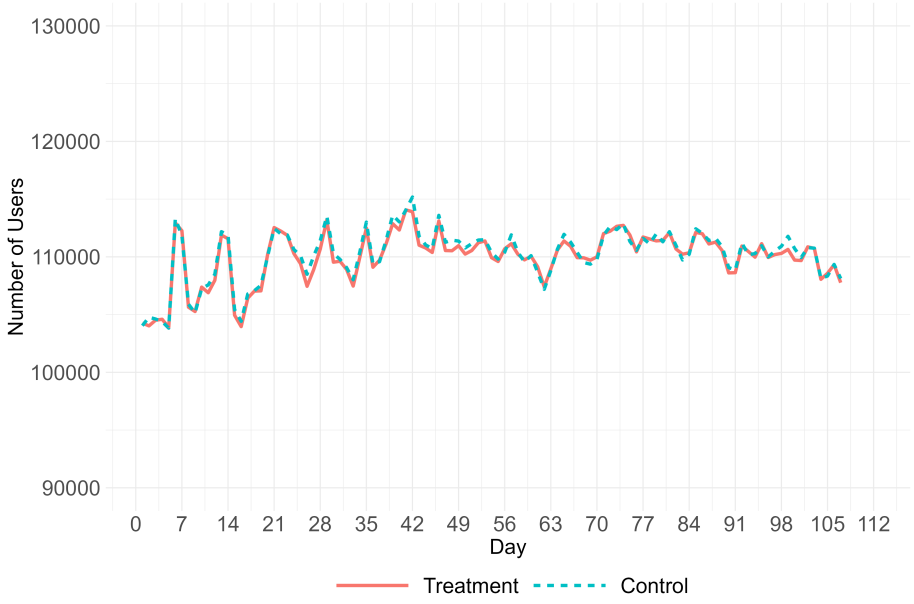
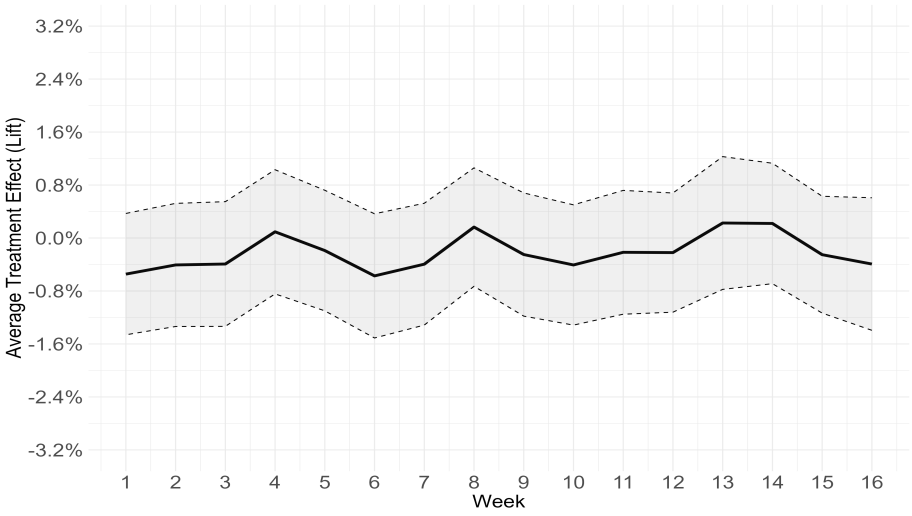


Figure A4: Number of Unique Daily Active Users in Treatment vs. Control Over Time



A.2.3 Ruling out Attrition

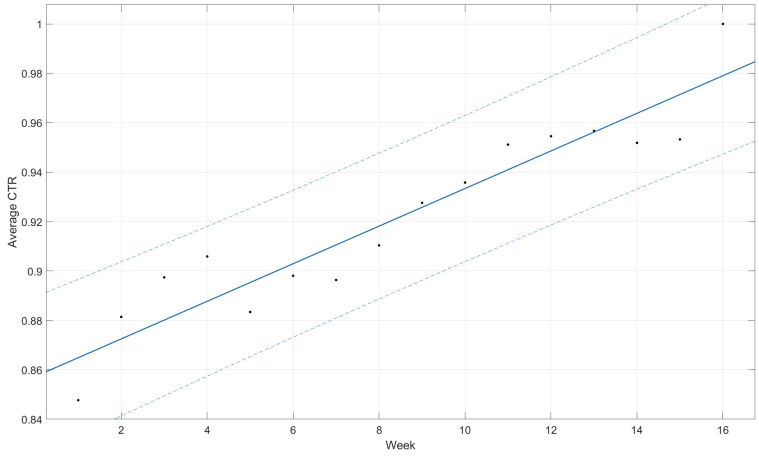
Figure A5: Longer-Term Effects of API Removal on Query Volume



Notes: Error bands represent 95% confidence intervals of lift estimates, which are calculated using the Delta method.

A.2.4 Temporal Variation in CTR among Users in the Control Condition

Figure A6: Temporal Variation in CTR among Users in the Control Condition



Notes: For confidentiality purposes, we added an undisclosed constant to the weekly average clickthrough rates (CTRs). Error bands represent 95% confidence intervals.

Appendix B Treatment Effects by Query Category

In this section, we explore how treatment effects vary by popularity of the categories queries belong to. When users enter a query term, the company's natural language processing tool can classify the content category of each term into 31 first-level categories (e.g., Health) and 167 second-level categories (e.g., Health-Disease). Conceptually, we can define the query popularity based on the number of searches for a specific content category (e.g., query in a popular or niche content category). However, we did not observe the information about query type before the experiment (unlike demographics, active days), so we cannot use pre-experimental query popularity as a potential source of heterogeneity. Using the post-treatment query popularity might cause a bad control problem (Angrist and Pischke, 2009). Therefore, we constructed the popular-niche threshold and split the sample based on clicks from users in the control condition on the first day of the experiment. Further, we examine only the first search occasion of users to restrict our analysis to their first exposure to the treatment. Our rationale is that, like any random sample of the user base of the focal search product, users in the control condition represent the status quo. Therefore, under the assumption of no spillovers across the two conditions, the variation in query popularity in the control condition shall mirror the population of interest and not be affected by the treatment. Further, focusing on the first day of the experiment alleviates the concern of violating SUTVA.

Specifically, we use the search records of users who clicked at least once and define the content categories that generate 75% of the clicks in the control condition on the first day of the experiment as popular (mainstream) content (e.g., Education-K-12, Books-Novel, Health-Disease) and the remaining 25% as niche content (e.g., Music-Music Radio, Government Affairs-Nonprofit Organization). In Table B1, we verified that the user demographics and usage activities are balanced among users in the treatment and control conditions in this subsample. Columns (1) and (2) of Table B2 suggest that the negative treatment effect is driven by popular queries. A plausible explanation is that the candidate items from the market leader's API are generated based on real searches by all users on the market leader's platform but are not personalized. In contrast, clicking niche content is not affected by API removal because recommending niche content relies more on personal data. This demonstrates that external candidate items informed by the market leader's real searches may provide better suggestions for popular queries that require less personalization.

Table B1: Randomization Checks

User Characteristics	Control	Treatment	p. value
Male	0.5151	0.5135	0.1520
	(0.0008)	(0.0008)	
Larger Cities	0.5386	0.5381	0.6793
	(0.0008)	(0.0008)	
Smaller Cities	0.4560	0.4563	0.7868
	(0.0008)	(0.0008)	
Mobile Operating System: Apple iOS	0.1161	0.1156	0.5405
	(0.0005)	(0.0005)	
Mobile Operating System: Android	0.8831	0.8836	0.5393
	(0.0005)	(0.0005)	
Active days in the past 30 days (search activities)	100	99.5790	0.2119
	(0.2390)	(0.2379)	
Query views in the past 30 days (search activities)	100	99.8770	0.8406
	(0.4303)	(0.4346)	

Notes. This table shows the balance between users in the treated relative to control groups along several observable dimensions for those who have clicked on a search suggestion at least once on the first day for each user. Following the hierarchical classification of Chinese cities, larger cities include tier 1 to 4 cities (e.g., tier 1: largest cities such as Beijing), whereas smaller cities refer to tier 5 cities and below. p-value is obtained based on a two-sided t-test on the equality of means with unequal variances. For confidentiality purposes, values reported in the last two rows were normalized so that the variable means in the control condition are 100.

Table B2: Treatment Effects by Query Type

	(1) Lift of CTR (Popular Categories)	(2) Lift of CTR (Niche Categories)
API Removal	-0.0134***	0.0021
	(0.0017)	(0.0050)
Sample time period	First Day	First Day
R^2	0.0001	0.0000
Observations	764,119	764,119

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. We estimate a linear regression and calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method.

Appendix C Back-of-the-Envelope Calculation

In this section, we carry out a back-of-the-envelope calculation to shed light on the economic implications of access (or lack thereof) to the market leader's candidate items over the course of the experimental period. The monetary value of the CTR on search suggestions can be decomposed into two components: (1) direct revenue gain from clicking on the search suggestions (i.e., tied to the average treatment effect of 4.6%) and (2) indirect revenue gain from clicking on search results through search suggestions (i.e., tied to the elasticity of 0.185). Next, we provide details of these computations while explicitly discussing the assumptions we make in the process.

Regarding (1), an average user generates 6.45 search-suggestion clicks per month; on average, the unit query revenue associated with a search-suggestion click is 0.012 Chinese yuan (CNY). Unit query revenue is equal to the total search keyword auction revenue divided by the total number of searches.[13] The number of monthly active users of the search product are approximately 100 million toward the end of our experiment.[14] For these 100 million users, a 4.6% decrease in search-suggestion CTR results in a loss of 356,000 CNY per month (i.e., $6.45 \times 0.012 \times 0.046 \times 100,000,000$) or $50,000 per month. Regarding (2), an average user generates 6.45 search-suggestion clicks per month; according to an industry report from Dongxing Securities, each click generates revenue of 0.35 CNY for smaller search engines such as our partner company[15]; the elasticity of CTR on the top-slot search result with respect to CTR on search suggestions is 0.185. Therefore, a 4.6% decrease in search-suggestion CTR creates a loss of 1,925,000 CNY per month (i.e., $0.046 \times 0.185 \times 6.45 \times 0.35 \times 100,000,000$) or $275,000 per month through a decrease in CTR on SERP. The total revenue loss due to API removal is approximately $3.9 million annually [$12 \times (\$50{,}000 + \$275{,}000)$]. We believe this is an economically significant number for a team that launched a new search product.

Notably, this calculation is very conservative. For example, based on industry estimates, unit query revenue can grow tenfold at the very least as the search engine matures. In addition, 0.35 CNY

[13] For more details, see https://dataforseo.com/blog/google-autocomplete-api-for-keyword-research-tool and https://www.dragonmetrics.com/guide-to-keyword-research-for-baidu-seo/

[14] This number aligns with global estimates of the small search players such as Ecosia (20 million as of 2022), DuckDuckGo (100 million as of 2023) and Bing (500 million as of 2023). See https://techcrunch.com/2022/06/09/ecosia-updates/; https://techreport.com/statistics/duckduckgo-statistics/ and https://backlinko.com/bing-users.

[15] https://pdf.dfcfw.com/pdf/H3_AP201509210010852974_1.pdf.

is an average estimate across all slots, thus significantly underestimating the value of top-slot clicks since most monetization happens through top-slot clicks. Zhang and Feng (2011) show that top-slot click generates $0.4 at Yahoo!. Based on publicly available information, we use a conservative estimate that 20% of top-slot search results are ads,[16] a 4.6% decrease in search-suggestion CTR creates a loss of $5.3 million per year through a decrease in CTR on SERP (i.e., $0.046 \times 0.185 \times 6.45 \times 0.4 \times 0.2 \times 100{,}000{,}000 \times 12$). In this scenario, the total revenue loss due to API removal can be $5.9 million (5.3 million + $50,000 $\times$ 12) rather than 3.9 million.

This exercise comes with a few caveats. First, our calculation rests on the assumption that only clicks on top-slot search results generate revenue. While the literature suggests that the top slot gets the maximum clicks and revenue, some of the revenue could be generated from lower slots. In that scenario, we would have to account for the reduction in clicks on other slots on the SERP as a result of the treatment. Even after accounting for the reduction in CTR of other slots (elasticity is –0.089) and making the extreme assumption that all slots on the SERP generate the same revenue, the total revenue would range from approximately $2 to $3 million annually (using the following as the adjustment factor: [(0.185-0.089)/0.185]).

Second, we do not have proprietary information about the cost per query to the market leader's API, and industry estimates can vary significantly across contexts, scale, and time.[17] Moreover, under the regulations proposed, tapping into external APIs for data-related inputs will not be priced through the market but from an external fund to aid smaller companies. Hence, when such an external fund is set up, access to such an API can be done without cost, leading to the gains quantified above. At the very least, from a regulatory perspective, there have been proposals for smaller search companies to access data from gatekeeper search engines at fair, reasonable, and non-discriminatory (FRAND) rates. We expect FRAND rates to be considerably lower than current market rates since such APIs are a tool for growth and profits for platform companies (Benzell et al., 2023). That noted, based on our revenue estimates of $3.9 million, we can estimate the cost per query, making it worthwhile for the focal company to query the API each time through a market mechanism. An average user performs 14.5 searches per month, and the external API is queried for every search. Based on our lower bound revenue estimates, the break-even point is approximately

[16] https://www.justice.gov/d9/2023-10/416881.pdf
[17] See https://www.geekpark.net/news/334965 for some anecdotes.

$0.00022. Using the data for top-rank click revenues, we find that the break-even point is 50% higher at $0.00033. As mentioned before, the unit query revenue for search suggestions can grow 10-fold while it can at least double for SERP. This can make the break-even cost significantly higher and make leveraging external data capabilities useful in the short run.

Overall, this exercise allows us to highlight the economic benefits and costs of external data capabilities through the API, which has implications for the focal company, the search engine industry, and policymakers.

Appendix D Supplemental Information

Table D1: Average Treatment Effect and Heterogeneous Treatment Effects by User Characteristics

	(1) Lift of CTR Overall	(2) Lift of CTR Female	(3) Lift of CTR Male	(4) Lift of CTR Larger Cities	(5) Lift of CTR Smaller Cities	(6) Lift of CTR Heavy Users (Active Days)	(7) Lift of CTR Light Users (Active Days)	(8) Lift of CTR Heavy Users (Query Views)	(9) Lift of CTR Light Users (Query Views)
API Removal	-0.0459***	-0.0461***	-0.0472***	-0.0503***	-0.0408***	-0.0633***	-0.0352***	-0.0584***	-0.0323***
	(0.0011)	(0.0018)	(0.0015)	(0.0015)	(0.0017)	(0.0013)	(0.0016)	(0.0012)	(0.0019)
Unit of analysis	User	User	User	User	User	User	User	User	User
R^2	0.0007	0.0008	0.0008	0.0010	0.0006	0.0029	0.0004	0.0023	0.0003
Observations	2,390,244	822,128	1,201,234	1,204,821	1,036,051	780,820	1,369,873	1,056,338	1,094,355

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. CTR is computed as the ratio of the total number of clicks to total number of exposures over the entire experiment (108 days). Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. We estimate a linear regression as specified in Equation 1 and calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. Overall refers to the average treatment effect based on the full sample, while heterogeneity treatment effects are based on subsamples by user characteristics (gender, city size, user activity). Following the hierarchical classification of Chinese cities, "larger" cities include tier 1 to 4 cities (e.g., tier 1: largest cities such as Beijing), whereas "smaller" cities refer to tier 5 cities and below.

Table D2: Elasticity of Search-Result CTR with respect to Search-Suggestion CTR

A. IV Estimates	(1) $ln(CTR_SERP)$ (Top Slot)	(2) $ln(CTR_SERP)$ (Other Slots)
$ln(CTR_SUG)$	0.1854***	-0.0889***
	(0.0578)	(0.0333)
Unit of analysis	User	User
Observations	1,653,659	1,653,659
B. First-stage Estimates	$ln(CTR_SUG)$	$ln(CTR_SUG)$
API Removal	-0.1067***	-0.1067***
	(0.0047)	(0.0047)
First stage F-statistic (instr.)	506.85	506.85
Unit of analysis	User	User
Observations	1,653,659	1,653,659

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Robust standard errors in parentheses. Instrumental-variables (IV) estimates are obtained using two-stage least-squares (2SLS) regression. First-stage equation: $ln(CTR_SUG)_i = \alpha_1 + \beta_1 \times API_Removal_i + \epsilon_{1i}$. Structural equation: $ln(CTR_SERP)_i = \alpha_2 + \beta_2 \times ln(CTR_SUG)_i + \epsilon_{2i}$.

Table D3: Longer-Term Treatment Effects on Search-suggestion CTR (First 8 Weeks)

	(1) Week 1 Lift in CTR	(2) Week 2 Lift in CTR	(3) Week 3 Lift in CTR	(4) Week 4 Lift in CTR	(5) Week 5 Lift in CTR	(6) Week 6 Lift in CTR	(7) Week 7 Lift in CTR	(8) Week 8 Lift in CTR
A. All Users								
API Removal	-0.0818***	-0.0895***	-0.0851***	-0.0677***	-0.0612***	-0.0602***	-0.0655***	-0.0544***
	(0.0022)	(0.0022)	(0.0021)	(0.0021)	(0.0021)	(0.0021)	(0.0021)	(0.0021)
R^2	0.0021	0.0027	0.0025	0.0016	0.0013	0.0013	0.0015	0.0011
Observations	620,628	632,598	631,729	648,681	646,238	648,812	646,005	646,153
B. New Users								
API Removal	-0.0647***	-0.0727***	-0.0713***	-0.0486***	-0.0368***	-0.0400***	-0.0371***	-0.0248***
	(0.0026)	(0.0046)	(0.0054)	(0.0057)	(0.0063)	(0.0065)	(0.0069)	(0.0070)
R^2	0.0010	0.0011	0.0010	0.0004	0.0002	0.0003	0.0002	0.0001
Observations	621,837	238,988	179,006	160,526	138,896	128,614	118,327	113,202
C. Returning Users								
API Removal	-0.0994***	-0.0961***	-0.0885***	-0.0715***	-0.0654***	-0.0639***	-0.0698***	-0.0586***
	(0.0027)	(0.0023)	(0.0022)	(0.0022)	(0.0022)	(0.0021)	(0.0021)	(0.0021)
R^2	0.0041	0.0040	0.0033	0.0021	0.0017	0.0017	0.0019	0.0014
Observations	333,548	449,599	486,199	517,304	531,150	542,394	547,967	552,399

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. The regression for a given week is cross sectional as specified in Equation 1 where the CTR is computed as the ratio of the total number of clicks to total number of exposures until the end of that week. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. New users are defined on a weekly basis: new users at week t are those who have never used the search bar since the start of the experiment and used the search bar for the first time at week t (returning users, otherwise).

Table D4: Longer-Term Treatment Effects on Search-suggestion CTR (Last 8 Weeks)

	(1) Week 9 Lift in CTR	(2) Week 10 Lift in CTR	(3) Week 11 Lift in CTR	(4) Week 12 Lift in CTR	(5) Week 13 Lift in CTR	(6) Week 14 Lift in CTR	(7) Week 15 Lift in CTR	(8) Week 16 Lift in CTR
A. All Users								
API Removal	-0.0587***	-0.0488***	-0.0390***	-0.0364***	-0.0406***	-0.0428***	-0.0368***	-0.0447***
	(0.0021)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0025)
R^2	0.0013	0.0009	0.0006	0.0005	0.0006	0.0007	0.0005	0.0008
Observations	636,990	637,452	644,985	645,390	645,208	643,873	643,051	394,381
B. New Users								
API Removal	-0.0201***	-0.0324***	-0.0178**	-0.0117	-0.0153*	-0.0237***	-0.0139*	0.0027
	(0.0073)	(0.0074)	(0.0073)	(0.0074)	(0.0075)	(0.0076)	(0.0077)	(0.0122)
R^2	0.0001	0.0002	0.0001	0.0000	0.0000	0.0001	0.0000	0.0000
Observations	103,421	97,533	95,969	94,546	91,395	87,914	85,624	32,579
C. Returning Users								
API Removal	-0.0630***	-0.0508***	-0.0415***	-0.0391***	-0.0436***	-0.0451***	-0.0392***	-0.0479***
	(0.0021)	(0.0021)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0026)
R^2	0.0016	0.0011	0.0007	0.0006	0.0008	0.0009	0.0007	0.0010
Observations	551,893	557,002	565,718	567,284	570,147	571,960	572,820	365,845

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. The regression for a given week is cross sectional as specified in Equation 1 where the CTR is computed as the ratio of the total number of clicks to total number of exposures until the end of that week. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. New users are defined on a weekly basis: new users at week t are those who have never used the search bar since the start of the experiment and used the search bar for the first time at week t (returning users, otherwise).

Table D5: Longer-Term Treatment Effects on Query Volume (First 8 Weeks)

	(1) Week 1 Lift in QV	(2) Week 2 Lift in QV	(3) Week 3 Lift in QV	(4) Week 4 Lift in QV	(5) Week 5 Lift in QV	(6) Week 6 Lift in QV	(7) Week 7 Lift in QV	(8) Week 8 Lift in QV
A. All Users								
API Removal	-0.0055	-0.0041	-0.0039	0.0009	-0.0019	-0.0057	-0.0040	0.0016
	(0.0047)	(0.0047)	(0.0048)	(0.0048)	(0.0046)	(0.0048)	(0.0047)	(0.0046)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	622,910	635,060	634,320	651,470	649,045	651,730	648,917	649,132
B. New Users								
API Removal	-0.0030	-0.0044	-0.0035	0.0023	0.0013	-0.0101	-0.0046	0.0006
	(0.0033)	(0.0047)	(0.0057)	(0.0064)	(0.0071)	(0.0079)	(0.0077)	(0.0072)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	621,837	238,988	179,006	160,526	138,896	128,614	118,327	113,202
C. Returning Users								
API Removal	-0.0062	-0.0051	-0.0060	-0.0012	-0.0020	-0.0065	-0.0053	0.0025
	(0.0055)	(0.0049)	(0.0048)	(0.0048)	(0.0047)	(0.0048)	(0.0047)	(0.0046)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	333,548	449,599	486,199	517,304	531,150	542,394	547,967	552,399

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental number of queries entered by treated users relative to control users as a percentage of number of queries entered by control users. The regression for a given week is cross sectional as specified in Equation 1 where query volume (QV) is computed as total number of queries entered until the end of that week. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. New users are defined on a weekly basis: new users at week t are those who have never used the search bar since the start of the experiment and used the search bar for the first time at week t (returning users, otherwise).

Table D6: Longer-Term Treatment Effects on Query Volume (Last 8 Weeks)

	(1) Week 9 Lift in QV	(2) Week 10 Lift in QV	(3) Week 11 Lift in QV	(4) Week 12 Lift in QV	(5) Week 13 Lift in QV	(6) Week 14 Lift in QV	(7) Week 15 Lift in QV	(8) Week 16 Lift in QV
A. All Users								
API Removal	-0.0025	-0.0041	-0.0022	-0.0022	0.0023	0.0022	-0.0025	-0.0039
	(0.0047)	(0.0046)	(0.0048)	(0.0046)	(0.0051)	(0.0046)	(0.0045)	(0.0051)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	640,200	640,881	648,685	649,281	649,151	647,882	647,084	396,746
B. New Users								
API Removal	-0.0022	0.0042	-0.0086	0.0058	0.0059	0.0192**	0.0053	0.0083
	(0.0077)	(0.0077)	(0.0078)	(0.0073)	(0.0079)	(0.0088)	(0.0094)	(0.0134)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	103,421	97,533	95,969	94,546	91,395	87,914	85,624	32,579
C. Returning Users								
API Removal	-0.0026	-0.0058	-0.0013	-0.0033	-0.0002	0.0009	-0.0017	-0.0071
	(0.0048)	(0.0046)	(0.0047)	(0.0046)	(0.0052)	(0.0046)	(0.0045)	(0.0051)
R^2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Observations	551,893	557,002	565,718	567,284	570,147	571,960	572,820	365,845

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental number of queries entered by treated users relative to control users as a percentage of number of queries entered by control users. The regression for a given week is cross sectional as specified in Equation 1 where query volume (QV) is computed as total number of queries entered until the end of that week. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. New users are defined on a weekly basis: new users at week t are those who have never used the search bar since the start of the experiment and used the search bar for the first time at week t (returning users, otherwise).

Table D7: The Relative Impact of API Removal and Rank Adjustment on Search-suggestion CTR

	Lift of CTR
API Removal	-0.0638***
	(0.0040)
Rank Adjustment	-0.0361***
	(0.0040)
R^2	0.0010
Unit of analysis	User
Observations	250,281

* $p<0.1$, ** $p<0.05$, *** $p<0.01$. Lift refers to the incremental CTR among treated users relative to control users as a percentage of CTR among control users. CTR is computed as the ratio of the total number of clicks to total number of exposures over the duration of this experiment (1 day). The regression is specified as: $Y_i = \alpha_3 + \beta_3 \times API_Removal_i + \gamma_3 \times Rank_Adjustment + \epsilon_{3i}$. We calculate heteroskedasticity-robust standard errors. Standard errors of lift estimates in parentheses are calculated using the Delta method. The estimate of the intercept is omitted due to confidentiality.