

Estimating Consumer Preferences for LLMs: Evidence from LMArena

El Hadi Caoui

U of Toronto

September 18, 2026*

Abstract

This paper studies product differentiation in the consumer market for large language models using data from randomized pairwise comparisons of model responses. I estimate a differentiated-product demand model in which consumer tastes vary with prompt embeddings. Perceived model quality is strongly aligned with benchmark performance. Taste heterogeneity arises over stylistic and syntactic attributes such as readability, verbosity, tone, and formatting, along which LLMs differ. Counterfactual simulations show that changes in model intelligence divert demand both within and across providers' LLM portfolios, with the relative magnitude of these flows depending crucially on the extent of multi-homing. The mix of tasks each user performs determines which models are close substitutes, with a larger share of technical tasks increasing substitution among high-capability models.

Keywords: large language model, demand estimation, random coefficients, mixed logit

JEL Classification: L11, L13, L86

*elhadi.caoui@rotman.utoronto.ca. Department of Management, University of Toronto Mississauga and Rotman School of Management. 105 St George St, Toronto, ON M5S 3E6. I thank Andrey Simonov for a helpful discussion of this paper. I am grateful to Victor Aguirregabiria, Heski Bar-Isaac, Kevin A. Bryan, Tom Cunningham, Mert Demirer, Joshua Gans, Avi Goldfarb, Matt Mitchell, Regina Seibel, and seminar and conference participants at U of Toronto, IESE, Mines Paris, CEMFI, London School of Economics, Pompeu Fabra, Google, Mercor, and the Montreal Summer IO conference for helpful comments. The analysis, findings, and conclusions expressed in this paper should not be attributed to Arena Intelligence, Inc. All errors remain my own.

1 Introduction

Since the release of ChatGPT in November 2022, large language models (LLMs) have diffused rapidly. By mid-2026, more than one billion people were using LLM-based services each week and exchanging billions of messages per day. Survey evidence indicates that users rely on these models for a diverse set of tasks, spanning professional uses such as coding and research to non-work activities such as health advice and shopping decisions.

Despite this growth, little is known about how consumers evaluate and choose among available LLM products. These models appear differentiated along both vertical dimensions—such as core capabilities in reasoning or coding—and horizontal ones related to writing style, personality, or tone. Anecdotal evidence suggests that many users have strong preferences for particular models and continue to use them despite the availability of more capable alternatives.¹ Empirically, however, it remains unclear how consumers trade off these vertical and horizontal attributes or how changes in specific product features shape demand. Understanding these determinants of substitution patterns is key to evaluating the nature of competition in this industry.

This paper quantifies the role of product differentiation in consumer choice by leveraging a novel revealed-preference dataset from LMArena ([Chiang et al. \(2024\)](#)), a public platform for LLM evaluation. On this platform, users submit prompts, which are assigned to two randomly sampled and anonymized LLMs (in a “battle”), users vote for the preferred response, after which model identities are revealed. These comparisons show how consumers value differences in model outputs, revealing preferences in the consumer market for LLMs, where usage and conversation data are usually proprietary.

Using LMArena conversation data from 2024 and 2025, I first document several stylized facts about systematic differences in LLM outputs. Although it is well understood that models differ in their core capabilities and overall intelligence, I find that models also differ meaningfully along other dimensions, such as response formatting, readability (e.g., word choice and sentence structure), and tone. For instance, readability measures indicate that OpenAI’s models have become progressively easier to read over time, whereas Anthropic’s have become more complex. Similarly, response assertiveness has increased for OpenAI and Google models over time, but has remained relatively stable for Anthropic. These patterns highlight persistent and evolving differences across providers that are not reflected in headline

¹For example, OpenAI’s GPT-5.1 release note ([OpenAI \(2025\)](#)) describes the model as “warmer by default and more conversational,” while [Anthropic \(2025\)](#) emphasizes that “each Claude model has a unique character” and that some users find older versions more compelling despite improved capabilities. [The Economist \(2026\)](#) shows that LLMs differ in their social and political values: GPT models express especially secular views, while Gemini models place particularly strong emphasis on individual freedom.

benchmark scores.

The paper proceeds to build a structural demand model for LLMs in two main steps. First, I model voting choices in LMArena battles using a discrete choice framework with random utilities over LLM *responses*. User votes allow me to estimate how consumer utilities depend on response-level characteristics. In a second step, utilities over LLM responses are aggregated to define utilities over products (i.e., LLMs). Each user chooses products based on the expected utility over responses to their prompt distribution. Finally, I construct diversion ratios from counterfactual changes in model intelligence to quantify substitution across models and providers.

To recover consumers’ valuations for LLM responses, I specify a random-coefficients discrete choice model for LMArena battles. Departing from the standard approach, I allow taste parameters to vary not only across consumers (as in [Berry et al. \(1995\)](#)) but also across prompts. Specifically, the mean of the random coefficients is a function of a low-dimensional representation of the prompt’s vector embedding. This structure captures rich heterogeneity in preferences across use cases—such as coding versus writing—and within use cases, while keeping the model tractable. The paper contributes to the growing literature on demand estimation with unstructured data ([Compiani et al. \(2023\)](#), [Magnolfi et al. \(2025\)](#)) by proposing a novel method for dimensionality reduction of vector embeddings based on Partial least squares (PLS) regression.

Estimation results reveal that, on average, consumers prefer responses by more recently released, larger (by number of parameters), and reasoning models. Additionally, preferences are closely aligned with benchmark performance: consumers tend to favor responses generated by models that score highly on benchmarks. Conditional on benchmark scores, however, reasoning models are on average less preferred, which is consistent with their slower response speed. Consumers systematically prefer responses with positive sentiment (relative to neutral) and strongly dislike responses with negative sentiment or tentative responses.

Preferences vary substantially across prompts, underscoring the importance of use-case heterogeneity. Linguistic complexity, for example, is positively valued in technical tasks such as mathematics and coding, but is negatively associated with utility in most prompts involving practical guidance. The distribution of estimated coefficients further highlights key tradeoffs across benchmark performance: responses by LLMs with higher knowledge benchmark scores (e.g., MMLU-Pro, HLE, and GPQA Diamond) are consistently valued across all prompts. By contrast, the utility from responses by LLMs with higher coding benchmark scores (e.g., LiveCodeBench, SciCode) depends on the task: these scores increase utility for math and coding tasks but are negatively associated with utility in writing, casual conversation, and information-seeking prompts. This pattern suggests that gains in coding

ability may come at the expense of fluency or alignment in non-technical domains. Finally, formatting features—such as the use of bold text, tables, and structured lists—generate the largest utility gains in information-seeking and practical guidance queries.

Comparing preference heterogeneity across users and across prompts reveals different patterns for model capabilities and response attributes. For model intelligence, variation in tastes across prompts exceeds idiosyncratic variation across users, suggesting agreement across users on the value of model intelligence, conditional on the task. For response attributes such as formatting, readability, and tone, the reverse holds: preferences vary more across users than across prompts. These response attributes thus provide a basis for horizontal differentiation, reflecting widely dispersed tastes across the user population.

I use the estimated preferences over responses to specify a demand model for LLMs given the choice set of models available in the actual market in each period. This aggregation maps response-level utilities into model-level utilities by integrating over a prompt distribution that mirrors real-world query patterns observed on ChatGPT (Chatterji et al. (2025)). I account for potential multi-homing by considering two polar demand specifications. Under *prompt-level multi-homing*, users select the utility-maximizing LLM separately for each query. Under *single-homing*, users commit to a single firm across all queries and choose the firm whose portfolio of LLMs delivers the highest expected utility given the user’s prompt distribution. These two extremes bound plausible demand behavior. I then compute diversion ratios from counterfactual degradation in model intelligence to characterize substitution across LLMs. These ratios measure the fraction of a model’s lost demand that shifts to each alternative. They reveal which models are close substitutes and whether lost demand remains within a provider’s portfolio or flows to competing providers.

The counterfactual simulations show that, under prompt-level multi-homing, a 10-point reduction in a model’s intelligence—which brings, for example, Claude Opus 4’s intelligence score down to approximately that of Claude Sonnet 4—lowers its predicted demand by approximately 21–25% relative to baseline. Substitution is particularly strong within Google’s portfolio, where the other Gemini model absorbs 35–38% of the lost demand. Across providers, Google and OpenAI are the main destinations for each other’s lost demand. Homing behavior also affects these patterns. Under single-homing, degrading an Anthropic or OpenAI model reduces demand for the provider’s unchanged models as well: users switching to rival providers outweigh substitution within the portfolio, generating negative diversion.

Substitution patterns also vary with the distribution of use cases. When coding and mathematics account for a larger share of users’ queries, Claude Opus absorbs a greater proportion of the demand lost by OpenAI and Google, becoming a closer substitute for these providers’ high-capability models (e.g., Gemini 2.5 Pro), particularly under single-

homing. The closeness of competition thus depends on both consumers’ homing behavior and allocation of queries across providers, and the mix of tasks they perform.

The paper provides the first structural model of consumer demand for generative AI products, offering a framework for analyzing how users evaluate and substitute among models. As emphasized by [Berry and Haile \(2021\)](#), demand estimation is a necessary foundation for understanding market structure. This is particularly relevant given the growing policy interest in the generative AI sector, with competition authorities in the US, UK, and EU launching inquiries into its value chain ([Competition and Markets Authority \(2024a\)](#), [Competition and Markets Authority \(2024b\)](#), [Vestager et al. \(2024\)](#), [Federal Trade Commission \(2025\)](#)). This paper shows that product differentiation operates along multiple dimensions: model intelligence is the primary driver of demand, but firms also compete by better “aligning” their models to user tastes, particularly in non-technical use cases. By quantifying substitution patterns, the analysis provides a first step for evaluating how product differentiation shapes competition in this industry.

The rest of the paper proceeds as follows: after a literature review, [Section 2](#) provides background information on the industry. [Section 3](#) presents the data used in the empirical exercise. [Section 4](#) describes the model. [Section 5](#) discusses the estimation strategy and results. [Section 6](#) presents the results of counterfactual exercises, and [Section 7](#) concludes.

Literature Review. This paper contributes to two strands of the literature: the literature documenting the demand and use of LLMs and the IO literature on differentiated product demand estimation, in particular, that incorporating unstructured data (text, images).

First, a growing literature examines the rapid diffusion of LLMs and their use. Survey evidence reveals widespread adoption across education levels, occupations, and other demographic groups ([Bick et al. \(2024\)](#)). Usage-based data highlight substantial heterogeneity in how individuals engage with LLMs. Drawing on large-scale ChatGPT conversations, [Chatterji et al. \(2025\)](#) show that users rely on the tool for both work and non-work tasks, with writing assistance, practical guidance, and information-seeking comprising the majority of interactions. While work-related messages have steadily increased, non-work-related usage has grown even more rapidly, rising from 53% to over 70% of total messages. [Padilla et al. \(2025\)](#) find that LLM adoption alters online behavior, leading to declines in traditional search activity and traffic to smaller content sites.

A related literature investigates the use of generative AI at work, focusing on tasks, productivity, and labor market impacts. Survey evidence and internal company data (from Microsoft) point to rapid adoption of LLMs in the workplace ([Hartley et al. \(2025\)](#), [Tomlinson et al. \(2025\)](#)). Experimental studies document sizable productivity gains from LLM

assistance across various professional settings, including management consulting and customer support (Dell’Acqua et al. (2023); Brynjolfsson et al. (2025b)). Complementing these task-level studies, Eloundou et al. (2023) quantify occupational exposure to LLMs, while Handa et al. (2025), using Claude interaction data, find that AI use is concentrated in writing and software-related tasks and reflects a mix of automation and augmentation. At the labor market level, Brynjolfsson et al. (2025a) document reduced hiring for young workers in AI-exposed occupations.

While existing work emphasizes who uses LLMs and for which tasks, this paper shifts focus to which LLMs consumers choose and why. As non-work usage continues to rise (Chatterji et al. (2025)), consumer-facing interactions represent a growing and economically significant share of overall LLM demand.

The closest papers within this literature are Demirer et al. (2026) and Nagle and Yue (2025), which both study the enterprise-facing LLM market using aggregate API usage data from platforms such as OpenRouter and Microsoft Azure. Demirer et al. (2026) provide indirect evidence of both vertical and horizontal differentiation, with no single model dominating across use cases (i.e., programming and marketing) and significant variation in the demand for intelligence across applications. The authors also find that most firms allocate the vast majority of usage to a single model, with limited persistent multi-homing. Nagle and Yue (2025) shows that closed-source models dominate usage on OpenRouter, with users continuing to select them even when open-source alternatives are cheaper and offer superior performance. This paper complements these two works by studying the consumer-facing segment, where usage patterns, substitution dynamics, and provider prominence differ from the firm-level API market. A key advantage of the LMArena setting is the observation of a large-scale sample of conversations across LLMs and firms, enabling measurement of fine-grained LLM attributes (such as formatting, tone, and sentiment) that shape consumer demand.

Second, the paper builds on the IO literature on demand estimation in differentiated-product markets, including seminal frameworks (Berry (1994), Berry et al. (1995), Nevo (2001), Backus et al. (2021); see also recent reviews: Berry and Haile (2021), Gandhi and Nevo (2021)). Within this literature, the paper contributes specifically to the strand using vector embeddings from unstructured data in demand estimation. Studies within this strand have used embeddings obtained from: e.g., product descriptions and pictures (Compiani et al. (2023), Bach et al. (2024), Bajari et al. (2025), Quan and Williams (2021), Lee (2025)), fonts (Han et al. (2024)), survey data on product similarities (Magnolfi et al. (2025)), and consumers’ transactions and search (Armona et al. (2025), Ruiz et al. (2020), Kumar et al. (2020), Gabel and Timoshenko (2022)).

This paper differs from prior work using unstructured data in two ways. First, it incorporates two types of unstructured input per choice (i.e., user prompts and LLM responses) and uses prompt embeddings to capture heterogeneity in user intent and preferences across tasks, by disciplining taste variation in the structural demand model.² Second, to obtain a low-dimensional representation of prompt embeddings, I depart from prior work (e.g., [Compiani et al. \(2023\)](#), [Bach et al. \(2024\)](#)) that relies on Principal Component Analysis (PCA). While PCA performs well in narrow domains, it is poorly suited to LLM prompts, which are isotropic and semantically diverse. Instead, I propose a novel approach based on Partial Least Squares (PLS), a supervised dimensionality-reduction method better able to extract utility-relevant variation in this setting.

2 Industry Background

This section provides background on the industry: how LLMs are trained, how firms differentiate their models, and how model performance is evaluated. It also describes the LMArena platform and some of its limitations.

2.1 LLM Development

Market definition. LLMs are statistical models that generate text responses to text inputs. They are implemented using deep neural networks based on the transformer architecture and operate on sequences of tokens, which represent sub-word units.

This paper focuses on the consumer-facing segment of the LLM market—as distinct from the enterprise segment—where users interact with models via chat interfaces. The analysis centers on foundation models: general-purpose systems trained on broad text corpora and capable of performing diverse language tasks without task-specific re-training. These models are developed by both tech incumbents (e.g., Google, Meta) and new entrants (e.g., OpenAI, Anthropic, DeepSeek). Geographically, the study is restricted to English-speaking consumers and to providers offering public-facing models accessible in North America.

LLM development. Training LLMs proceeds in two stages. In the pre-training phase, models learn to predict the next token in large-scale text corpora drawn from books, websites, and other written sources. Post-training adjusts model outputs to better align with human preferences. This second phase typically involves supervised fine-tuning (SFT) on curated question–answer pairs written or rated by human experts, followed by reinforcement learning from human feedback (RLHF), where models are trained to prefer (and generate) outputs ranked more highly by human evaluators ([Ouyang et al. \(2022\)](#)). In settings where

²For instance, a prompt like “write a concise Python script” reveals a distaste for verbosity, while “provide a detailed and structured plan for a trip to Paris” reflects a preference for organized responses using lists or headings.

task performance can be objectively verified (math, coding), post-training additionally uses reinforcement learning from verifiable rewards (RLVR), in which rewards are computed automatically based on correctness rather than human judgments. Additional tuning may incorporate safety guidelines (Bai et al. (2022)) and application-specific constraints.

Differentiation levers. LLMs are differentiated through a series of design and training decisions made by developers. Pre-training choices—such as model architecture, scale, and the composition of training data—shape core capabilities such as general knowledge and intelligence.

Post-training decisions, particularly the design of SFT and RLHF, further shape model behavior by selecting the type of data and environments used in post-training and how to weight different aspects of model behavior such as helpfulness, safety, and style. For instance, developers can steer models toward concise or elaborate answers, more formal or conversational tone, or stronger preference for cautious versus assertive responses. These attributes emerge from the choice of (post-)training data, prompt formats, human annotator instructions, and reward models used during RLHF and RLVR.³

System prompts, fixed instructions provided by the *user* in a conversation, offer a limited degree of control over LLM response. System prompts can shape model persona or verbosity but exert weaker influence on deeper model attributes. Although some platforms allow users to personalize interactions,⁴ these adjustments operate within constraints set during pre- and post-training. As a result, key dimensions of differentiation—such as core capabilities, tone, and response style—are to a large extent hardwired by firms during training rather than adjusted dynamically by users.

Various company statements accompanying model releases illustrate these points. OpenAI’s GPT-5.1 release note (OpenAI (2025)) state that “GPT-5.1 Instant, ChatGPT’s most used model, is now warmer by default and more conversational,” underscoring that shifts in tone arise from developer-level choices. Similarly, Anthropic (2025) observes that “each Claude model has a unique character, and some users find specific models especially useful or compelling, even when new models are more capable.” Together, these examples suggest that differentiation is embedded in the training process and that consumers perceive models as being differentiated along multiple dimensions (beyond core capabilities and intelligence).

³For instance, OpenAI publishes a “model spec” which provides detailed guidelines about model behavior, personality, and safety. See <https://model-spec.openai.com/2025-12-18.html> [Last Accessed: 2025-12-23]. Similarly, Anthropic also publishes a “Constitution” that guides Claude models’ values and behavior during training. See <https://www.anthropic.com/constitution> [Last Accessed: 2026-01-28].

⁴For example, OpenAI allows users to customize aspects of interaction with GPT, including warmth, enthusiasm, or emoji usage. In the framework developed in this paper, such personalization does not constitute within-model adaptation but instead corresponds to selecting among a discrete set of pre-configured models. From the consumer’s perspective, these variants are treated as distinct models part of the firm’s portfolio.

LLM evaluations. LLM evaluations (or benchmarks) are tests designed to assess and compare model performance across various dimensions such as general knowledge, reasoning ability, or safety. These evaluations shape perceptions of model quality and inform strategic decisions by developers.

Traditional benchmarks rely on static problem sets with predefined correct answers—such as MMLU, HellaSwag, or GSM8K—while other benchmarks draw from evolving sources like Codeforces competitions. In contrast, preference-based benchmarks compare outputs using human judgments, either on static problem sets (e.g., MT-Bench, AlpacaEval) or through live user interactions, as in LMArena.

Static benchmarks based on fixed problem sets face several limitations: they often focus on closed-ended tasks, are vulnerable to data contamination (i.e., problems sets become part of the training data for new models) and overfitting, and struggle to define ground truth for complex, open-ended, or subjective questions. In many real-world settings, users pose open-ended prompts that require reasoning, synthesis, and judgment. For such tasks, human evaluation remains the only reliable approach for assessing helpfulness and quality.

2.2 The LMArena Platform

LMArena. LMArena (arena.ai) is a public platform for human evaluation of LLMs through side-by-side, blind comparisons.⁵ Launched in 2023 by researchers at UC Berkeley’s SkyLab (Chiang et al. (2024)), it was spun out as Arena Intelligence Inc. in 2025 and has since raised \$150 million in funding (at a valuation of \$1.7 billion) from a16z, UC Investments, and Lightspeed.

Users interact with the platform by submitting prompts and voting on paired model responses, in what LMArena terms a battle. After entering a prompt, the user is shown responses from two randomly sampled and anonymized models (model A and B) and has the option to vote for their preferred response. There are four possible battle outcomes: model A wins, model B wins, tie, or both responses are bad. Voting need not occur after one turn: i.e., users can continue the conversation by submitting a follow-on prompt (multi-turn conversation) and cast a vote at any turn after viewing the paired responses. Once a vote is submitted, the model identities are revealed, and the user may start a new conversation with a newly sampled pair of models. By October 2025, the platform had recorded more than 4.2 million votes.

LMArena aggregates these votes into public leaderboards using an Elo rating system, enabling incremental score updates from each user vote. The platform maintains multiple leaderboards, including for text, image generation, agentic coding, vision, and search. The

⁵The company shortened its name to Arena in January 2026.

empirical analysis in this paper focuses on the text battles, though the framework could in principle be extended to model demand for image-generation or search. LMArena evaluates both publicly released LLMs and, in some cases, pre-release models supplied for private testing; the data used in this paper covers only models available to the public. Since its launch, the platform has become a widely cited benchmark in model announcements and promotional materials, shaping perceptions of model quality across the industry (Google (2025), xAI (2025)).⁶

LMArena offers several advantages for analyzing consumer preferences. First, it directly captures human judgments over model outputs, providing revealed-preference data rather than relying on proxy measures such as benchmark scores. This is important because user preferences are not necessarily aligned with benchmark performance: models that score highly on standardized tasks may not be perceived as more useful or engaging in practice. Second, because different models respond to identical prompts, the data permit credible inference on substitution patterns across models: i.e., there is no selection into use of a particular LLM. Finally, the availability of full conversation transcripts enables the analysis of rich heterogeneity in preferences across both users and prompt types.

Limitations of the LMArena data. A first limitation of the LMArena platform is sample representativeness. The platform attracts a specialized group of users, including researchers, engineers, and technically engaged early adopters, whose preferences may differ from those of the general population. For a given LLM response, these users may be more attuned to subtle differences in quality or formatting, which may bias preference coefficients away from zero. This can lead to estimated substitution patterns from changes in LLM capabilities that are upper bounds. Nonetheless, this population of users represents a commercially relevant segment whose evaluations feature prominently in firms’ model announcements, and whose preferences likely shape model development.

A second concern is that LMArena users may submit a different mix of prompts and use cases than the general population. This issue is less consequential, as the empirical approach controls for prompt content and conditions on use case. Moreover, when conducting counterfactual exercises, I calibrate the prompt distribution to match that of the general ChatGPT user base using data from Chatterji et al. (2025) (Section 5.2).

A third set of limitations arises from the structure of the LMArena choice environment.

⁶Singh et al. (2025) raises some structural issues in the LMArena leaderboard, including data access disparities that may favor certain large providers. While the machine learning community often treats LMArena rankings as reflecting intrinsic model quality, I interpret the data as measuring revealed human preferences over LLMs. These preferences are informative but may not fully align with objective notions of model capabilities. Indeed, the purpose of this study is to identify human preferences over a range of LLM attributes, which are not limited to core capabilities.

Users make binary comparisons in a frictionless setting, with zero prices, no switching costs or platform-level complementarities (e.g., integration of Gemini into Google search and Gmail).

The absence of prices restricts the scope of elasticities that can be estimated. Nevertheless, widespread use of free tiers supports the relevance of LMArena’s zero-price comparisons.⁷ For paying users, subscriptions typically provide access to a portfolio of models with usage limits varying across subscription tiers. Conditional on accessing a set of LLMs via subscriptions, LMArena is informative about which model is preferred for a given task. The data, however, do not identify willingness to pay for access, selection into subscription tiers, or responses to changes in subscription prices. As a consequence, the counterfactual exercise focuses on substitution driven by changes in model intelligence, conditional on access to the same set of models.

A separate limitation concerns switching costs. The framework abstracts from persistence in provider choice due to accumulated context, personalization, or investments in provider-specific workflows. The counterfactual analysis partly addresses the implications for substitution by considering two homing scenarios: users either commit to a single provider across all queries or choose a model and provider separately for each query.⁸ These scenarios illustrate how restrictions on provider switching affect substitution. However, both specifications are static, and the available data do not identify switching costs or the speed of adjustment following changes in product characteristics.

A final concern is that votes may be manipulated. Because LMArena rankings are widely cited in model announcements, a provider has an incentive to cast votes in favor of its own models or against those of competitors. Such an attack requires two steps: identifying which of the two anonymous responses was generated by the target model (for instance, from a classifier trained on the model’s outputs, or prompts that induce the model to disclose its identity) and then voting for that model irrespective of response quality. Bots and coordinated voting make this strategy scalable. Recent work in computer science (Huang et al. (2025), Min et al. (2025), Cho and Kim (2026)) show that such attacks are, in theory, feasible: anonymous responses can be attributed to their source model with high accuracy, and rigged votes can move a model’s ranking absent defenses.

LMArena has implemented several defenses against such attacks, some of them in collaboration with Huang et al. (2025). First, the platform filters out conversations in which the user attempts to extract the identity of the model.⁹ Second, the platform detects anomalous users through a statistical test that compares each user’s votes to the historical distribution

⁷Financial Times (2025) reports that around 5% of ChatGPT users are on the subscription tier.

⁸Brynjolfsson et al. (2026) report that US consumers of generative AI use an average of 2.6 distinct providers, with a median of two, suggesting that multi-homing is common.

⁹See <https://arena.ai/faq> [Last Accessed: 2026-08-14].

of outcomes for the same model pair (Chiang et al. (2024)); votes by users exhibiting statistically anomalous voting patterns are removed from the leaderboard computation. Finally, bot protection (Cloudflare, reCAPTCHA), login requirements, and rate limits raise the cost of automated voting; Huang et al. (2025) conclude that these mitigations substantially increase the cost of manipulation. Taken together, these defenses arguably mitigate the concern that vote manipulation contaminates the battle data used in this paper.

The data identify preferences over response and model attributes for this population of users. The counterfactual analysis uses these preferences to predict substitution under assumptions about users’ distribution of tasks and the extent of multi-homing. These predictions are conditional on access to a given set of models and abstract from subscription decisions, brand effects, ecosystem complementarities, and switching costs. The estimated preferences thus provide necessary inputs to richer models of market demand that can incorporate these additional frictions.

3 Data and Descriptive Statistics

3.1 Data sources

This paper draws on two primary data sources: LMArena, the platform described in the previous section, and Artificial Analysis, an independent aggregator of LLM benchmarks and capabilities.

LMArena. The main dataset consists of battles from LMArena, covering two time windows: June 2024–August 2024 and April 2025–July 2025.¹⁰ Each observation includes an (anonymized) user identifier, a battle identifier, the two model identities sampled in the battle, the user’s vote, and the full conversation text (user prompts and both model responses in all turns of the conversation). The analysis focuses only on battles in English.

I classify battles into topic categories, following the same classification used by OpenAI’s internal conversation classifier and reported in Chatterji et al. (2025) (Section 5.2). The topic categories are: information seeking, practical guidance, writing, technical help (math, coding), casual conversation (referred to as “Self-Expression” in Chatterji et al. (2025)), multimedia, and other. Topic definitions and examples are provided in Appendix Table A3.

From the conversation data, I extract three sets of LLM response-level characteristics. The first are formatting-related variables. These include the number of Markdown headers (section headings); the number of bold text blocks; and the number of ordered and unordered list items. I also record whether the response includes a table or emojis, and measure the

¹⁰These datasets are `arena-human-preference-140k` and `arena-human-preference-100k`, they are publicly available at: <https://huggingface.co/lmarena-ai> [Last Accessed: 2025-12-05].

total length of the response in tokens.

The second group of variables pertains to readability and syntactic complexity. I compute two standard readability metrics: the Flesch Reading Ease score and the SMOG Index. The FRE score is based on sentence length and the number of syllables per word, with higher values indicating more accessible text. The SMOG Index estimates the years of education required to understand a text, and is based on the frequency of polysyllabic (three or more syllables) words within a fixed number of sentences. While both metrics aim to quantify readability, they emphasize different linguistic features: the Flesch score is more sensitive to sentence and word length, whereas the SMOG Index emphasizes lexical complexity.

The third group of variables captures model tone. While many tonal features could be analyzed, I focus on two dimensions: sentiment and hedging. Sentiment is classified into three categories—positive, neutral, and negative—based on the emotional valence and intensity of the LLM’s response. Positive sentiment includes favorable expressions (e.g., “insightful question,” “incredibly helpful”), while neutral sentiment reflects objective, factual, or informational content without evaluative tone. Negative sentiment reflects unfavorable, critical, or emotionally intense response (e.g., “this is a terrible idea”).

Hedging refers to the degree of linguistic caution or tentativeness in a response. I classify tone as assertive when the LLM uses confident, directive language (e.g., “You should do X,” “This is the best approach”), tentative when the response includes hedging markers such as modal verbs or qualifiers (e.g., “This might help,” “You could try...”, “I don’t know”), and neutral when the tone is descriptive without conveying strong confidence or hesitation. Hedging focuses specifically on the LLM’s epistemic commitment: how strongly the model asserts or qualifies its claims.

The classification of conversation topics and response-level sentiment and hedging are performed using an LLM as judge. Appendix A.1 provides the full prompt and implementation details used for each classification tasks.

Finally, I obtain vector embedding representations for each user prompt. An embedding is a numerical representation of text that encodes semantic meaning by mapping prompts into a continuous vector space, where prompts with similar meanings are located closer together. I compute embeddings using OpenAI’s `text-embedding-3-large` model, a state-of-the-art embedding model designed for high-quality semantic representation. The embedding model produces, for each user prompt,¹¹ a 3,072-dimensional embedding vector, which I use to control for heterogeneity in preferences across conversations.

In addition to the battles data, I collect biweekly snapshots of the LMArena leaderboard,

¹¹In multi-turn conversations where the user submits follow-on prompts, I concatenate the prompts before recovering the embedding.

which reports the list of models being tested, their providers, and licensing status (open-source, open-weights, proprietary), from June 2023 to July 2025.

Artificial Analysis. To measure model core capabilities, the paper draws on public data from Artificial Analysis, a market research firm that provides standardized benchmarking and related model information. The dataset covers all publicly released LLMs as of July 2025 and includes, for each LLM, its performance scores on a wide range of benchmarks, along with model attributes such as release date, context window size, and whether the model is a reasoning model.¹²

A key advantage of this data source is its consistent evaluation framework. LLM providers often report benchmark performance for new models under differing conditions—such as varying numbers of demonstrations (n -shot prompting)—which undermines comparability. Artificial Analysis addresses this issue by applying uniform testing protocols across all models, enabling credible comparisons of benchmark scores. Appendix Table A4 provides definitions of the benchmarks used. In the remainder of the paper, I use 4 indexes formed by averaging benchmark scores: the “Intelligence index” is a weighted average of all 7 scores (where weights are given by the last column in Table A4), the “Coding index” is an average of SciCode and LiveCodeBench, the “Math index” is an average of MATH-500 and AIME, finally the “Knowledge index” is an average of MMLU-Pro, HLE, and GPQA Diamond.

3.2 Descriptive statistics

Figure 1 shows the cumulative number of LLMs tested on LMArena over time, with the periods for which conversation data are available shaded. By July 2025, 411 LLM from 65 providers have been tested. Providers are major technology firms and commercial AI labs as well as smaller startups and university research groups. Proprietary models are supplied primarily by firms such as Google, OpenAI, Anthropic; open-weights models by firms such as Meta, Google, and Cohere; and open-source models by firms such as Alibaba, Microsoft, Mistral, and several academic groups.¹³ Together, these data capture the full range of LLM development activity across licensing regimes and provider types.

Table 1 shows descriptive statistics of LMArena battles. The dataset contains 237,503 battles, of which 53% are in English, and have on average 1.31 turns per conversation. For

¹²The context window size is the maximum number of tokens a model can process in a single prompt–response interaction. A reasoning model is one explicitly optimized, typically through SFT and RL, for multi-step reasoning and problem-solving tasks. These models generate “reasoning” tokens before outputting a response and are therefore slower than non-reasoning models. This paper uses Artificial Analysis’s v2.0 suite of benchmarks.

¹³For open-weights models, model weights (parameters) are released for use subject to a license, whereas open-source models make both the weights and the full training and implementation code available under an open-source license.

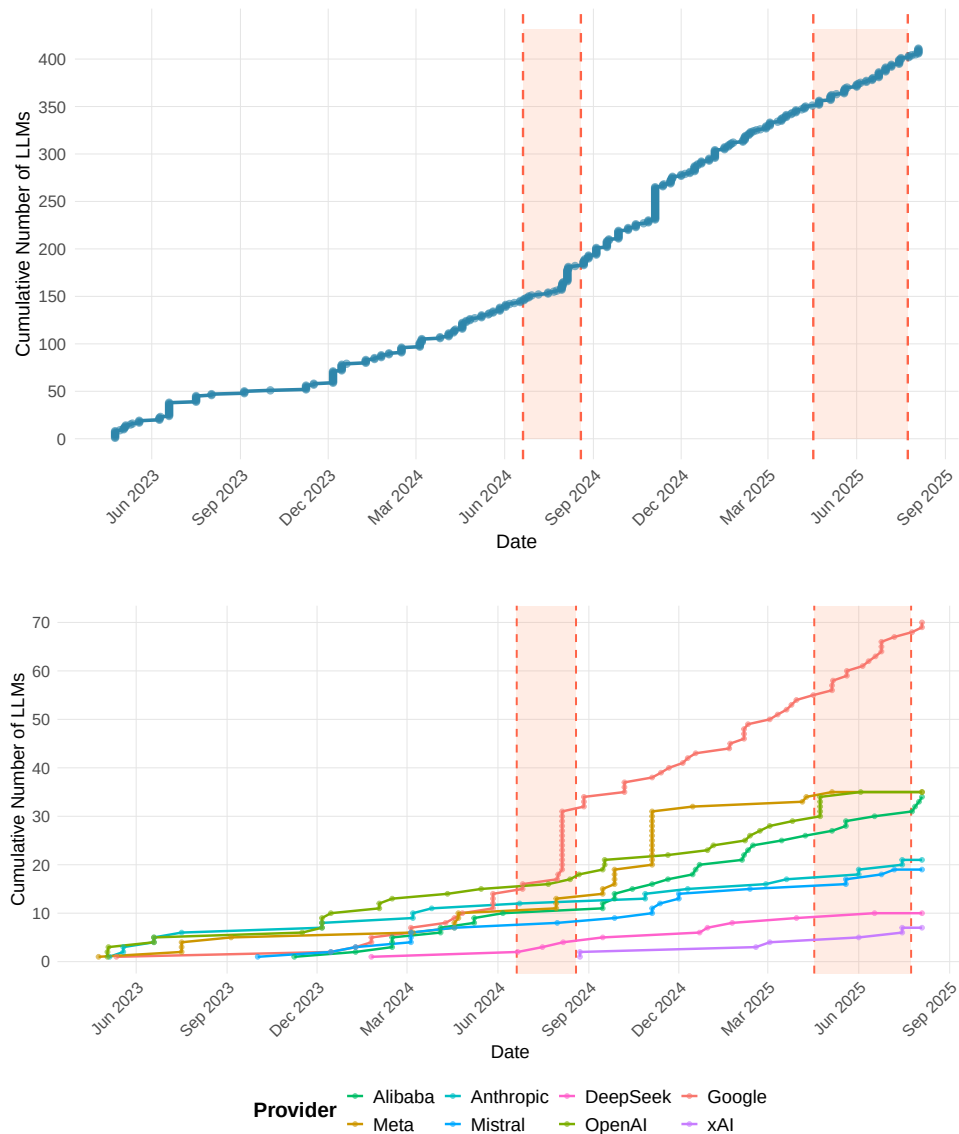


Figure 1: Cumulative number of LLMs on LMArena (top shows total, bottom shows the top 8 providers)

battles in English, users vote for one of the two model outputs in 68% of battles, with 16.9% ties and 14.6% rated both bad. The distribution over conversation topics is shown in Panel C: the top 3 topics are information seeking, writing, and practical guidance. This distribution is broadly in line with the most popular topics reported by [Chatterji et al. \(2025\)](#) for a large representative sample of ChatGPT users. However, as expected, math and coding prompts are more frequent on LMArena (21%) compared to the overall population of ChatGPT users, where such prompts drop from 12% to 5.1% of conversations between 2024 and 2025.

Table 1: Descriptive Statistics: Battles

	Jun 2024 – Aug 2024	Apr 2025 – Jul 2025	All
Panel A: Basic Statistics			
Number of battles	101,869	135,634	237,503
Number of users/sessions	48,471	115,372	163,843
Number of LLMs	55	52	103
Number of providers	17	13	22
Number of battles in English	56,061	71,175	127,236
Share of battles in English	0.550	0.525	0.536
Avg. number of turns per conversation	1.38	1.26	1.31
Panel B: Share of Battle Outcomes (English Battles Only)			
Model A wins	0.311	0.359	0.338
Model B wins	0.322	0.366	0.347
Tie	0.180	0.160	0.169
Both bad	0.187	0.114	0.146
Panel C: Share of Battles by Topic Category (English Battles Only)			
Information Seeking	0.338	0.406	0.372
Writing	0.230	0.140	0.185
Practical Guidance	0.101	0.190	0.147
Coding	0.137	0.130	0.133
Math	0.109	0.042	0.075
Other/Multimedia	0.041	0.053	0.047
Casual Conversation	0.044	0.039	0.041

Note: For the 2025 dataset, users are identified only per session but cannot be tracked across sessions.

Table 2 shows the distribution of LLM capabilities in the Artificial Analysis dataset. Benchmark indices span a wide range: the intelligence index ranges from 8.3 to 73.2, coding from 0.1 to 63.8, and math from 2.9 to 96.7. The context window sizes vary from 2,000 to 10 million tokens, and about 28% of models are classified as reasoning models. Appendix Figure A2 plots the highest intelligence index across LLMs for each of the top 8 providers: OpenAI’s models scored higher on benchmark metrics through mid-2024, but providers such as Google, xAI, and others rapidly closed the gap by mid-2025.

Finally, Table 3 provides descriptive statistics for LLM responses in the LMArena dataset. Panel A shows modest readability variation, with an average Flesch Reading Ease score of 46.6 and SMOG index of 12.7. Panel B highlights increased use of formatting elements over time: e.g., the number of text blocks in bold increases from 6.23 to 17.95; the number of bullet points increases from 5 to 14.5, and responses are twice as long in 2025 than in 2024. Panel C shows that response sentiment is predominantly neutral, but the share of positive sentiment responses increases from 12% to 18% of all responses between 2024 and 2025. Assertive responses are also more frequent, whereas neutral and tentative responses are less frequent in 2025 relative to 2024.

Table 2: Descriptive Statistics: LLM Characteristics

Variable	Mean	SD	Min	Median	Max
Panel A: Benchmark Scores					
MMLU Pro	0.63	0.17	0.14	0.68	0.87
GPQA	0.48	0.17	0.10	0.47	0.88
HLE	0.06	0.03	0.03	0.05	0.24
LiveCodeBench	0.30	0.20	0.002	0.27	0.82
SciCode	0.23	0.11	0.00	0.24	0.47
Math 500	0.72	0.23	0.06	0.77	0.99
AIME	0.28	0.29	0.00	0.15	0.94
Intelligence Index (0-100)	37.83	15.01	8.30	36.65	73.20
Coding Index (0-100)	26.51	14.71	0.10	25.35	63.80
Math Index (0-100)	50.75	24.48	2.90	45.80	96.70
Panel B:					
Context window ('000s tokens)	356.27	840.35	2	128	10,000
Reasoning Model	0.28	0.45	0	0	1

Note: The unit of observation is an LLM. Benchmark scores are between 0 and 1. Benchmark indices are between 0 and 100.

To illustrate potential differences in responses across LLMs, I regress the response-level characteristics in Table 3 on calendar quarter, conversation topic category, and LLM fixed effects. Given the market definition used in this paper, I include fixed effects for LLMs developed by private firms that offer consumer-facing products (which excludes enterprise-focused providers such as Cohere) and make their product available in the North American consumer market (which excludes some Chinese providers such as Alibaba). The firms are: Anthropic, DeepSeek, Google, Meta, Mistral, OpenAI, and xAI. LLMs by providers outside this list are labelled as “other” and serve as the omitted reference category.¹⁴

Figure 2 plots LLM fixed effects for two formatting-related outcomes: response length and table usage. To facilitate visual comparison, the figure shows up to four representative LLMs per provider. Appendix Figures A3 and A4 provide the full set of LLMs. More recent models tend to produce longer responses on average, with notable variation across providers. xAI and Google models yield the lengthiest outputs, followed by OpenAI and DeepSeek. By contrast, Anthropic models are consistently associated with shorter responses. Table usage is even more heterogeneous. OpenAI’s high-capability models (o3 and o4-mini) stand out, including tables in roughly 30-40% of their responses—substantially higher than Google’s Gemini 2.5 Pro (24%) and significantly above models from Anthropic, which rarely format content using tables.

¹⁴In practice, I define two sets of “other” LLMs: those released up to end of the first window (June 2024-August 2024), and those released after that date.

Table 3: Descriptive Statistics: LLM Responses

	Jun 2024 – Aug 2024		Apr 2025 – Jul 2025		All	
	Mean	(SD)	Mean	(SD)	Mean	(SD)
Panel A: Readability						
Flesch Reading Ease	49.17	(94.03)	44.16	(50.52)	46.59	(74.91)
SMOG Index (years)	12.38	(4.39)	12.96	(4.08)	12.68	(4.24)
Panel B: Formatting						
Markdown headers	0.72	(4.03)	3.39	(8.98)	2.09	(7.16)
Bold text	6.23	(15.27)	17.95	(31.99)	12.26	(25.96)
Ordered lists	3.52	(8.25)	4.62	(10.29)	4.09	(9.37)
Unordered lists	5.00	(12.62)	14.50	(27.38)	9.89	(22.04)
Response length (tokens)	562.74	(906.52)	994.63	(1392.98)	785.04	(1201.72)
Has table	0.02	(0.14)	0.10	(0.30)	0.06	(0.24)
Has emoji	0.02	(0.13)	0.15	(0.36)	0.09	(0.28)
Panel C: Tone						
Sentiment: positive	0.12	(0.32)	0.18	(0.38)	0.15	(0.36)
Sentiment: negative	0.02	(0.14)	0.01	(0.10)	0.02	(0.12)
Sentiment: neutral	0.86	(0.34)	0.81	(0.39)	0.84	(0.37)
Hedging: assertive	0.42	(0.49)	0.48	(0.50)	0.46	(0.50)
Hedging: tentative	0.16	(0.37)	0.14	(0.34)	0.15	(0.36)
Hedging: neutral	0.42	(0.49)	0.38	(0.48)	0.40	(0.49)

Note: Descriptive statistics for response-level features from the LMArena dataset. Flesch Reading Ease: Higher scores indicate easier-to-read text. Components: words per sentence and syllables per word. SMOG Index: Estimates years of education needed to understand the text. Components: count of polysyllabic (≥ 3 syllables) words. Higher implies harder to read. Bold text corresponds to the number of blocks of text in bold.

Figure 3 shifts focus to readability and tone, using the SMOG index and a measure of positive sentiment. Again, the figure shows up to four representative LLMs per provider. Appendix Figures A5 and A6 provide the full set of LLMs. The SMOG index, which approximates lexical complexity, shows diverging trends across firms. Anthropic’s newer models score higher, indicating more complex word choices, while OpenAI’s models display the opposite pattern, becoming easier to read. This may reflect differences in product positioning or targeted user segments. Regarding sentiment, OpenAI and Google exhibit an upward trend in positive framing, with newer models increasingly producing upbeat language (e.g., “That’s a great question!”). Anthropic’s models, by contrast, show little movement on this dimension.

These patterns are not restricted to the four outcomes presented in these figures: sim-

ilar degrees of heterogeneity can be found across other formatting, readability, and tone attributes. Overall, these patterns illustrate meaningful LLM-level differences in response attributes with variation occurring both across firms and across LLMs for a given firm.

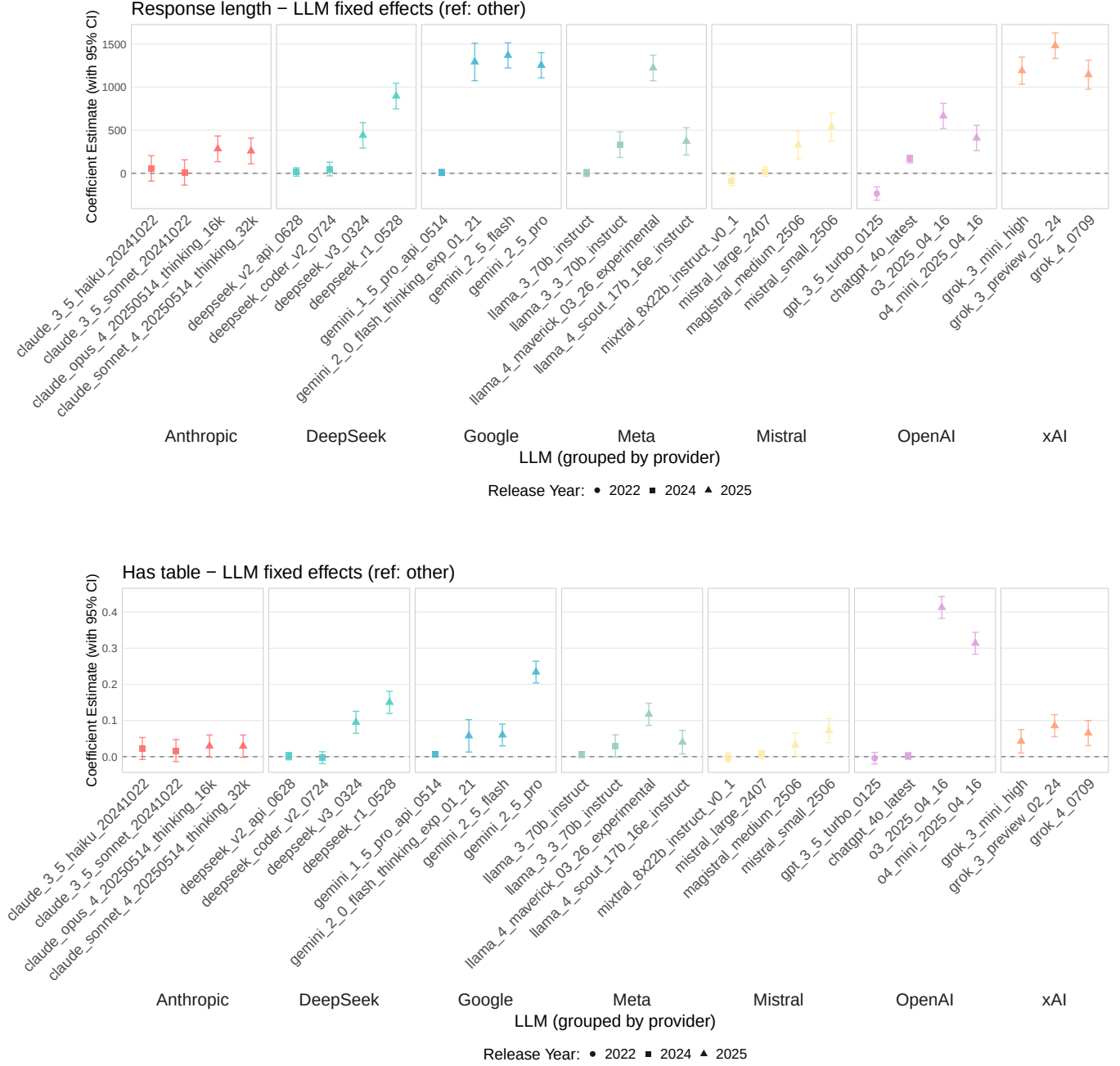


Figure 2: Formatting characteristics by LLM: Response length (tokens) and use of tables

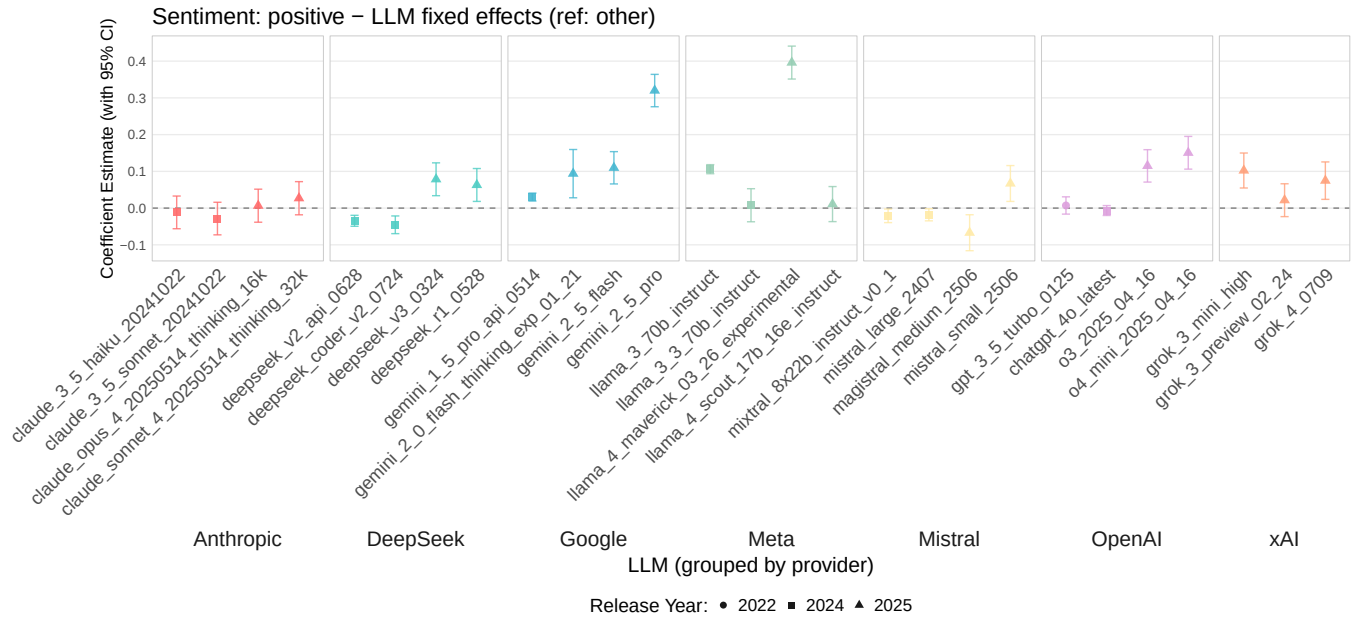
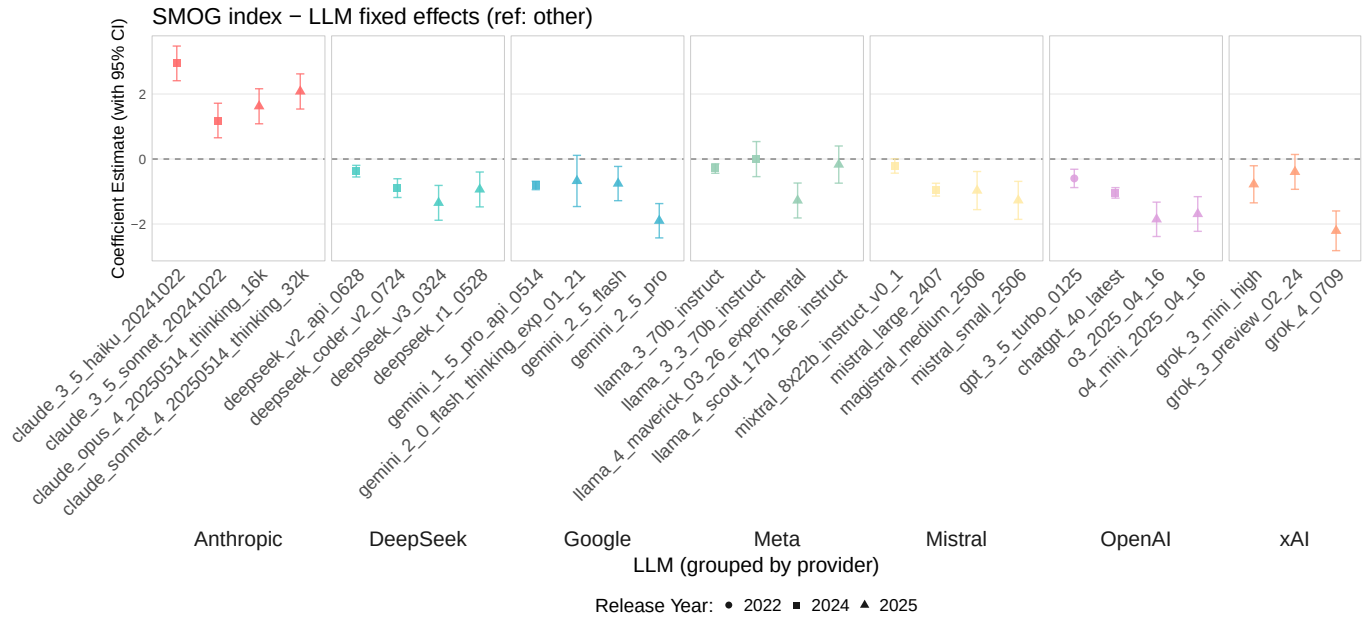


Figure 3: Readability and tone by LLM: SMOG index and positive sentiment

4 A Choice Model for LMArena Battles

This section develops a random-coefficient discrete choice model tailored to the institutional structure of LMArena battles. In each interaction, a user submits a prompt and evaluates two competing LLM responses. The model maps these observed choices into latent utilities over responses, decomposing utility into response-specific attributes and model-level capabilities. By allowing preferences to vary systematically with prompts and across users, the framework provides a disciplined way to recover latent LLM quality and to quantify how different dimensions of response attributes shape user valuations.

Setup. A user i submits a text prompt p to the LMArena platform. Each prompt p is mapped into a d -dimensional embedding vector $\mathbf{E}_p \in \mathcal{P}$, where $\mathcal{P} \subset \mathbb{R}^d$. For each user i , there exists a user-specific probability measure F_i on $\mathcal{P}$ describing the distribution of that user’s prompt embeddings, and $\mathbf{E}_p$ represents a random draw from the distribution F_i . Two LLMs m_A and m_B generate responses. The resulting “battle” is a tuple

$$(i, p, \{m_A, m_B\}, o),$$

where the outcome $o \in \{m_A, m_B, \emptyset\}$ (model A wins, model B wins, or both bad) is determined by

$$o = \begin{cases} m_A & \text{if } U_{ipm_A} > U_{ipm_B} \text{ and } U_{ipm_A} > U_{ip0}, \\ m_B & \text{if } U_{ipm_B} > U_{ipm_A} \text{ and } U_{ipm_B} > U_{ip0}, \\ \emptyset & \text{if } U_{ipm_A} < U_{ip0} \text{ and } U_{ipm_B} < U_{ip0}, \end{cases}$$

with U_{ipm} denoting user i ’s utility for LLM m ’s response to prompt p and U_{ip0} the outside-option utility (an acceptability threshold, corresponding to the vote “both bad”).¹⁵ The baseline framework ignores ties, Appendix B shows how to accommodate ties.

Utility specification. The utility a user derives from a response has two distinct components: (i) a component that varies at the *response level*, reflecting observable features of the particular output generated in response to prompt p , and (ii) a component that varies at the *LLM level*, reflecting core capabilities of model m that shape the semantic quality of all its responses. Response-level characteristics X_{pm} capture measured attributes of the specific output—formatting, readability, and tone. In contrast, semantic attributes such as correctness, relevance, and completeness are modeled as functions of the capabilities of the LLM, summarized by the model-level characteristics X_m (which include the LLM’s

¹⁵Operationally, the “both bad” outcome corresponds to the event that neither model’s output exceeds this intrinsic threshold. The threshold may reflect the quality the user expects from a general search engine (Google) or a minimal notion of usefulness, completeness, or correctness.

benchmark scores, context window size, reasoning ability, and a constant). User i 's utility from model m 's response to prompt p is specified as

$$U_{ipm} = \beta_i^R(p)'X_{pm} + \beta_i^L(p)'X_m + \xi_m + \varepsilon_{ipm}, \quad (1)$$

where ξ_m captures unobserved LLM quality, and ε_{ipm} is an idiosyncratic shock distributed according to a type-1 extreme value distribution. Taste parameters $(\beta_i^R(p), \beta_i^L(p))$ vary across *users* and *prompts* through random coefficients, with a mean that is a function of the user prompt embedding $\mathbf{E}_p$, and a variance-covariance matrix to be estimated, as follows:

$$\begin{pmatrix} \beta_i^R(p) \\ \beta_i^L(p) \end{pmatrix} = \begin{pmatrix} \beta_0^R + \beta_1^{R'}h(\mathbf{E}_p) \\ \beta_0^L + \beta_1^{L'}h(\mathbf{E}_p) \end{pmatrix} + \Sigma\nu_i, \quad \nu_i \sim \mathcal{N}(0, I_K), \quad (2)$$

where K is the dimension of the observed characteristics vector $[X_{pm} \ X_m]$, $h(\cdot)$ is a function mapping $\mathbb{R}^d$ into $\mathbb{R}^g$ (with $g < d$) that extracts a low-dimensional representation of the prompt embedding, Σ is a diagonal scaling matrix with element σ_k representing the dispersion in user-specific taste shock for characteristic k . (β_0^R, β_0^L) is a $K \times 1$ vector, (β_1^R, β_1^L) is a $K \times g$ matrix. Each characteristic k in $[X_{pm} \ X_m]$ is associated with a mean coefficient $\beta_{0,k}^R$ and a set of prompt-level interactions $\beta_{1,k}^R$, which capture the dependence of taste parameters on the prompt p . This specification ensures that prompts that are closer in the embedding space will have similar mean taste parameters.

To control for the unobserved quality term ξ_m , it is useful to define the following LLM-specific fixed effect parameter

$$\delta_m = \beta_0^{L'}X_m + \xi_m \quad (3)$$

The term δ_m represents the average quality of model m that is common across consumers. In the baseline model, the mean of the outside option is normalized to zero: $U_{ip0} = \varepsilon_{ip0}$. Appendix C considers a time-varying outside option (i.e., to control for shifts in user expectations over time).

Choice Probabilities. Conditional on the random taste shock ν_i , the probability that user i votes for model m_A 's response is

$$P(o = m_A \mid p, \{m_A, m_B\}, \nu_i) = \frac{\exp(\beta_i^R(p)'X_{pm_A} + \beta_i^L(p)'X_{m_A} + \xi_{m_A})}{1 + \sum_{k \in \{m_A, m_B\}} \exp(\beta_i^R(p)'X_{pm_k} + \beta_i^L(p)'X_{m_k} + \xi_{m_k})}. \quad (4)$$

The unconditional probability integrates over the distribution of user heterogeneity:

$$P(o = m_A \mid p, \{m_A, m_B\}) = \int P(o = m_A \mid p, \{m_A, m_B\}, \nu_i) d\Phi(\nu_i). \quad (5)$$

where Φ is the cdf of the standard multivariate normal distribution. The choice probabilities for the other outcomes can be similarly defined. In anticipation of the counterfactual analysis, note that a user i is characterized by (i) a distribution over prompts $\mathbf{E}_p \sim F_i$ and (ii) a taste vector ν_i drawn from the population distribution.

The framework is general enough to allow the distribution of prompts to depend both on consumer tastes and on the capabilities of the models available at a given point in time. Dependence on the consumer taste shock, $F_i = F(\cdot \mid \nu_i)$, would capture, for instance, the fact that consumers who use LLMs mostly for coding tasks may value coding performance more than the average user. More generally, as models become more capable, users may submit tasks that they previously would not have delegated to an LLM, changing the distribution of prompts over time. This channel can be incorporated by specifying $\mathbf{E}_p \sim F_{it}(\cdot \mid \nu_i, S_t)$, where $S_t = (\mathcal{J}_t, \{X_m\}_{m \in \mathcal{J}_t})$ denotes the choice set of available models and their capabilities.¹⁶

5 Estimation

5.1 Identification

The outcome probabilities (Equation (5)) conditional on the prompt and model pair are identified nonparametrically. These win rates constitute the fundamental choice probabilities that the structural model must rationalize.

Persistent preference heterogeneity across users enter utility through a random-coefficients structure. The diagonal matrix Σ governs the variance of these taste shocks across dimensions of response-level and model-level attributes. Identification of Σ relies on the fact that heterogeneous preferences generate systematic violations of the independence of irrelevant alternatives (IIA) that cannot arise under homogeneous tastes. Appendix E provides a concrete example of how consumer heterogeneity is identified from violations of IIA substitution patterns.

Prompt embeddings $\mathbf{E}_p$ vary exogenously across battles and enter the taste parameter coefficients through the reduced-dimensional representation $h(\mathbf{E}_p)$. Identification of prompt-level heterogeneity in taste parameters (the terms: $\beta_1^{R'}h(\mathbf{E}_p)$ and $\beta_1^{L'}h(\mathbf{E}_p)$) exploits systematic variation in battle outcomes across prompts. For example, the ability to produce structured output (table, bullet point lists) may matter more for prompts related to information-

¹⁶Changes in F_{it} may have important implications for competition. A capability improvement can increase the private return to innovation not only by improving responses to existing tasks, but also by unlocking tasks for which the innovating model has an advantage. For welfare, capability improvements can create value both by improving responses to tasks users already submit and by expanding the set of tasks that users can productively delegate to LLMs. Estimating the causal dependence of F_{it} on S_t is not straightforward and would require a source of exogenous variation that shifts model capabilities independently of other factors affecting the distribution of prompts.

seeking or practical guidance than for creative writing prompts. Since prompts are assigned randomly to model pairs, this exogenous variation traces out the importance of a particular characteristic in explaining user choice, and identifies the parameters β_1^R and β_1^L .

5.2 Estimation Approach

The goal of estimation is to recover the parameters governing response-level preferences, model-level preferences, and the distribution of random taste heterogeneity, $(\beta_0^R, \beta_1^R, \beta_0^L, \beta_1^L, \Sigma)$, as well as the LLM-specific mean utilities $\{\delta_m\}$, defined in Equation (3). Estimation proceeds in two steps.

Step 1: Simulated Maximum Likelihood. For each user, the likelihood contribution integrates over the individual-specific random coefficients. Let b index battles faced by user i , and let ν_i denote the vector of random taste shocks entering through the random-coefficients specification. The exact (log-)likelihood contribution for user i is

$$LL_i = \ln \left(\int \prod_b P(o \mid b; \nu_i) d\Phi(\nu_i) \right),$$

where $P(o \mid b; \nu_i)$ is the model-implied probability of the observed outcome in battle b given ν_i . Because the integral does not admit a closed form, it is approximated by simulation.¹⁷ Drawing $\{\nu_i^r\}_{r=1}^R$ independently from the multivariate normal distribution, the simulated log-likelihood for user i is

$$SLL_i = \ln \left(\frac{1}{R} \sum_{r=1}^R \prod_b P(o \mid b; \nu_i^r) \right). \quad (6)$$

Summing over all users yields the simulated log-likelihood function. Maximizing this function produces estimates $(\hat{\beta}_0^R, \hat{\beta}_1^R, \hat{\beta}_0^L, \hat{\beta}_1^L, \hat{\Sigma}, \{\hat{\delta}_m\}_m)$. The $\hat{\delta}_m$ terms can be interpreted as model fixed effects capturing average performance after accounting for response-level attributes and heterogeneity in user tastes.

Step 2: Generalized Least Squares. The second step estimates β_0^L by exploiting the relationship

$$\delta_m = \beta_0^{L'} X_m + \xi_m.$$

Given estimates $\hat{\delta}_m$ from Step 1 and their estimated covariance matrix V_δ , the coefficient vector β_0^L is obtained by generalized least squares:

$$\hat{\beta}_0^L = (X_m' V_\delta^{-1} X_m)^{-1} X_m' V_\delta^{-1} \hat{\delta}. \quad (7)$$

¹⁷The simulated LL uses 250 Sobol sequence draws. Robustness checks are conducted in Appendix C.

This step follows the logic of classical two-step estimators for random-coefficients discrete-choice models, where the first step includes products fixed effects, as in [Nevo \(2001\)](#). The first step recovers consistent estimates of model-specific mean utilities (which control for the unobserved term ξ_m), whereas the second step relates these utilities to model characteristics in a linear specification, improving efficiency by accounting for correlation across the $\hat{\delta}_m$.¹⁸

Prompt heterogeneity. The attributes that make an LLM output more preferred depend on the task described by the prompt. For example, model intelligence may be especially valuable for coding and math related tasks, whereas the value of tone, verbosity, or formatting may differ between writing and information-seeking tasks. Imposing a common mean taste parameter across prompts would mask systematic differences across use cases and could cause prompt-level preference heterogeneity to be misattributed to persistent heterogeneity across users. I therefore allow the means of the random taste parameters for both model- and response-level characteristics (the terms $\beta_0^R + \beta_1^{R'}h(\mathbf{E}_p)$ and $\beta_0^L + \beta_1^{L'}h(\mathbf{E}_p)$ in Equation (2)) to vary with the prompt. I implement two alternative specifications for the function $h(\cdot)$.

I first use a simple specification, where $h(\mathbf{E}_p)$ corresponds to the topic of prompt p , where topic categories are described in Section 3. I interact every model- and response-level characteristic with topic indicators. The resulting coefficients recover, for example, separate mean valuations of intelligence, response length, and tone for writing and coding prompts. These topic-specific means supplement the random coefficients, which capture taste heterogeneity across users within a given topic.

As a second approach, I use the continuous information contained in prompt embeddings. Because each embedding contains 3,072 dimensions, including the full vector $\mathbf{E}_p$ in the taste parameters is infeasible; I therefore reduce their dimension using Partial Least Squares (PLS) ([Wold \(1966\)](#), [Wold \(1985\)](#)), a supervised counterpart to PCA. PCA, the standard approach in the recent literature using unstructured data in demand estimation (e.g., [Compiani et al. \(2023\)](#), [Bach et al. \(2024\)](#)), selects directions that explain the most variation in the embeddings. This is potentially limiting here because LLM prompts are semantically dispersed and the directions along which prompts vary most need not be those that drive substitution across models. PLS instead selects directions of the prompt embeddings that *covary* most strongly with the model and response attributes that distinguish the chosen alternative from the rejected alternative in each battle. The resulting PLS scores therefore retain the semantic dimensions of prompts that are most informative about variation in tastes. In this specification, I interact every characteristic with the first three PLS scores. Appendix F

¹⁸Standard errors of the second-step estimates are obtained from the covariance matrix $(X_m' V_\delta^{-1} X_m)^{-1}$, which accounts for the estimation error in the first-stage LLM FEs.

provides the technical details. In Appendix C, I discuss robustness checks with respect to the choice of PLS scores.

The two approaches have complementary strengths. Topic interactions are easy to interpret and directly yield a mean taste parameter for each characteristic and use case. However, topic categories are coarser than the continuous prompt representation: they capture heterogeneity between topics but not variation among prompts within the same topic. PLS can capture both sources of variation; indeed, the results below reveal substantial within-topic heterogeneity in tastes. The PLS approach is also more parsimonious because the function $h(\mathbf{E}_p)$ is lower-dimensional (there are 6 included topics vs 3 PLS scores in the preferred specification). This flexibility comes at the cost of less directly interpretable components and additional first-stage uncertainty introduced by the data-driven dimensionality-reduction step.

5.3 Estimation results

This section presents the results of the two-step estimation approach. To facilitate optimization, all continuous response-level variables (i.e., response length, SMOG index) are normalized to have mean zero and unit variance. Categorical features, including sentiment, hedging, reasoning, response has table, and response has emojis, are coded as indicator variables. Benchmark scores are rescaled to lie between 0 and 10, so that a one-unit change corresponds to a 10-point increase on the original 0–100 benchmark-index scale, equivalent to one-tenth of the maximum possible score. For reference, the average intelligence index in the data is 37.83 on the original scale (3.78 after rescaling; see Table 2). The context window size is similarly rescaled so that a one-unit change corresponds to an increase of one million tokens. Finally, a dummy variable is included for whether the LLM appears on the right in the battle (Model B), to account for any potential “position bias” in user comparisons.

Figure 4 presents coefficient estimates for the LLM fixed effects ($\hat{\delta}_m$ terms), capturing average consumer utility for each LLM by the top 7 providers.¹⁹ More recent models tend to generate responses that are on average preferred by consumers. Among top-performing models, Gemini 2.5 Pro yields the highest estimated utility, though other models (e.g., Claude Opus 4 (thinking), and OpenAI’s o3) achieve comparable scores. Rankings broadly align with underlying model size (number of parameters in the LLM): for instance, within the Llama family of models created by Meta, estimated utility increases from Llama 3.1 8B to 70B to 405B (billions of parameters). Finally, reasoning models—such as Gemini 2.5 Pro, Opus-thinking, Sonnet-thinking, and o3—rank at the top of the distribution, indicating that reasoning capabilities are particularly valued by consumers.

¹⁹The plotted fixed effects are from the baseline specification (2) described next, in Table 4.

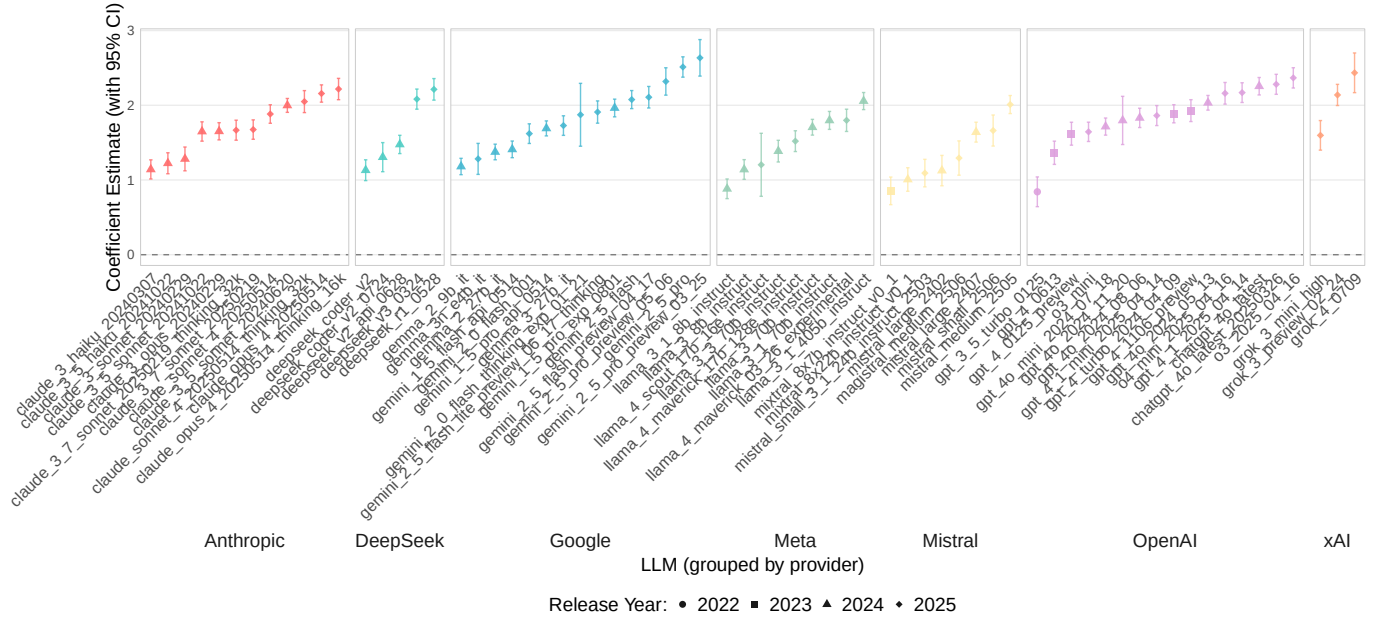


Figure 4: Estimation results: LLM fixed effects by provider

An important implication of these FE estimates concerns the relative performance of open-weight models. The highest-performing open-weight models exhibit FE estimates comparable to those of leading proprietary models offered by OpenAI, Google, Anthropic, and xAI. This pattern is consistent with Artificial Analysis’s benchmark rankings, in which several open-weight models—including models by Meta and Deepseek—are included in the top 10 models by intelligence index.

Table 4 reports estimates for the coefficients $(\hat{\beta}_0^R, \hat{\beta}_0^L, \hat{\Sigma})$, corresponding to the means and standard deviations of the taste parameters for model- and response-level characteristics. Note that the coefficients $\hat{\beta}_0^L$ are obtained in step 2 (GLS), whereas all other coefficients are obtained in step 1 (simulated MLE). Specification (1) uses interactions with topic dummies to control for heterogeneity in tastes across prompts. Specification (2) and (3) use interactions with the PLS scores. Specification (3) replaces the intelligence index with separate coding and knowledge indexes.²⁰ Given that the parameters are identified up to the scale of the idiosyncratic taste shock, the magnitudes of the coefficients are not directly comparable across specifications. In addition, the mean taste parameters under specification (1) correspond to the omitted topic (“Other”) whereas the mean taste parameters under specification

²⁰I do not include the math index (constructed as the average of MATH-500 and AIME scores) due to its high correlation with the coding index, likely because the latter index (based on Scicode and LiveCodeBench) emphasizes scientific and mathematically intensive coding tasks. Including both leads to multicollinearity issues.

Table 4: Estimation Results

	(1)		(2)		(3)	
	Mean (β)	SD (σ)	Mean (β)	SD (σ)	Mean (β)	SD (σ)
<i>Model capabilities</i>						
Intelligence index	0.269 (0.019)	0.056 (0.020)	0.341 (0.010)	0.075 (0.020)		
Knowledge index					0.361 (0.029)	0.045 (0.021)
Coding index					0.036 (0.021)	0.025 (0.039)
Context window	0.044 (0.042)	0.152 (0.061)	-0.027 (0.012)	0.210 (0.059)	-0.021 (0.013)	0.233 (0.057)
Reasoning model	-0.296 (0.099)	0.241 (0.372)	-0.368 (0.028)	0.402 (0.267)	-0.247 (0.027)	0.389 (0.254)
<i>Readability and tone</i>						
Flesch reading ease	0.127 (0.116)	0.219 (0.090)	0.037 (0.042)	0.181 (0.080)	0.031 (0.043)	0.219 (0.089)
SMOG index	0.097 (0.047)	0.092 (0.059)	0.013 (0.016)	0.085 (0.064)	0.012 (0.016)	0.066 (0.069)
Sentiment: positive	0.410 (0.074)	0.365 (0.124)	0.054 (0.026)	0.432 (0.122)	0.052 (0.026)	0.467 (0.118)
Sentiment: negative	-0.350 (0.193)	0.480 (0.188)	-0.337 (0.085)	0.424 (0.185)	-0.332 (0.085)	0.393 (0.195)
Hedging: assertive	-0.077 (0.066)	0.277 (0.072)	-0.044 (0.017)	0.300 (0.071)	-0.048 (0.017)	0.372 (0.065)
Hedging: tentative	-0.178 (0.091)	0.133 (0.184)	-0.215 (0.023)	0.256 (0.113)	-0.215 (0.023)	0.217 (0.137)
<i>Formatting</i>						
Response length	0.745 (0.109)	1.146 (0.051)	0.694 (0.032)	1.098 (0.052)	0.682 (0.031)	1.053 (0.048)
Ordered lists	0.109 (0.075)	0.365 (0.048)	0.043 (0.014)	0.242 (0.048)	0.048 (0.015)	0.273 (0.048)
Unordered lists	0.006 (0.104)	0.287 (0.073)	0.031 (0.022)	0.297 (0.072)	0.028 (0.022)	0.248 (0.086)
Markdown headers	0.311 (0.101)	0.480 (0.073)	0.032 (0.021)	0.310 (0.068)	0.037 (0.022)	0.424 (0.073)
Bold text	0.098 (0.111)	0.612 (0.056)	0.103 (0.025)	0.680 (0.048)	0.094 (0.024)	0.617 (0.050)
Has emoji	0.082 (0.107)	0.831 (0.155)	0.093 (0.036)	1.176 (0.146)	0.089 (0.035)	0.996 (0.151)
Has table	-0.292 (0.186)	1.162 (0.179)	0.207 (0.048)	0.852 (0.214)	0.213 (0.049)	0.970 (0.191)
LLM FE	✓		✓		✓	
Random Coefficients	✓		✓		✓	
Topic dummies	✓					
PLS scores			✓		✓	
Observations	85,257		85,257		85,257	
# LLM FE	70		70		70	
Log-likelihood	-82036.0		-81763.5		-81699.1	

Note: Standard errors are clustered at the user level and shown in parentheses. Mean estimates for model capabilities (and the constant) are obtained via GLS regression of LLM FEs on model capabilities; their standard errors are GLS standard errors computed using the full covariance matrix of the estimated LLM FEs. Specification (1) includes topic dummies and interactions of all variables with the topic dummies. Specifications (2) and (3) include interactions of all variables with each prompt-level PLS score. The reference topic category is Other/Multimedia, and the reference categories for the tone dummies are neutral sentiment and neutral hedging. Additional controls not reported: dummy for whether the model appears on the right (model B), and fixed effects and random coefficients on indicators for LLMs in the other categories (i.e., not belonging to the top 7 providers).

(2) and (3) correspond to the “average” prompt (i.e., the PLS scores are centered).

Across all specifications, consumer utility from a response increases with benchmark performance, as captured by the intelligence and knowledge index.²¹ This indicates that consumer preferences for LLM responses are strongly aligned with an LLM’s benchmark performance. Utility from a response is only weakly responsive to an LLM’s context window size. This weak effect is expected, as most LMArena conversations are single-turn, rendering the context window size non-binding in practice. Conditional on benchmark scores, reasoning models are on average less preferred, likely reflecting their slower response speed.

Most formatting features—such as use of lists, bold text, and structured elements like tables—have a positive mean effect on the utility from a response. Longer responses are also preferred by users, all else equal. Note, however, that because the accuracy and completeness of individual responses are not directly observed, the response length may capture both a taste for verbosity and correlated dimensions of response quality. For example, a comprehensive answer to a complex query may require more tokens than a superficial one. The estimated coefficient on response length should thus be interpreted as a reduced-form valuation of verbosity and any completeness or accuracy associated with longer outputs.

Turning to readability and tone, users prefer responses with positive relative to neutral sentiment, and systematically dislike negative sentiment in responses. Assertive and tentative responses are also penalized relative to neutral; in specifications (2) and (3), the disutility from tentative hedging is more than four times larger than that from assertive hedging. However, as shown next, these mean effects mask substantial variation across prompts and topics.

I construct the deterministic component of user tastes accounting for heterogeneity across prompts: $\beta_0^R + \beta_1^{R'}h(\mathbf{E}_p)$ for response-level characteristics, and $\beta_0^L + \beta_1^{L'}h(\mathbf{E}_p)$ for model-level characteristics. The function $h(\mathbf{E}_p)$ corresponds to topic indicators under specification (1) and PLS scores under specification (2) and (3). Figures 5, 6, and 7 plot the distribution of prompt-specific coefficients, under specification (3), for a subset of variables, across all prompts in the data grouped by topic. A central finding is that many variables exhibit sign variation across and within topics, indicating that the utility derived from a given feature is highly use-case dependent. For instance, Figure 7 shows that the SMOG index—a proxy for linguistic complexity—is positively valued in technical tasks like math and coding, but negatively associated with utility in most practical guidance prompts. A few features, however, show consistent directional effects across topics. Intelligence benchmark scores and

²¹Although the mean effect of the coding index is statistically insignificant in Specification (3), its interaction with PLS scores is statistically significant as shown in Figure 5, highlighting its importance only for a subset of use cases.

response length are uniformly positive, while tentative tone tends to reduce utility regardless of the domain.

The distribution of coefficients also reveals important tradeoffs. Figure 5 shows that the knowledge index (average of MMLU-Pro, HLE, GPQA Diamond) is positively valued across all prompts. In contrast, the utility from higher coding index (average of LiveCodeBench, SciCode) depends on the task: a higher score is beneficial for math and coding prompts, but negatively associated with utility in at least half of writing, casual conversation, and information-seeking prompts. This pattern suggests that gains in coding ability may come at the cost of fluency or alignment in non-technical domains.

Formatting features also exhibit nuanced effects. The use of tables (Figure 6) is especially useful for information-seeking and practical guidance queries, but provide limited benefit—or even reduce utility—in writing tasks. Bold text, by contrast, appears most effective in highlighting key elements in math and information-seeking conversations, but is largely neutral elsewhere. Finally, sentiment effects vary by topic: users generally prefer responses with positive tone, though the effect is muted or slightly negative in coding-related prompts.

Broadly, the interaction terms with topic dummies (specification (1)) are well aligned with the average taste parameters (across prompts within a given topic) obtained under specification (2) and (3). For instance, Appendix Figure A7 shows that the use of tables is particularly valued in information-seeking and practical guidance prompts, consistent with the average taste parameter for this topic in Figure 6.

A comparison of the standard deviations shown in Table 4 (random coefficients Σ) with the standard deviation of prompt-specific coefficients shown in Figures 5–7 reveals that heterogeneity in taste parameters across prompts exceeds heterogeneity across consumers for model capabilities (intelligence, coding, knowledge indexes). This implies that users broadly agree on the value of core capabilities like reasoning, coding, and knowledge, but that their relative importance varies considerably across use cases. The opposite holds for formatting, readability, and tone: consumer-level variation dominates, suggesting that preferences for presentation style and linguistic framing are more idiosyncratic.

To compare the relative magnitudes of the estimated coefficients, I simulate a hypothetical scenario in which two identical models are compared head-to-head in a battle. In this hypothetical setting, I increase one model- or response-level characteristic by a single unit for one of the two models, while holding all other attributes fixed. Appendix Figure A8 reports the resulting change in win probability, averaged across prompts and broken down by topic. Since baseline win probabilities are 50% (identical models), the estimates capture the percentage point gain (or loss) associated with each characteristic relative to 50%.

The results confirm several patterns documented earlier. Formatting features, such as

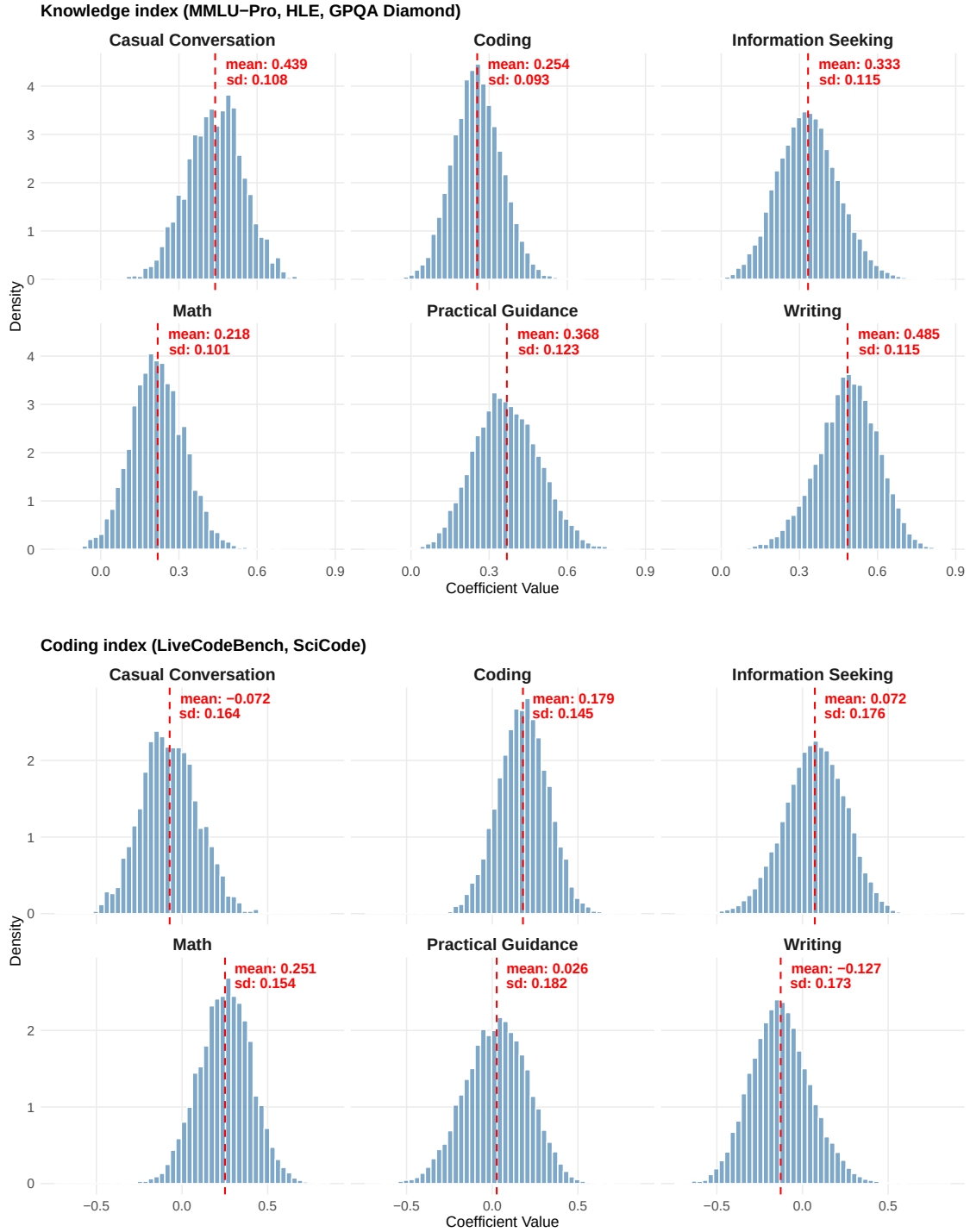


Figure 5: Distribution of taste parameters over prompts by topic: Knowledge and Coding Indices

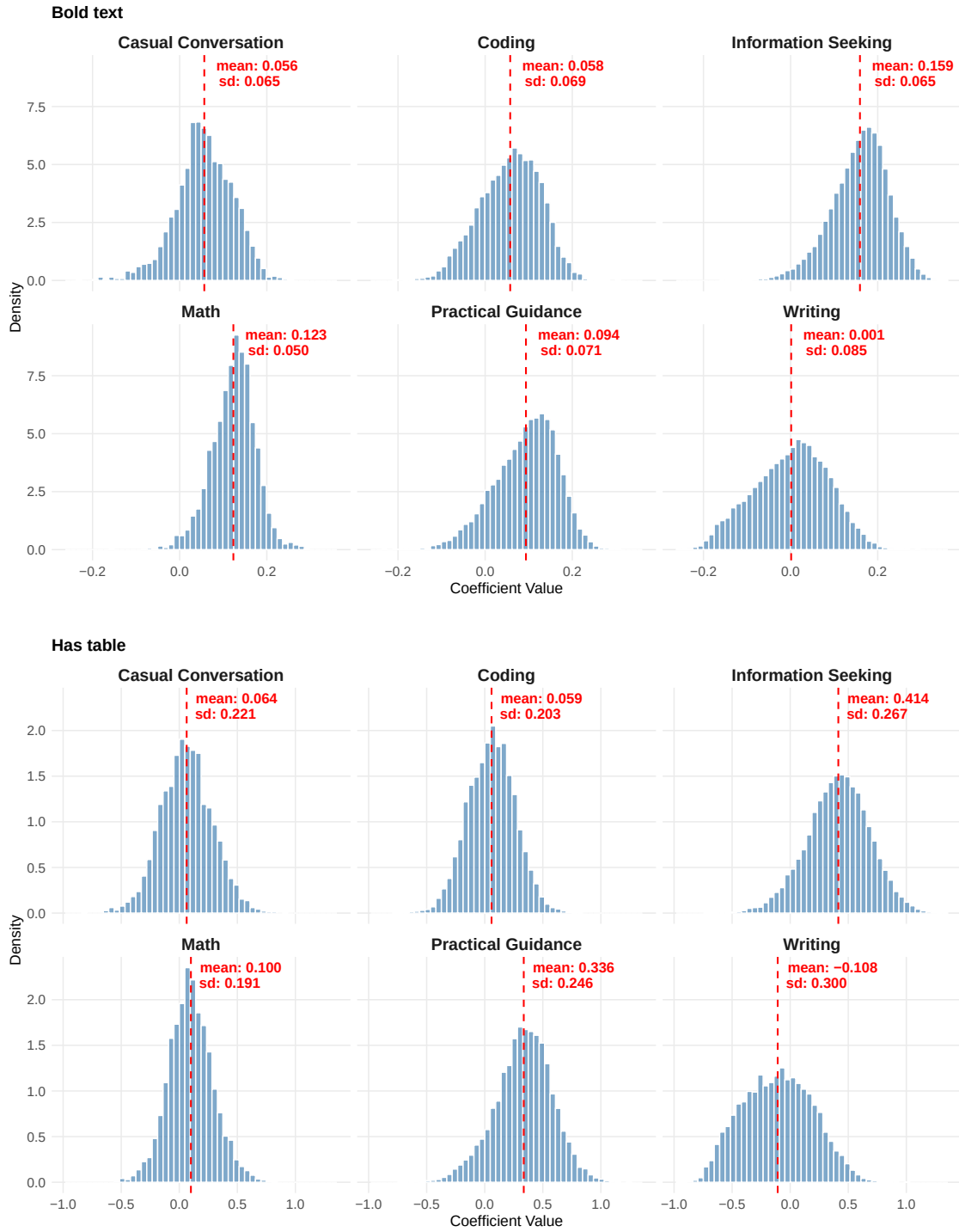


Figure 6: Distribution of taste parameters over prompts by topic: Bold text and table use

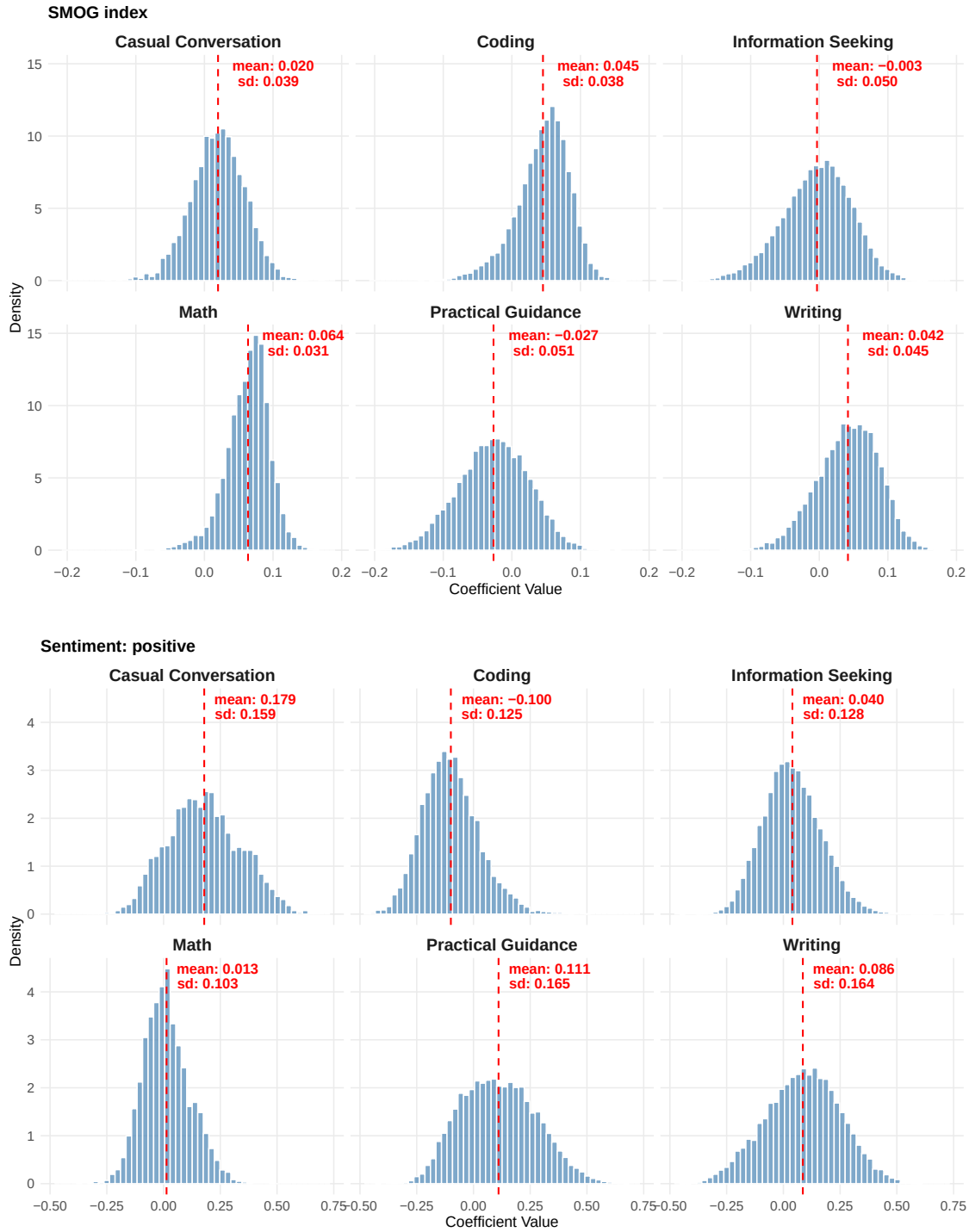


Figure 7: Distribution of taste parameters over prompts by topic: SMOG index and positive sentiment

bold text, tables, and list structures, yield the largest gains in information-seeking and practical guidance contexts, and to a lesser extent in math. Response length improves win probability the most for writing, casual conversation, information-seeking, and practical guidance queries. For information seeking conversations, a one standard deviation increase in response length (about 1,200 tokens) raises the win probability by 13pp whereas a 10pp increase in the knowledge index increases this probability by 8pp.

Model capabilities show more domain-specific effects. Notably, the coding index improves win probability for math and coding prompts but has negative marginal effects in other domains such as writing and casual conversation. This divergence indicates that, whereas model improvements in benchmarks may enhance performance for specific tasks, they can reduce performance elsewhere.

Model fit. The use of random coefficients and interactions with PLS scores improves the fit significantly relative to a specification with only LLM fixed effects. Adding random coefficients lowers the AIC from 169,913 to 166,156. Further incorporating interaction terms with PLS scores (specification (2)) yields the best overall fit, with the lowest AIC of 163,843 and the highest out-of-sample log-likelihood of $-9,072$. In comparison, a similar model using interactions with the first three principal components obtained from PCA of the embedding matrix, instead of PLS scores, performs slightly worse, with an AIC of 165,174 and an out-of-sample log-likelihood of $-9,141$.

The PLS specification captures greater heterogeneity in preferences across prompts and topics (the terms $\beta_0^R + \beta_1^{R'}h(\mathbf{E}_p)$ and $\beta_0^L + \beta_1^{L'}h(\mathbf{E}_p)$ shown in Figures 5, 6, and 7) compared to PCA. The latter yields distribution of coefficients that are more compressed. For instance, the distribution of the coefficient on the Knowledge index exhibits a coefficient of variation (across use cases) of 29.5% under PLS versus 13.4% under PCA, suggesting that there is more than twice as much dispersion in the marginal utility from model intelligence across use cases under PLS than PCA.

Specification (2) lowers the AIC relative to the topic-interaction specification (1), from 164,484 to 163,843, while using 48 fewer parameters. The topic specification remains economically informative because it yields readily interpretable patterns across use cases; however, its inferior fit despite a larger number of parameters suggests that the PLS specification captures preference heterogeneity more effectively.

Robustness checks. In Appendix C, I test the robustness of the estimates in several ways. First, I consider the role of latency, by controlling for a measure of end-to-end response time (at the LLM-level) and analyze the trade-off between intelligence and latency. Second, I verify whether increasing the number of PLS latent components in $h(\mathbf{E}_p)$ beyond three has an impact on the distribution of parameters. Third, I relax the normalization of the

outside option (“both responses bad”) utility by allowing it to vary over time. Fourth, I estimate the discrete choice model that accounts for ties (presented in Appendix B). Finally, I consider the stability of the parameters when increasing the number of simulations draws used to approximate the likelihood function; and I compute bootstrap standard errors to account for uncertainty in the PLS mapping. The baseline results are quantitatively similar and robust to these extensions. Appendix A.2 considers alternative classification approaches for the tone variables (sentiment and hedging) to deal with issues of measurement error in these ML-generated regressors.

6 Counterfactual Analysis

This section develops a structural model of LLM demand to analyze substitution patterns across products and providers. I embed response-level utility estimates from LMArena into a discrete choice demand model over LLMs in the actual consumer market. Under two polar assumptions about user behavior (prompt-level multi-homing and single-homing), I compute demand shares for each LLM and provider and simulate diversion ratios from counterfactual changes in model intelligence. The analysis highlights the importance of consumer multi-homing and heterogeneity in use cases (technical vs. non-technical queries) in shaping substitution patterns.

6.1 A Demand Model for LLMs

The central premise of the demand model is that consumers select LLMs based on the expected utility they derive from model responses to their prompts. In period t , let $\mathcal{J}_t$ denote the set of LLMs available in the market, where each LLM is associated with a provider (e.g., Google, OpenAI) and characterized by observable attributes X_m . For example, in July 2025, $\mathcal{J}_t$ includes models such as Gemini 2.5 Flash and Pro (Google) and Claude Sonnet 4 and Opus 4 (Anthropic). Let $\mathcal{J}_t = \{\mathcal{J}_{ft}\}_{f \in \mathcal{F}}$ denote the partition of available LLMs in period t across firms $f \in \mathcal{F}$, where $\mathcal{J}_{ft} \subseteq \mathcal{J}_t$ is the portfolio offered by firm f .

An important modeling choice concerns the extent to which consumers multi-home—i.e., use multiple LLMs and providers across queries. In practice, some users may switch models on a prompt-by-prompt basis, while others may remain committed to a single provider. To capture this heterogeneity, I consider two polar cases for aggregating response-level utilities. In the first case, *prompt-level multi-homing*, users select the utility-maximizing LLM for each individual query. In the second case, *single-homing*, users commit to a single provider across all queries and select the provider whose portfolio of LLMs delivers the highest expected utility over the user’s prompt distribution. These two extremes correspond to two natural benchmarks.

I derive the demand shares for each product m and firm f under each scenario. The model cannot quantify substitution from the outside option (i.e., not using LLMs) to the inside goods, which is not identified from the LMArena choice data. The analysis is therefore restricted to the market of users who choose among LLMs.

In the remainder of this section, I rewrite the utility function (Equation (1)) as:

$$U_{ipm} = \delta_{ipm} + \epsilon_{ipm}. \quad (8)$$

Each user i is characterized by a taste parameter vector ν_i (constant across prompts) and is endowed with a unit mass of prompts. As defined in Section 4, the embedding $\mathbf{E}_p \in \mathcal{P}$ of each prompt is drawn from the user-specific probability measure F_i , where $F_i(\mathcal{P}) = 1$. Let $H(\nu_i, F_i)$ denote the population distribution of user tastes and prompt distributions.

Prompt-level multi-homing. The share of user i 's prompts addressed to a given LLM m is given by

$$q_{imt}(\nu_i, F_i) = \int \frac{\exp(\delta_{ipm})}{\sum_{m' \in \mathcal{J}_t} \exp(\delta_{ipm'})} dF_i(\mathbf{E}_p) \quad (9)$$

Aggregating across users, the share of prompts addressed to a given LLM m is given by

$$\text{Demand share for model } m: \quad q_{mt} = \int q_{imt}(\nu_i, F_i) dH(\nu_i, F_i) \quad (10)$$

Finally, the share of prompts addressed to firm f is given by

$$\text{Demand share for firm } f: \quad q_{ft} = \sum_{m \in \mathcal{J}_{ft}} q_{mt} \quad (11)$$

Single-homing. Under single-homing, each user selects a single provider to handle all prompts. This provider choice is made before the realization of prompts and their prompt-specific idiosyncratic utility shocks (ϵ_{ipm}). After a prompt and the associated shocks are realized, the user selects the utility-maximizing model from the chosen provider's portfolio. Accordingly, for each user i and prompt p , the expected utility from selecting firm f is given by the logit inclusive value over LLMs in $\mathcal{J}_{ft}$:

$$U_{ipf} = \mathbb{E}_\epsilon \left[\max_{m \in \mathcal{J}_{ft}} \{\delta_{ipm} + \epsilon_{ipm}\} \right] = \ln \left[\sum_{m \in \mathcal{J}_{ft}} \exp(\delta_{ipm}) \right] + \gamma \quad (12)$$

where γ is the Euler constant. Aggregating over prompts using user i 's prompt distribu-

tion $F_i(\mathbf{E}_p)$ gives the expected utility of choosing firm f :

$$\text{Expected utility of user } i \text{ if choosing firm } f: \quad U_{if} = \int U_{ipf} dF_i(\mathbf{E}_p) \quad (13)$$

Each user selects the firm that delivers the highest expected utility (inclusive value) across their prompt distribution. Therefore, the share of prompts (and equivalently users) addressed to firm f is given by²²

$$\text{Demand share for firm } f: \quad q_{ft} = \int \mathbb{1}\{U_{if} = \max_{f' \in \mathcal{F}} U_{if'}\} dH(\nu_i, F_i). \quad (14)$$

For a model m in firm f 's portfolio, demand comes only from users who select firm f . Conditional on selecting firm f , user i assigns prompt p to model m with the within-firm logit probability. The demand share for model m of firm f is given by

$$q_{mt} = \int \int \mathbb{1}\left\{U_{if} = \max_{f' \in \mathcal{F}} U_{if'}\right\} \frac{\exp(\delta_{ipm})}{\sum_{m' \in \mathcal{J}_{ft}} \exp(\delta_{ipm'})} dF_i(\mathbf{E}_p) dH(\nu_i, F_i). \quad (15)$$

Beliefs about response-level characteristics X_{pm} . In the LMArena platform, users evaluate responses after observing them; however, in real-world applications, users must select an LLM before observing its output. This distinction raises the question of how to evaluate utility terms δ_{ipm} that depend on response-level characteristics X_{pm} , which are not known at the time of choice.²³

To address the unobservability of X_{pm} at choice time, I assume users have rational expectations and evaluate LLMs based on the expected utility conditional on the LLM m and the topic category c of the prompt—where the expectation is taken over X_{pm} . That is, when constructing the δ_{ipm} terms entering demand shares and inclusive values, I replace X_{pm} with $\mathbb{E}[X_{pm}|m, c]$, where $c \in \mathcal{C}$ is the topic category of prompt p . This captures the idea that users expect, for example, that Claude models tend to use more complex syntax on writing prompts, or that Google and OpenAI models produce warmer or more enthusiastic responses on practical guidance queries. Because prompts are randomly assigned to models in LMArena, I can obtain unbiased estimates of these conditional expectations by regressing

²²Because each user generates the same mass of prompts, normalized to one, firm f 's share of users equals its share of prompts. Conditional on type (ν_i, F_i) , provider choice is deterministic: there is no additional firm-level taste shock. Ties occur only on a set of user types with measure zero under H (as long as firm portfolios are differentiated).

²³I assume that users observe model-level characteristics X_m when choosing a model. For example, users are aware that Claude 4 Sonnet performs better on benchmarks than Claude 3.7 Sonnet, that o3 is a reasoning model, or that Gemini 2.5 Pro has a larger context window size than similar models. These characteristics are known by consumers without the need to observe model outputs.

X_{pm} on LLM fixed effects and topic category indicators.

Distribution of user prompts F_i . To operationalize the computation of demand shares, it is necessary to specify the distribution of prompt embeddings F_i . I calibrate this distribution using the empirical distribution of conversation topics on ChatGPT reported in [Chatterji et al. \(2025\)](#) (Figure 9). The latter paper uses internal data from OpenAI on approximately 1.1 million conversations from May 2024 through June 2025. The reported shares by topic category are: Casual Conversational (4.3%), Coding (4.6%), Information Seeking (21.3%), Math (3.0%), Practical Guidance (28.3%), and Writing (28.1%). I omit the “Other” and “Multimedia” topic categories from the analysis, as my focus is restricted to text-based interactions, and rescale the remaining six topics’ shares to sum to one.

I assume that all users draw prompts from a common distribution $F_i = F$ over prompt embeddings. Specifically, prompts are drawn in two stages. First, I draw a topic category $c \in \mathcal{C}$ according to the empirical topic shares described above. Then, conditional on category c , I draw a prompt embedding $\mathbf{E}_p$ from the empirical distribution $\hat{F}(\cdot \mid c)$ estimated from the LMArena data. This sampling procedure ensures that the distribution of prompts used in the counterfactual simulations closely mirrors the real-world distribution of conversation topics on ChatGPT.

While the assumption of a common F simplifies the analysis, it is made for convenience and can be relaxed using external data on user-level heterogeneity in prompt distributions.

Diversion ratios. The main objects of interest in the counterfactual analysis are diversion ratios, which summarize the substitution patterns identified by the choice model. For a focal LLM m and an alternative k , the diversion ratio D_{mk} measures the fraction of demand lost by m , following an adverse change in its attributes, that reallocates to k . These ratios reveal the closeness of competition among differentiated products and, together with product ownership, whether lost demand remains within a provider’s portfolio or flows to rival providers; accordingly, they are a central demand-side input into analyses of competition between multi-product firms.

Diversion ratios are typically measured following a price increase or product removal. Because prices are absent in the LMArena choice data, I cannot recover price-based diversion ratios. I instead examine diversion induced by a *ceteris paribus* degradation in model intelligence, an important dimension of competition in this market. [Conlon and Mortimer \(2021\)](#) show that a change in a nonprice product characteristic can be used to construct a local average diversion ratio provided that the characteristic shifts the focal product’s utility in the same direction for every consumer. In the present setting, this monotonicity condition requires the marginal utility of intelligence, $\beta_i^I(p)$, to be positive for every consumer-prompt

pair, which I verify for the intelligence index.²⁴

Given a choice set $\mathcal{J}_t$, with corresponding model-level characteristics $\{X_m\}_{m \in \mathcal{J}_t}$ and response-level expectations $\{\mathbb{E}[X_{pm} \mid m, c_p]\}_{m \in \mathcal{J}_t, c_p \in \mathcal{C}}$, let $h \in \{M, S\}$ index the prompt-level multi-homing and single-homing regimes. I first simulate baseline utilities and the resulting model shares q_{mt}^h by drawing from the distribution of prompt embeddings $\mathbf{E}_p$ and consumer taste heterogeneity ν_i . I then consider each model m in turn and reduce only its intelligence index I_m by $\Delta I < 0$, holding fixed the choice set, all other model characteristics, response-level expectations, and simulation draws. Let $\beta_i^I(p)$ denote user i 's prompt-specific marginal utility from the intelligence index. The counterfactual mean utility of the focal model for user i and prompt p is

$$\delta_{ipm}^{\text{CF}} = \delta_{ipm} + \beta_i^I(p)\Delta I. \quad (16)$$

Let $q_{kt}^{h,m}(\Delta I)$ denote the counterfactual share of model k when model m is degraded, under homing regime h . For $k \neq m$, I define the diversion ratio as

$$D_{mk}^h(\Delta I) = \frac{q_{kt}^{h,m}(\Delta I) - q_{kt}^h}{q_{mt}^h - q_{mt}^{h,m}(\Delta I)}, \quad k \neq m, \quad (17)$$

$$\sum_{k \in \mathcal{J}_t \setminus \{m\}} D_{mk}^h(\Delta I) = 1. \quad (18)$$

The numerator is the change in the destination model's share, while the denominator is the focal model's share loss.

The interpretation of the destination-specific effects differs across homing regimes. Under prompt-level multi-homing, degrading model m lowers only its prompt-level choice probability, so every remaining model weakly gains share and the diversion ratios are nonnegative. Under single-homing, reducing model m 's intelligence also lowers the inclusive value of its provider's portfolio and may cause some users to select a different provider. Demand can therefore fall not only for the focal model but also for un-shocked models in the same portfolio when users leave the provider altogether. Consequently, the diversion ratio for a same-provider model (i.e., Anthropic's Claude Sonnet, if the focal model is Claude Opus) can be negative, while diversion to rival providers can collectively exceed one.

For the empirical exercise, I focus on OpenAI, Google, and Anthropic, three leading providers in the North American consumer LLM market in 2025. To make provider portfolios comparable and prevent differences in the number of included models from mechanically raising a provider's inclusive value, I include two models from each provider: one versatile

²⁴One could analogously study substitution patterns from *improvements* in model intelligence; I focus on degradations because they preserve the conventional interpretation of diversion ratios as apportioning a focal model's lost demand across competitors.

“generalist” model and one higher-capability model intended for more demanding tasks: ChatGPT-4o and o3 from OpenAI, Gemini 2.5 Flash and Gemini 2.5 Pro from Google, and Claude Sonnet 4 and Claude Opus 4 from Anthropic. This common two-model structure isolates substitution patterns from differences in portfolio size and facilitates comparisons of within-provider and across-provider diversion. I reduce each focal model’s intelligence score by 10 points on its original 0–100 scale, holding all other attributes fixed. For example, the shock lowers Claude Opus 4 from 64.4 to 54.4, approximately the observed intelligence score of Claude Sonnet 4 (53.0). The counterfactual simulations use specification (2) of Table 4.²⁵

6.2 Counterfactual simulations

Figure 8 reports diversion ratios following a 10-point reduction in each model’s intelligence index.²⁶ Under prompt-level multi-homing (top panel), each model loses approximately 21–25% of its baseline demand. Substitution is particularly strong within Google’s portfolio: Gemini 2.5 Pro absorbs 38% of Flash’s lost demand, while Flash absorbs 35% of Pro’s. Across providers, Google and OpenAI are the main destinations for each other’s lost demand. OpenAI receives approximately 41–42% of the demand lost by either Google model, compared with 21–23% for Anthropic; conversely, Google receives 48–50% of the demand lost by either OpenAI model, compared with 23–27% for Anthropic. Thus, substitution is substantial within Google’s portfolio, and diversion across providers suggests that Google and OpenAI models are closer substitutes for one another than for Anthropic models.

Under single-homing (bottom panel), portfolio choice changes both baseline demand and the response to a model’s degradation. Anthropic’s combined baseline demand share falls from 19.3% under multi-homing to 8.8%, indicating that its portfolio is less often preferred when consumers select one provider for all their queries. Degrading either Claude model also reduces demand for the unchanged model: consumer switching to rival providers outweighs substitution within Anthropic’s portfolio, so its total share loss is 1.33–1.47 times the focal model’s loss. OpenAI (and, in particular, its high-capability model o3) absorbs most of this lost demand. OpenAI also exhibits negative diversion for its unchanged model, but the additional loss is smaller, at approximately 8–9% of the focal model’s loss. Google, by contrast, retains 14–18% of its focal model’s lost demand through its other model. Substitution between Google and OpenAI becomes more pronounced: OpenAI absorbs about 68–69% of either Google model’s lost demand, while Google absorbs 75–82% of either OpenAI model’s

²⁵Constructing diversion ratios from changes in the coding index used in specification (3) of Table 4 is problematic because the marginal utility from this index may be positive or negative across user-prompt pairs (see Figure 5).

²⁶Predicted market shares do not necessarily match reported market shares in the consumer LLM market, given the data used for estimation is from LMArena, although they are directionally consistent, e.g., if one also accounts for the use of Gemini in information-seeking tasks (via Google’s AI mode).

loss. These results indicate that within-provider substitution can limit a provider’s aggregate share loss following a model’s quality decline, while consumer switching to rival providers can also reduce demand for its unchanged models.

Next, I examine how substitution patterns vary with the distribution of prompts. I vary the share of technical prompts (coding and math), denoted τ , from zero to one, preserving the relative topic weights within the technical and nontechnical groups. That is, I consider a prompt distribution given by²⁷

$$F_\tau = \tau F_{\text{technical}} + (1 - \tau) F_{\text{nontechnical}}. \quad (19)$$

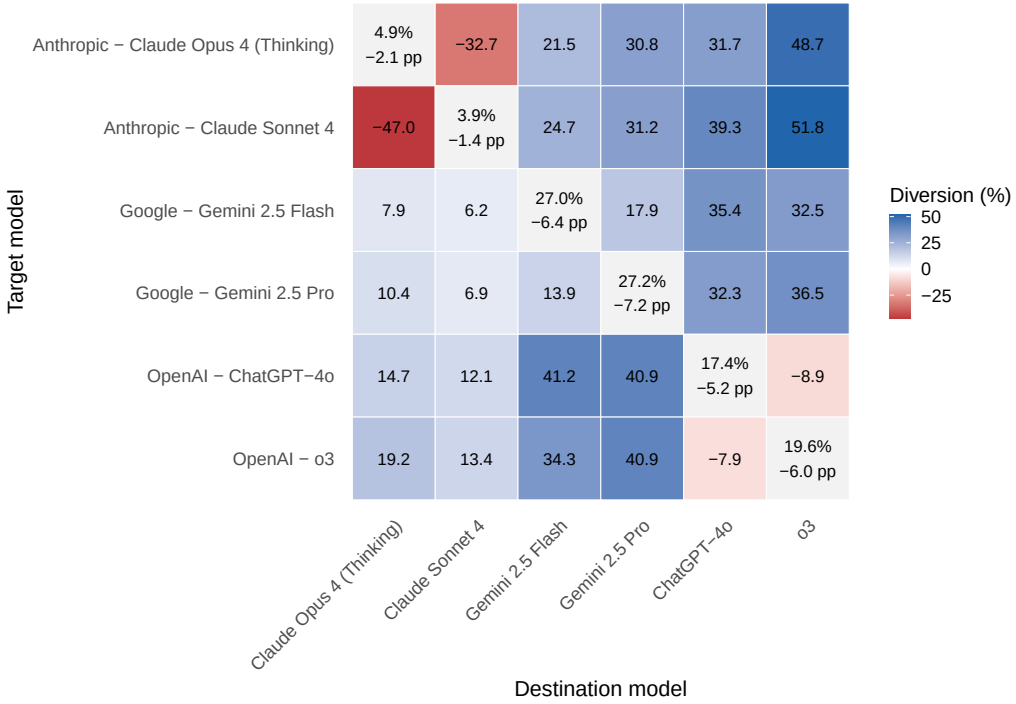
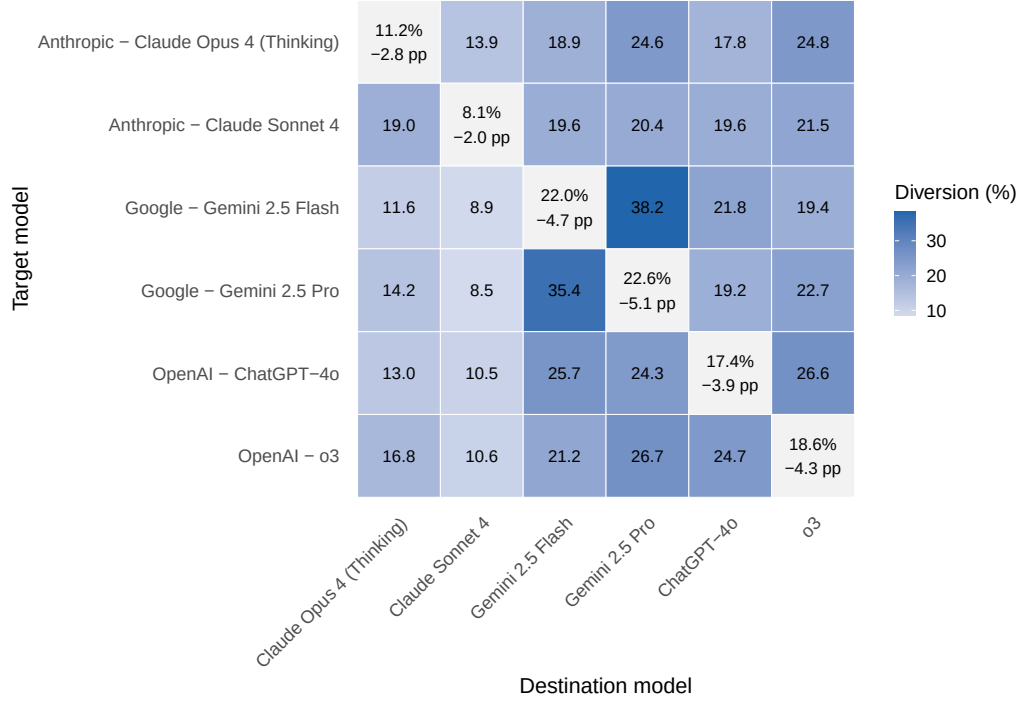
Figure 9 reports diversion following a 10-point reduction in the intelligence index for OpenAI’s o3 or Google’s Gemini 2.5 Pro at each task mix (value of τ). Under prompt-level multi-homing, Anthropic absorbs an increasing proportion of either focal model’s lost demand as technical prompts receive greater weight. Diversion to Claude Opus rises from 16.6% to 19.4% for o3 and from 14.0% to 16.6% for Gemini 2.5 Pro as τ increases from zero to one; diversion to Sonnet also increases, though by less.

The shift toward Anthropic is more pronounced under single-homing. Between the two endpoints for τ , Anthropic’s combined diversion rises from 31.9% to 45.1% for o3 and from 16.7% to 25.3% for Gemini 2.5 Pro, with Claude Opus accounting for most of each increase. Task composition also affects substitution within a provider’s portfolio: diversion from Gemini 2.5 Pro to Flash falls from 14.0% to 8.6%, so Google retains a smaller proportion of Gemini 2.5 Pro’s lost demand as technical prompts become more prevalent. This is consistent with the fact that for technical prompts, high-capability models (o3, Gemini 2.5 Pro, and Claude Opus) are more preferred by consumers. The increasing diversion to Anthropic indicates that its models, particularly Claude Opus, become closer substitutes for the two focal models as the technical-prompt share rises.

Overall, substitution among LLMs depends on how consumers allocate queries across providers and on the distribution of use cases. The extent of multi-homing determines whether consumers substitute at the model level or at the provider level, and therefore whether a quality decline is absorbed within a firm’s portfolio or leads to switching toward rivals. Under single-homing, this switching can also reduce demand for the firm’s remaining products. The mix of tasks further affects the closeness of products: a larger share of technical prompts increases substitution among high-capability models.

²⁷For each group g , $F_g = \sum_{c \in \mathcal{C}_g} \frac{s_c}{\sum_{c' \in \mathcal{C}_g} s_{c'}} \hat{F}(\cdot \mid c)$, where s_c is share of topic c reported in Chatterji et al. (2025) and $\hat{F}(\cdot \mid c)$ is the empirical distribution of LMArena prompt embeddings conditional on that topic. The technical group contains coding and math; the nontechnical group contains casual conversation, information seeking, practical guidance, and writing.

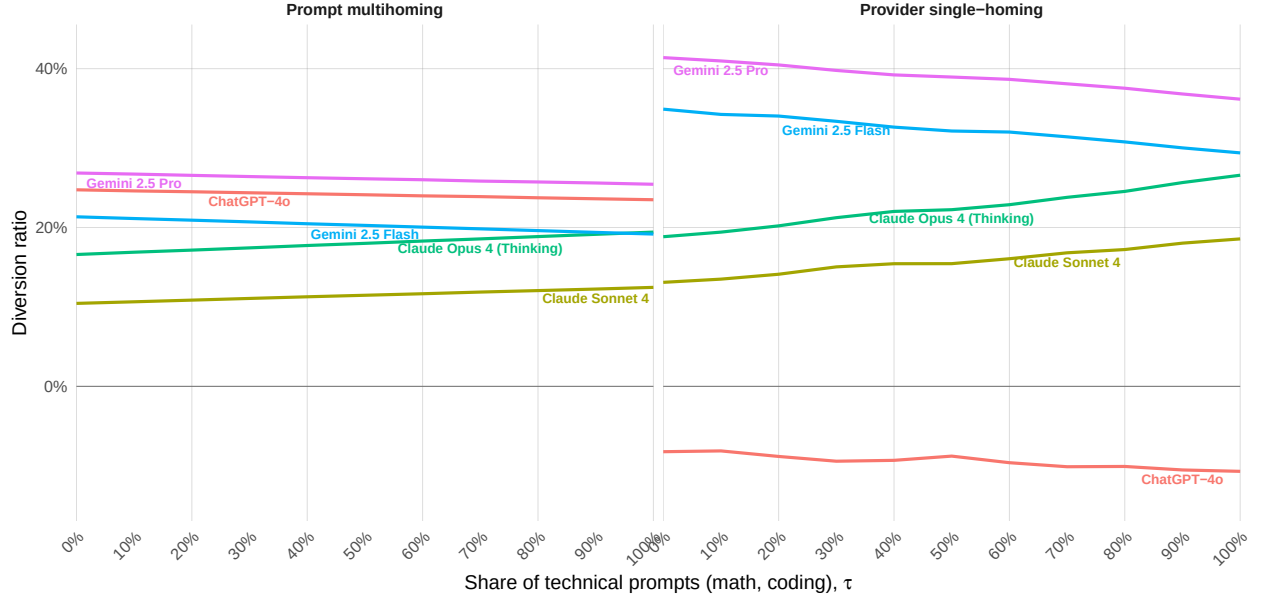
Figure 8: Diversion matrices from a degradation in the intelligence index



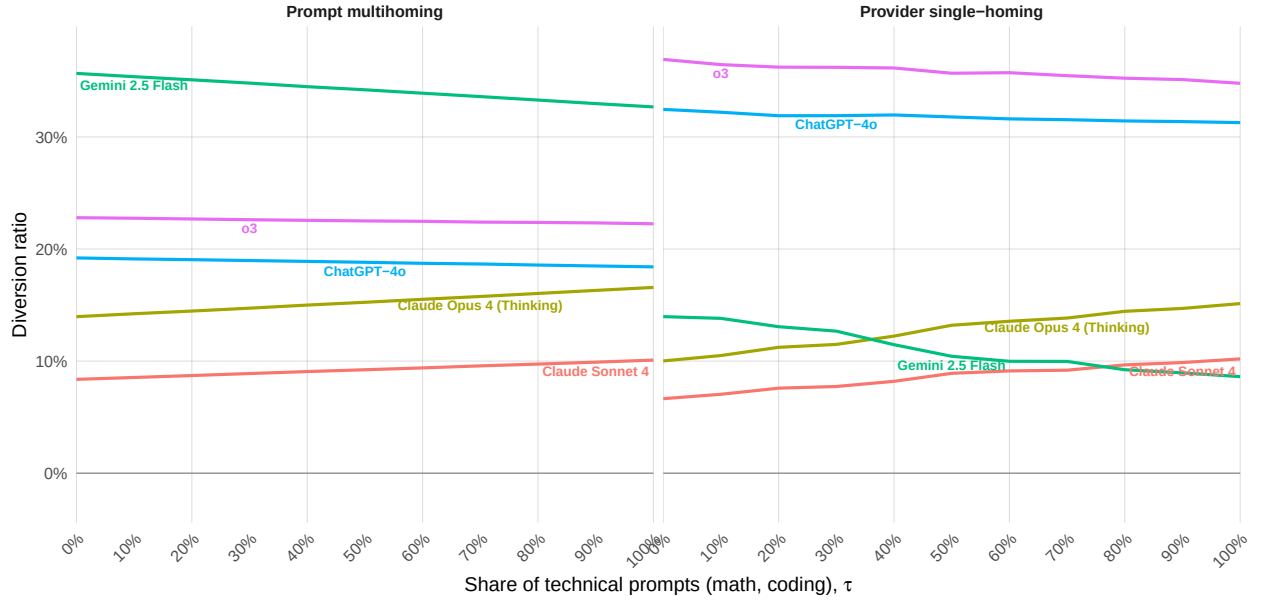
Note: Top: prompt-level multi-homing. Bottom: single-homing. Each row reduces the target model’s intelligence by 10 points. Off-diagonal entries report $D_{mk}^h(\Delta I)$ in %, i.e., the destination share changes as percentages of the target’s share loss; diagonal entries report baseline shares q_{mt}^h and changes in percentage points.

Figure 9: Diversion ratios by the share of technical prompts

(a) Focal model: o3



(b) Focal model: Gemini 2.5 Pro



Note: Each panel reports diversion following a 10-point reduction in the focal model's intelligence index as a function of the share of technical prompts. Left: prompt-level multi-homing. Right: single-homing. Technical prompts comprise coding and math; relative topic weights within the technical and nontechnical groups follow the distribution reported in [Chatterji et al. \(2025\)](#). Diversion ratios are expressed as percentages of the focal model's share loss.

In Appendix D.1, I conduct robustness checks to address the potential bias in the counterfactual objects from misspecification of demand due to the use of ML-generated proxy variables (e.g., response tone variables such as sentiment and hedging). In Appendix D.2, I leverage the demand model to compute the consumer welfare gains between the July 2024 and July 2025 choice sets. Surplus increases by 38% under prompt-level multi-homing and by 54% under single-homing, a range consistent with survey evidence in [Brynjolfsson et al. \(2026\)](#).

7 Conclusion

This paper develops a framework for analyzing consumer demand for LLMs. Using novel revealed-preference data from LMArena, an LLM evaluation platform, I estimate a structural demand model that maps prompt-specific preferences over responses to utilities over LLMs. The model provides estimates of substitution patterns that are key ingredients to quantify the nature of competition in this market. It also provides a granular view of how product features translate into consumer choice.

The estimated model reveals that preferences for LLMs are heterogeneous both across tasks and across users, with the former heterogeneity being more important for model capabilities such as intelligence, and the latter being more important for response style—that is, how models structure and convey information to users. Preferences are broadly consistent with benchmark performance, especially for technical tasks such as coding and mathematics. However, in non-technical domains, horizontal product attributes play a significant role in shaping demand. Simulations of diversion ratios from counterfactual degradation of model intelligence highlight the role of multi-homing and the distribution of use cases in determining which products and providers are close substitutes.

While the LMArena setting offers a unique window into preferences over LLMs, it abstracts from several real-world aspects of the market. Notably, the framework does not account for switching costs and firm-level complementarities (i.e., integration of LLMs into broader ecosystems). Despite these limitations and the rapid evolution of product characteristics in this market, the framework provides a foundation on which to build richer models of consumer demand: it offers tools to decompose preference heterogeneity across tasks and users, characterize product differentiation, and simulate substitution patterns among competing models.

References

- Anthropic.** 2025. “Deprecation Commitments for Claude Models.” <https://www.anthropic.com/research/deprecation-commitments>, Accessed: 2025-12-01.
- Armona, Luis, Greg Lewis, and Georgios Zervas.** 2025. “Learning Product Characteristics and Consumer Preferences from Search Data.” *Marketing Science* 44 (4): 838–855.
- Bach, Philipp, Victor Chernozhukov, Sven Klaassen, Martin Spindler, Jan Teichert-Kluge, and Suhas Vijaykumar.** 2024. “Adventures in Demand Analysis Using AI.” 1–42, <http://arxiv.org/abs/2501.00382>.
- Backus, Matthew, Christopher Conlon, and Michael Sinkinson.** 2021. “Common Ownership and Competition in the Ready-to-Eat Cereal Industry.” *SSRN Electronic Journal*.
- Bai, Yuntao et al.** 2022. “Constitutional AI: Harmlessness from AI Feedback.” <http://arxiv.org/abs/2212.08073>.
- Bajari, P., Z. Cen, V. Chernozhukov, M. Manukonda, S. Vijaykumar, J. Wang, R. Huerta, J. Li, L. Leng, G. Monokroussos, and S. Wang.** 2025. “Hedonic prices and quality adjusted price indices powered by AI.” *Journal of Econometrics* 251.
- Battaglia, Laura, Timothy Christensen, Stephen Hansen, and Szymon Sacher.** 2025. “Inference for Regression with Variables Generated by AI or Machine Learning.” arXiv:2402.15585.
- Berry, Steven, James Levinsohn, and Ariel Pakes.** 1995. “Automobile Prices in Market Equilibrium.” *Econometrica* 63 (4): 841–890.
- Berry, Steven T.** 1994. “Estimating Discrete-Choice Models of Product Differentiation.” *The RAND Journal of Economics* 25 (2): 242.
- Berry, Steven T., and Philip A. Haile.** 2021. “Foundations of Demand Estimation.” *Handbook of Industrial Organization* (2301): .
- Bick, Alexander, Adam Blandin, and David Deming.** 2024. “The Rapid Adoption of Generative AI.” *SSRN Electronic Journal*.
- Brynjolfsson, Erik, Bharat Chandar, and Ruyu Chen.** 2025a. “Canaries in the Coal Mine ? Six Facts about the Recent Employment Effects of Artificial Intelligence.”
- Brynjolfsson, Erik, Avinash Collis, Felix Eggers, Sophia Kazinnik, and David Nguyen.** 2026. “What is Generative AI Worth?” Technical report.
- Brynjolfsson, Erik, Danielle Li, and Lindsey Raymond.** 2025b. “Generative AI at Work.” *Quarterly Journal of Economics* 140 (2): 889–942.
- Carlson, Jacob, and Melissa Dell.** 2026. “A Unifying Framework for Robust and Efficient Inference with Unstructured Data.” arXiv:2505.00282.
- Chatterji, Aaron, Thomas Cunningham, David J. Deming, Zoe Hitzig, Christopher Ong, Carl Yan Shan, and Kevin Wadman.** 2025. “How People Use ChatGPT.” <http://www.nber.org/papers/w34255>.
- Chernozhukov, Victor, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins.** 2024. “Double/Debiased Machine Learning for Treatment and Causal Parameters.” <http://arxiv.org/abs/1608.00060>.
- Chiang, Wei Lin, Lianmin Zheng, Ying Sheng, Anastasios N. Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E.**

- Gonzalez, and Ion Stoica.** 2024. “Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference.” *Proceedings of Machine Learning Research* 235 8359–8388.
- Cho, Minsung, and Jaehyung Kim.** 2026. “InterPol: De-anonymizing LM Arena via Interpolated Preference Learning.” <http://arxiv.org/abs/2603.15220>.
- Christensen, Timothy, and Giovanni Compiani.** 2026. “From Unstructured Data to Demand Counterfactuals: Theory and Practice.” *arXiv preprint arXiv:2601.05374* 2521471 (2521471): .
- Competition and Markets Authority.** 2024a. “AI Foundation Models Technical update report.” (April): , www.nationalarchives.gov.uk/doc/open-government-.
- Competition and Markets Authority.** 2024b. “AI Foundation Models Update paper.” (April): 1–24, <https://www.gov.uk/government/publications/ai-foundation-models-update-paper>.
- Compiani, Giovanni, Ilya Morozov, and Stephan Seiler.** 2023. “Demand Estimation with Text and Image Data.” *SSRN Electronic Journal*.
- Conlon, Christopher, and Julie Holland Mortimer.** 2021. “Empirical properties of diversion ratios.” *RAND Journal of Economics* 52 (4): 693–726.
- Dell’Acqua, Fabrizio, Edward McFowland, Ethan R. Mollick, Hila Lifshitz-Assaf, Katherine Kellogg, Saran Rajendran, Lisa Kraymer, François Candelon, and Karim R. Lakhani.** 2023. “Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality.” *SSRN Electronic Journal*.
- Demirer, Mert, Andrey Fradkin, and Nadav Tadelis.** 2026. “The Emerging Market for Intelligence: How Firms Buy and Sell AI.” *Journal of Economic Perspectives* 40 (3): 23–46.
- Duan, Junting, and Markus Pelger.** 2026. “Inference with AI-Generated Covariates.” NBER Working Paper 35481, National Bureau of Economic Research.
- Egami, Naoki, Musashi Hinck, Brandon M. Stewart, and Hanying Wei.** 2023. “Using Imperfect Surrogates for Downstream Inference: Design-based Supervised Learning for Social Science Applications of Large Language Models.” In *Advances in Neural Information Processing Systems*.
- Egami, Naoki, Musashi Hinck, Brandon M. Stewart, and Hanying Wei.** 2024. “Using Large Language Model Annotations for the Social Sciences: A General Framework of Using Predicted Variables in Downstream Analyses.” Working paper.
- Eloundou, Tyna, Sam Manning, Pamela Mishkin, and Daniel Rock.** 2023. “GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models.” *Science* 384, <http://arxiv.org/abs/2303.10130>.
- Federal Trade Commission.** 2025. “Partnerships Between Cloud Service Providers and AI Developers FTC Staff Report on AI Partnerships & Investments 6(b) Study.” 6 (January): .
- Financial Times.** 2025. “OpenAI needs to raise at least \$207bn by 2030 so it can continue to lose money, HSBC estimates.” <https://www.ft.com/content/23e54a28-6f63-4533-ab96-3756d9c88bad>, Accessed: 2025-12-03.
- Gabel, Sebastian, and Artem Timoshenko.** 2022. “Product Choice with Large Assortments: A Scalable Deep-Learning Model.” *Management Science* 68 (3): 1808–1827.

- Gandhi, Amit, and Aviv Nevo.** 2021. “Empirical Models of Demand and Supply in Differentiated Products Industries.” *Handbook of Industrial Organization*.
- Google.** 2025. “Gemini 3 Announcement.” <https://blog.google/products/gemini/gemini-3/#note-from-ceo>, Accessed: 2025-12-01.
- Han, Sukjin, Eric H. Schulman, Kristen Grauman, and Santhosh Ramakrishnan.** 2024. “Shapes as Product Differentiation.” 1–41, <http://arxiv.org/abs/2107.02739>.
- Handa, Kunal et al.** 2025. “Which Economic Tasks are Performed with AI? Evidence from Millions of Claude Conversations.” <http://arxiv.org/abs/2503.04761>.
- Hartley, Jonathan S., Filip Jolevski, Vitor Melo, and Brendan Moore.** 2025. “The Labor Market Effects of Generative Artificial Intelligence.” <http://dx.doi.org/10.2139/ssrn.5375017>.
- Huang, Yangsibo et al.** 2025. “Exploring and Mitigating Adversarial Manipulation of Voting-Based Leaderboards.” <http://arxiv.org/abs/2501.07493>.
- Kumar, Madhav, Dean Eckles, and Sinan Aral.** 2020. “Scalable bundling via dense product embeddings.” (January): 1–47, <http://arxiv.org/abs/2002.00100>.
- Lee, Kevin.** 2025. “Generative Brand Choice.” 1–60.
- Ludwig, Jens, Sendhil Mullainathan, and Ashesh Rambachan.** 2025. “Large Language Models: An Applied Econometric Framework.” NBER Working Paper 33344, National Bureau of Economic Research.
- Magnolfi, Lorenzo, Jonathon McClure, and Alan Sorensen.** 2025. “Triplet Embeddings for Demand Estimation.” *American Economic Journal: Microeconomics* 17 (1): 282–307.
- Min, Rui, Tianyu Pang, Chao Du, Qian Liu, Minhao Cheng, and Min Lin.** 2025. “Improving Your Model Ranking on Chatbot Arena by Vote Rigging.” *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, <http://arxiv.org/abs/2501.17858>.
- Nagle, Frank, and Daniel Yue.** 2025. “The Latent Role of Open Models in the AI Economy.” *Working paper*.
- Nevo, Aviv.** 2001. “Measuring market power in the ready-to-eat cereal industry.” *Econometrica* 69 (2): 307–342.
- OpenAI.** 2025. “GPT-5.1 Release Announcement.” <https://openai.com/index/gpt-5-1/>, Accessed: 2025-12-01.
- Ouyang, Long et al.** 2022. “Training language models to follow instructions with human feedback.” *Advances in Neural Information Processing Systems* 35.
- Padilla, Nicolas, H Tai Lam, Anja Lambrecht, and Brett Hollenbeck.** 2025. “The Impact of LLM Adoption on Online User Behavior.” (August): .
- Quan, Thomas W, and Kevin R Williams.** 2021. “Extracting Characteristics from Product Images and its Application to Demand Estimation.” 53 (9): 1689–1699.
- Rao, P. V., and L. L. Kupper.** 1967. “Ties in Paired-Comparison Experiments: A Generalization of the Bradley-Terry Model.” *Journal of the American Statistical Association* 62 (317): 194–204.
- Ruiz, Francisco J.R., Susan Athey, and David M. Blei.** 2020. “Shopper: A probabilistic model of consumer choice with substitutes and complements.” *Annals of Applied Statistics* 14 (1): 1–27.

- Singh, Shivalika et al. 2025. “The Leaderboard Illusion.” <http://arxiv.org/abs/2504.20879>.
- The Economist. 2026. “AI Models’ Values Are Very Different from Most People’s.” <https://www.economist.com/briefing/2026/06/25/ai-models-values-are-very-different-from-most-peoples>, June, Briefing, June 25. Accessed: 2026-09-11.
- Tomlinson, Kiran, Sonia Jaffe, Will Wang, Scott Counts, and Siddharth Suri. 2025. “Working with AI: Measuring the Occupational Implications of Generative AI.” 1–42, <http://arxiv.org/abs/2507.07935>.
- Vestager, Margrethe, Sarah Cardell, Jonathan Kanter, and Lina M. Khan. 2024. “Joint Statement on Competition in Generative AI Foundation Models and AI Products.” *Federal Trade Commission* 1–3, <https://www.ftc.gov/legal-library/browse/joint-statement-competition-generative-ai-foundation-models-ai-products>.
- Wold, Herman. 1966. “Estimation of principal components and related models by iterative least squares.” In *Multivariate Analysis*, edited by Krishnaiah, P.R. 391–420, New York: Academic Press.
- Wold, Herman. 1985. “Partial Least Squares.” In *Encyclopedia of Statistical Sciences*, edited by Kotz, Samuel, and Norman L. Johnson Volume 6. 581–591, New York: Wiley.
- xAI. 2025. “Introducing Grok-4 Fast.” <https://x.ai/news/grok-4-fast/>, Accessed: 2025-12-03.

Online Appendix to “Estimating Consumer Preferences for LLMs: Evidence from LMArena”

A Details on the Classification Approach

A.1 Baseline Classification Approach

This appendix describes the classification procedures used to assign conversation topics and evaluate the tone of LLM responses along two dimensions: sentiment and hedging. All classifications are performed using the `openai/gpt-oss-20b` model.

Model Description. `openai/gpt-oss-20b` is a high-performance open-weight language model released by OpenAI in August 2025 under the Apache 2.0 license. It was trained using a mix of reinforcement learning and techniques informed by OpenAI’s advanced internal systems, including `o3` and other frontier models. The model delivers results comparable to `o3-mini` on standard benchmarks. Its combination of strong reasoning performance and low deployment cost makes it suitable for large-scale classification tasks. This is particularly relevant for classifying LLM responses, which is more computationally costly than classifying user prompts due to their substantially higher token counts.

Classification procedure. For topic classification, the LLM receives each user-submitted prompts.¹ For tone classification (sentiment and hedging), the model is instead fed the text of the LLM-generated responses. In each case, classification is conducted by passing the relevant text into `gpt-oss-20b` along with task-specific instructions. To ensure deterministic output, all classification runs are performed with a temperature of zero.

System Prompts. The system prompts used to instruct the model in each classification task are provided at the links below:

- [topic_classification_prompt.txt](#)
- [sentiment_classification_prompt.txt](#)
- [hedging_classification_prompt.txt](#)

Each file contains the full system prompt and classification schema used in that task.

A.2 Measurement Error in LLM-Derived Features

Sentiment and hedging are categorical measurements produced by applying a fixed LLM and classification prompt to each response. If these classifications differ from those that would be

¹For multi-turn conversations, I concatenate all user-submitted prompts into a single input.

assigned under a trusted annotation (“ground-truth”) protocol, the corresponding variables enter the demand system with measurement error. Treating the classifications as error-free can bias the estimated coefficients on tone and on other correlated response attributes; the bias need not be an attenuation bias because the measurement error may be non-classical.

Recent work on the use of ML-generated variables in estimation takes two broad, overlapping approaches to address the measurement error problem. A first group of papers uses ground-truth annotations from a validation sample to correct the discrepancy between downstream estimating objects computed with ML-generated and validated variables: e.g., Egami et al. (2023, 2024), Ludwig et al. (2025), Carlson and Dell (2026), and Duan and Pelger (2026). A second group models the classification-error process more directly. Battaglia et al. (2025) propose analytical bias corrections based on estimable classification-error quantities and, alternatively, a parametric likelihood that jointly estimates the downstream regression and the relationship between the unobserved true label and the classifier’s label. The takeaway from these approaches is that measurement-error correction generally requires either validation data or stronger assumptions that identify the relevant classification-error process.

Implementing these corrections in my setting would require either obtaining a human-annotated validation sample drawn from the estimation population or imposing strong parametric assumptions that identify the relevant classification-error probabilities. Instead, I assess how sensitive the estimates are to two alternative constructions of sentiment and hedging. This exercise addresses the narrower question of whether the estimates depend on the particular procedure used to construct sentiment and hedging, by re-estimating the model using alternative classification approaches.

The first alternative changes the LLM used for classification while holding the annotation task fixed. I reclassify every response using `google/gemini-3.1-flash` rather than the baseline `openai/gpt-oss-20b`. The alternative model receives the same instructions as in the baseline procedure and returns the categorical classifications at temperature zero with a fixed seed.

The second alternative replaces LLM judgment with deterministic, lexicon-based measures. For sentiment, I use `sentimentr`, an R package for dictionary-based sentiment analysis, to score polarized terms while accounting for nearby negators, intensifiers, and adversative language (e.g., “but”, “however”, “nevertheless”). I aggregate the resulting sentence-level evaluations into a response-level score, and use the standardized score in lieu of the categorical sentiment labels when estimating demand. For hedging, I use an explicit dictionary of linguistic markers of epistemic uncertainty, including direct expressions of uncertainty (e.g., “I’m not sure” and “I can’t guarantee”), modal or probabilistic language (e.g., “might,”

“possibly,” and “could indicate”), and scope qualifications (e.g., “appears,” “the evidence suggests,” “it depends,” and “in some cases”). I normalize the number of uncertainty markers by response length and standardize the resulting score, which enters the alternative demand specification as a continuous measure. A higher sentiment score indicates more positive language, whereas a higher hedging score corresponds to more tentative language.

The alternative measurements exhibit meaningful but incomplete alignment with the baseline classifications. The sentiment and hedging labels produced by `gemma-3.1-flash` agree with those from `gpt-oss-20b` in 89.32% and 68.45% of responses, respectively. The lexicon-based scores also vary in the expected direction across the baseline categories: median sentiment scores increase from negative to neutral to positive-labeled responses, while responses labeled tentative have (significantly) higher hedging scores than those labeled assertive or neutral.

I re-estimate specification (2) in Table 4, using the alternative specifications for the sentiment and hedging variables. The results are shown in Appendix Table A5, where specification (1) reproduces the baseline estimates. Specification (2) in Table A5 replaces the baseline sentiment and hedging indicators with indicators constructed using `gemma-3.1-flash` as judge, whereas specification (3) replaces them with the standardized continuous sentiment and hedging scores. Because the categorical and continuous measures have different units, their tone coefficients are not directly comparable in magnitude. The estimates are qualitatively similar across the three specifications, with especially close agreement between the two LLM-based classifications. Under the lexicon-based measures, the coefficients have the expected signs—more positive responses are preferred and more tentative responses are penalized—but are not statistically significant. This lack of precision is not necessarily surprising, given that the lexicon-based scores are relatively coarse measures.

B Modeling Ties in LMArena Battles

In the main text, outcomes are restricted to strict preferences between models or selection of the outside option (“both responses bad”). Here, I outline an extension of the baseline framework that accommodates ties.

A natural way to model ties is to introduce an indifference threshold in the spirit of Rao and Kupper (1967). The key idea is that utility differences below a fixed threshold are not salient and therefore result in a tie. Let $\eta > 0$ denote this threshold parameter. The outcome

o of a battle is then determined by

$$o = \begin{cases} m_A & \text{if } U_{ipm_A} - U_{ipm_B} > \eta \text{ and } U_{ipm_A} > U_{ip0}, \\ m_B & \text{if } U_{ipm_B} - U_{ipm_A} > \eta \text{ and } U_{ipm_B} > U_{ip0}, \\ \emptyset & \text{if } U_{ipm_A} < U_{ip0} \text{ and } U_{ipm_B} < U_{ip0}, \\ \{m_A, m_B\} & \text{if } (U_{ipm_A} > U_{ip0} \text{ or } U_{ipm_B} > U_{ip0}) \text{ and } |U_{ipm_A} - U_{ipm_B}| < \eta. \end{cases}$$

where $o = \{m_A, m_B\}$ corresponds to a tied battle. Under this specification, ties arise when at least one response exceeds the outside-option utility, but the difference in utilities between the two responses lies within the indifference band defined by η . This approach provides a structural interpretation of ties as reflecting near-equality in latent utilities.

Rewriting utility as $U_{ipm} = \delta_{ipm} + \varepsilon_{ipm}$, the following proposition gives closed-form expressions for the choice probabilities of the four battle outcomes.

Proposition 1. *Let $\theta = \exp(\eta) > 1$. Conditional on ν_i , the choice probabilities are*

$$P(o = \emptyset \mid p, \{m_A, m_B\}, \nu_i) = \frac{1}{1 + e^{\delta_{ipm_A}} + e^{\delta_{ipm_B}}}, \quad (20)$$

$$P(o = m_A \mid p, \{m_A, m_B\}, \nu_i) = \frac{e^{\delta_{ipm_A}}}{1 + e^{\delta_{ipm_A}} + \theta e^{\delta_{ipm_B}}}, \quad (21)$$

$$P(o = m_B \mid p, \{m_A, m_B\}, \nu_i) = \frac{e^{\delta_{ipm_B}}}{1 + \theta e^{\delta_{ipm_A}} + e^{\delta_{ipm_B}}}, \quad (22)$$

$$P(o = \{m_A, m_B\} \mid p, \{m_A, m_B\}, \nu_i) = 1 - P(o = \emptyset \mid \cdot) - P(o = m_A \mid \cdot) - P(o = m_B \mid \cdot). \quad (23)$$

Proof. The outcome $o = \emptyset$ occurs when $U_{ipm_A} < U_{ip0}$ and $U_{ipm_B} < U_{ip0}$, i.e., $U_{ip0} > \max(U_{ipm_A}, U_{ipm_B})$. By the multinomial logit formula,

$$P(o = \emptyset) = \frac{1}{1 + e^{\delta_{ipm_A}} + e^{\delta_{ipm_B}}}.$$

The outcome $o = m_A$ requires $U_{ipm_A} > U_{ipm_B} + \eta$ and $U_{ipm_A} > U_{ip0}$. Define $\tilde{U}_{ipm_B} = U_{ipm_B} + \eta = (\delta_{ipm_B} + \eta) + \varepsilon_{ipm_B}$. Since $\tilde{U}_{ipm_B}$ is also Gumbel with location $\delta_{ipm_B} + \eta$, the probability becomes

$$P(o = m_A) = \frac{e^{\delta_{ipm_A}}}{e^{\delta_{ipm_A}} + e^{\delta_{ipm_B} + \eta} + 1} = \frac{e^{\delta_{ipm_A}}}{1 + e^{\delta_{ipm_A}} + \theta e^{\delta_{ipm_B}}}.$$

The choice probability for $o = m_B$ follows by symmetry. The probability of a tie is determined by subtracting the other three probabilities from one. $\square$

The choice probabilities above can be used to form the likelihood of outcomes in the data. Demand estimates based on this extension, with the additional parameter θ , are presented in Appendix C.

C Robustness Checks: Estimation

This appendix considers several extensions to the baseline estimation approach. Specification (2) in Table 4 is used as baseline.

Controlling for response time. As a first robustness check, I consider the trade-off consumers face between model intelligence and response speed. Reasoning models tend to perform better on complex tasks because they generate intermediate reasoning tokens before producing an answer, but this process also increases response time. Because LMArena does not report the time taken by each model to generate a response, I use model-level estimates from Artificial Analysis. The measure is the estimated end-to-end time, in seconds, required to generate a 500-token answer. It includes the time required to process the prompt, generate reasoning tokens (if any), and produce the answer tokens.²

I re-estimate baseline specification (2) from Table 4 after adding end-to-end response time (in seconds). Because the reasoning-model indicator is highly collinear with response time, including both variables leads to multicollinearity; I therefore omit the reasoning indicator from the specification reported in column (1) of Table A1. The estimated mean coefficient on response time is -0.00322 ($SE = 0.000654$), indicating that longer response times reduce consumer utility from a response. The estimated standard deviation of the random coefficient is 0.0141 ($SE = 0.00348$), revealing substantial heterogeneity in consumers' sensitivity to latency.

To quantify the trade-off between performance and latency, I compare the mean response-time coefficient with the coefficient on the intelligence index. At the mean coefficients, a 1-percentage-point increase in intelligence (on the 0-100 scale of intelligence index) offsets the utility loss from approximately 8.68 additional seconds of end-to-end response time.³ More

²Artificial Analysis constructs, for each model, end-to-end response time as

$$\text{End-to-End Response Time} = \text{Input Processing Time} + \frac{\text{Average Reasoning Tokens}}{\text{Reasoning Output Speed}} + \frac{500}{\text{Answer Output Speed}}.$$

Reasoning- and answer-token output speeds are computed from repeated API test runs, each of which applies a newly generated prompt specifying a task such as summarization, question answering, comparative analysis, translation, coding, etc. The average reasoning-token count is computed separately using a diverse set of test prompts.

³Because the intelligence index coefficient reported in the table corresponds to a 10-percentage-point

concretely, users are willing to trade off 1.5 additional minutes of response time in exchange for a response by Claude Opus 4 rather than Claude Sonnet 4. Finally, I calculate the mean PLS-adjusted response-time coefficient within each topic category. Sensitivity to latency is greatest for coding prompts, for which the mean coefficient is -0.00602 , and lowest for math prompts, for which it is -0.00237 . This difference suggests that consumers are more willing to wait for a model to work through a mathematics problem than to generate code.

Number of PLS scores. Second, I increase the number of PLS scores in $h(\mathbf{E}_p)$ from 3 to 5. The mean estimates are shown in specification (2) of Table A1. The resulting parameter estimates remain quantitatively similar to the baseline. Moreover, the distribution of prompt-specific coefficients (shown for a subset of variables in Figures 5, 6, and 7 for the baseline) is essentially unchanged: the distribution of $\beta_0^R + \beta_1^{R'}h(\mathbf{E}_p)$ for response-level characteristics and $\beta_0^L + \beta_1^{L'}h(\mathbf{E}_p)$ for model-level characteristics, are not materially affected by the inclusion of additional PLS scores.

Time-varying outside option. Third, I allow the outside-option utility U_{ip0} to vary over time. In the baseline model of Section 4, the mean utility from the outside option is normalized to zero $U_{ip0} = \varepsilon_{ip0}$. This restriction may be too strong if user expectations evolve over time: as frontier model performance improves, the level of utility required for a response to be judged acceptable may shift, even holding fixed prompt and response characteristics.

I implement this extension by introducing time fixed effects in the outside-option utility. Let $q(p)$ denote the calendar quarter in which prompt p in battle b is evaluated. I specify $U_{ip0} = \tau_{q(p)} + \varepsilon_{ip0}$. Specification (3) of Table A1 shows that quarter FE are precisely estimated and increasing over time (relative to 2024-Q2, the reference period), suggesting that responses need to clear a higher bar over time.⁴ At the same time, the frequency of the “both bad” outcome declines between 2024 and 2025 (Table 1). Taken together, these patterns imply that average response utilities U_{ipm} are rising over time (from changes in the set of models tested and improvements in model and response-level attributes) sufficiently to offset a rising acceptability threshold U_{ip0} . The estimated parameters remain broadly aligned with the baseline estimates.

Accounting for ties. Fourth, I estimate the extension of the model that accounts for ties in LMArena battles, presented in Appendix B. This model features an additional parameter η , that defines an indifference band within which the consumer is indifferent between two good responses, leading to a tie. Specification (4) in Table A1 shows the estimation results. The threshold parameter η is precisely estimated and equals $0.567 (= \exp(-0.566))$. This

increase in benchmark scores, I divide it by ten for this comparison.

⁴Because the utility specification conditions on the prompt $h(\mathbf{E}_p)$, these time fixed effects are not mechanically driven by systematic shifts toward more difficult prompts over time (to the extent that the prompt embedding captures these shifts).

corresponds to 16% of the average utility U_{ipm} over prompts and models available in July 2025.

Increasing simulation draws. I re-estimate the model progressively increasing the number of simulation draws used to approximate the likelihood function (Equation (6)) from 250 to 500.⁵ The resulting parameter estimates (not shown) are quantitatively unchanged, indicating that simulation error is negligible at the baseline draw count and that the estimates (in particular the standard deviations σ 's of the random coefficients) are stable with respect to the numerical approximation of the likelihood.

Accounting for first-stage uncertainty. Several inputs to the demand system are constructed through pre-processing steps. Model characteristics are assembled from external benchmark data, while tone and topic labels and prompt embeddings are generated using pre-trained language models. Conditional on these fixed inputs and pre-trained models, these steps do not introduce additional sampling uncertainty from the LMArena sample: a given response or prompt receives the same classification and embedding when observations are resampled. By contrast, the mapping from the high-dimensional prompt embeddings to the PLS scores is estimated using LMArena data and therefore introduces first-stage sampling uncertainty. The baseline estimation approach does not propagate uncertainty in the estimated PLS mapping to the demand estimates, so the conventional standard errors reported may underestimate the true uncertainty.

To account for this source of uncertainty, I compute standard errors using 150 bootstrap replications in which users are resampled and the complete estimation procedure is repeated, including re-estimation of the PLS mapping, construction of the PLS scores, estimation of the mixed-logit model, and the GLS estimation of the mean coefficients on model characteristics. This procedure is computationally intensive: each likelihood estimation requires approximately 30 GB of memory to store the simulation draws and run the optimization routine, which precludes running the bootstrap replications in parallel. I therefore implement the bootstrap using a restricted specification with random coefficients on only the constant, the intelligence index, and response length. Table A2 shows that the bootstrap standard errors are quantitatively similar to the conventional standard errors for the mean coefficients and are slightly larger for the standard deviations of the random coefficients. This suggests that the first-stage PLS derivation does not add substantial uncertainty to the demand estimates.

⁵This robustness check is computationally intensive. Increasing the number of simulation draws substantially raises memory requirements due to the combination of a large number of individuals and high-dimensional integration (each draw involves an 18-dimensional multivariate normal vector). As a result, this estimation cannot be run on standard hardware and required a rented virtual machine with 256GB of RAM.

Table A1: Estimation Results: Robustness Checks

	(1)		(2)		(3)		(4)	
	Mean (β)	SD (σ)	Mean (β)	SD (σ)	Mean (β)	SD (σ)	Mean (β)	SD (σ)
<i>Model capabilities</i>								
Intelligence index	0.280 (0.010)	0.081 (0.016)	0.337 (0.010)	0.089 (0.017)	0.391 (0.011)	0.059 (0.021)	0.311 (0.008)	0.028 (0.019)
Context window	-0.028 (0.014)	0.112 (0.080)	-0.018 (0.012)	0.190 (0.059)	-0.025 (0.012)	0.150 (0.059)	-0.023 (0.008)	0.140 (0.028)
Response time (sec)	-0.003 (0.001)	0.014 (0.003)						
Reasoning model			-0.339 (0.028)	0.204 (0.433)	-0.390 (0.028)	0.269 (0.359)	-0.344 (0.023)	0.438 (0.116)
<i>Readability and tone</i>								
Flesch reading ease	0.021 (0.041)	0.145 (0.071)	0.034 (0.044)	0.155 (0.089)	0.038 (0.043)	0.199 (0.084)	0.029 (0.040)	0.241 (0.081)
SMOG index	-0.002 (0.016)	0.188 (0.037)	0.011 (0.017)	0.178 (0.045)	0.012 (0.016)	0.106 (0.050)	-0.006 (0.014)	0.240 (0.028)
Sentiment: positive	0.057 (0.026)	0.443 (0.121)	0.043 (0.027)	0.386 (0.126)	0.054 (0.026)	0.484 (0.100)	0.040 (0.023)	0.753 (0.062)
Sentiment: negative	-0.345 (0.085)	0.349 (0.260)	-0.288 (0.085)	0.003 (0.513)	-0.330 (0.083)	0.232 (0.359)	-0.357 (0.072)	0.686 (0.142)
Hedging: assertive	-0.048 (0.017)	0.373 (0.064)	-0.042 (0.017)	0.371 (0.066)	-0.043 (0.017)	0.339 (0.068)	-0.021 (0.014)	0.543 (0.039)
Hedging: tentative	-0.214 (0.023)	0.058 (0.161)	-0.215 (0.023)	0.195 (0.127)	-0.216 (0.023)	0.255 (0.124)	-0.200 (0.020)	0.490 (0.065)
<i>Formatting</i>								
Response length	0.658 (0.032)	1.067 (0.051)	0.681 (0.034)	1.125 (0.053)	0.678 (0.032)	1.057 (0.052)	0.626 (0.026)	1.122 (0.039)
Ordered lists	0.031 (0.014)	0.234 (0.045)	0.053 (0.016)	0.348 (0.045)	0.049 (0.015)	0.341 (0.049)	0.035 (0.011)	0.177 (0.034)
Unordered lists	0.051 (0.022)	0.228 (0.077)	0.035 (0.023)	0.234 (0.089)	0.029 (0.022)	0.276 (0.072)	0.021 (0.018)	0.260 (0.045)
Markdown headers	0.028 (0.022)	0.436 (0.077)	0.042 (0.022)	0.407 (0.069)	0.036 (0.022)	0.410 (0.073)	0.029 (0.016)	0.192 (0.052)
Bold text	0.151 (0.025)	0.750 (0.047)	0.074 (0.026)	0.705 (0.049)	0.100 (0.025)	0.625 (0.056)	0.048 (0.019)	0.390 (0.034)
Has emoji	0.106 (0.037)	1.139 (0.145)	0.091 (0.038)	0.911 (0.162)	0.089 (0.035)	1.026 (0.148)	0.068 (0.028)	0.707 (0.104)
Has table	0.214 (0.049)	0.932 (0.196)	0.161 (0.050)	0.820 (0.207)	0.209 (0.048)	0.933 (0.202)	0.163 (0.037)	0.707 (0.130)
<i>Quarter dummies</i>								
2024-Q3					0.192 (0.047)			
2025-Q2					0.698 (0.194)			
2025-Q3					0.863 (0.197)			
<i>Threshold parameter</i>								
$\ln(\eta)$							-0.566 (0.008)	
Observations	85,257		85,257		85,257		102,333	
Log-likelihood	-81933.9		-81460.4		-81737.3		-127442.0	

Note: Standard errors are clustered at the user level and shown in parentheses. Mean estimates for model capabilities (and the constant) are obtained via GLS regression of LLM FEs on model capabilities; their standard errors are GLS standard errors computed using the full covariance matrix of the estimated LLM FEs. All specifications include LLM fixed effects, random coefficients, and interactions of all variables with each prompt-level PLS scores. The reference categories for the tone dummies are neutral sentiment and neutral hedging. Spec (1): controls for end-to-end response time. Spec (2): 5 PLS components instead of 3. Spec (3): time-varying outside option (quarter dummies). Spec (4): accounts for tied battles. Additional controls not reported: dummy for whether the model appears on the right (model B), and a fixed effect and random coefficient on the indicator of LLMs in the other category (i.e., not belonging to the top 7 providers).

Table A2: Estimation Results: Bootstrap Standard Errors

	Mean (β)			SD (σ)		
	Coefficient	Conv. S.E.	Bootstrap S.E.	Coefficient	Conv. S.E.	Bootstrap S.E.
<i>Model capabilities</i>						
Intelligence index	0.333	(0.009)	(0.009)	0.068	(0.014)	(0.020)
Context window	-0.011	(0.007)	(0.008)			
Reasoning model	-0.370	(0.024)	(0.022)			
<i>Readability and tone</i>						
Flesch reading ease	-0.005	(0.032)	(0.034)			
SMOG index	0.002	(0.013)	(0.014)			
Sentiment: positive	0.048	(0.022)	(0.038)			
Sentiment: negative	-0.285	(0.073)	(0.075)			
Hedging: assertive	-0.031	(0.015)	(0.015)			
Hedging: tentative	-0.186	(0.020)	(0.021)			
<i>Formatting</i>						
Response length	0.616	(0.027)	(0.038)	1.029	(0.040)	(0.060)
Ordered lists	0.027	(0.010)	(0.011)			
Unordered lists	0.013	(0.016)	(0.018)			
Markdown headers	0.013	(0.015)	(0.016)			
Bold text	0.019	(0.015)	(0.020)			
Has emoji	0.074	(0.027)	(0.031)			
Has table	0.158	(0.035)	(0.041)			
Observations	85,257					
Log-likelihood	-81959.8					
Bootstrap replications	150					

Note: Conventional standard errors do not account for uncertainty from estimating the PLS scores. Bootstrap standard errors are computed from 150 bootstrap replications (where sampling is at the user level), with the PLS regression re-estimated in each replication. Mean estimates for model capabilities are obtained via GLS regression of LLM fixed effects on model capabilities. The specification includes LLM fixed effects, random coefficients on the constant, intelligence index, and response length, and interactions of all variables with each prompt-level PLS score. The reference categories for the tone dummies are neutral sentiment and neutral hedging. Additional controls not reported are a dummy for whether the model appears on the right (model B), and fixed effects and random coefficients for LLMs in the other category (i.e., not belonging to the top 7 providers).

D Robustness Checks: Counterfactual Analysis

D.1 Bias Correction for the Diversion Ratios

The counterfactual predictions depend on whether the observed response and model characteristics adequately capture the underlying dimensions of product differentiation. As emphasized by Christensen and Compiani (2026), variables extracted from unstructured data may be imperfect proxies for the attributes that drive consumer choices and substitution patterns. Treating these proxies as the true attributes can therefore misspecify demand, affecting counterfactual predictions both directly, through the characteristics entering the prediction, and indirectly, through the estimated demand parameters.

Christensen and Compiani (2026) propose a post-estimation correction for this source of bias. Starting from demand estimates that treat the proxies as correctly measured, the procedure expresses utilities in terms of composite parameters combining product proxies and preference coefficients. It then evaluates the likelihood score in an expanded parameterization that relaxes the restrictions imposed by the proxies. Discrepancies between observed and

predicted choices in these additional directions are translated into an adjustment to the counterfactual of interest. Under regularity conditions, this adjustment removes the first-order effect of proxy error without requiring the expanded model to be re-estimated.

In the baseline specification, both the PLS scores summarizing prompt content and the sentiment and hedging classifications of responses may imperfectly represent the underlying attributes relevant to choice. These variables vary across prompts and responses, whereas the correction in [Christensen and Compiani \(2026\)](#) is formulated for product-level proxies. To bring the specification closer to their framework, I replace the PLS scores with topic indicators (as in specification (1) of Table 4) and replace response-level sentiment and hedging with their averages by model.⁶ The four tone variables—positive sentiment, negative sentiment, assertive hedging, and tentative hedging—enter through interactions with topic indicators and random coefficients; their common mean effects are absorbed by model fixed effects. I re-estimate this specification, treating the remaining controls as correctly measured. The exercise thus focuses on misspecification of model-level tone dimensions.

The composite parameters combine topic effects and the topic-specific contributions of model capabilities and tone into *model* $\times$ *topic* mean-utility effects. The expanded parameterization allows these effects to vary freely and relaxes the restrictions that the tone proxies impose on the covariance of random utilities across models. Because the *model* $\times$ *topic* effects absorb the capability interactions, they do not separately identify how utility responds to a change in the intelligence index. I therefore examine product-removal diversion ratios rather than the intelligence-degradation experiment in Section 6.2. Product removal can be evaluated using the existing utilities of the surviving models.

For each model in the six-model July 2025 choice set, I remove that model while holding the utilities of all surviving models fixed. For a given homing regime, let q_k^0 denote model k 's baseline demand share and q_k^{-m} its share after model m is removed. The product-removal diversion ratio is

$$D_{mk}^{\text{rem}} = \frac{q_k^{-m} - q_k^0}{q_m^0}, \quad k \neq m. \quad (24)$$

I apply the bias-correction approach in [Christensen and Compiani \(2026\)](#) (Section 3) to each diversion ratio. The expanded score and information matrix, computed from the observed estimation sample, determine a common direction of adjustment. I then evaluate how baseline and removal shares change in that direction and combine these changes using the derivative of the diversion ratio in Equation (24). Thus, the correction accounts for

⁶Allowing latent attributes to vary freely across prompts and responses would introduce observation-specific unknowns whose number grows with the sample. Accommodating this variation over the continuous prompt space represented by the PLS scores would require additional restrictions on how latent attributes depend on prompt content. Topic indicators and model-level tone averages yield a finite collection of model–topic utility effects.

changes in both the destination’s share gain and the removed model’s baseline share.⁷

Figure A1 compares uncorrected and bias-corrected product-removal diversion ratios under prompt-level multi-homing. The results are quantitatively and directionally similar across the two panels. Substitution remains particularly strong within Google’s portfolio: diversion from Gemini 2.5 Flash to Pro changes from 47.5% to 43.9%, while diversion from Pro to Flash changes from 44.5% to 38.0%. Google and OpenAI also remain important destinations for each other’s lost demand. For example, Google’s combined share of diversion from ChatGPT-4o rises from 48.6% to 52.7%, while OpenAI’s combined share of diversion from Gemini 2.5 Pro changes from 36.6% to 34.5%. Although individual entries change, the correction preserves these broad patterns of within- and across-provider substitution. The correction nevertheless produces appreciable changes for Anthropic. The baseline share of Claude Opus 4 rises from 9.8% to 12.2%, while that of Claude Sonnet 4 falls from 7.7% to 4.8%. These changes suggest that the estimated relative appeal of Anthropic’s models is sensitive to the restrictions imposed by the tone proxies.⁸

The bias correction requires reliable derivatives of counterfactual demand shares. Under single-homing, provider choice involves a hard maximum over inclusive values (Equation (14)). In a finite simulation, small parameter changes can therefore cause consumers to switch abruptly between providers, making numerical derivatives sensitive to the consumer draws and the size of the parameter perturbation. This problem can arise even when population shares are smooth after integrating over consumer heterogeneity. I therefore report bias-corrected diversion ratios only for prompt-level multi-homing.

D.2 Welfare Analysis

Consumer surplus can be obtained from the inclusive values used to construct demand shares. This appendix reports the change in surplus implied by the evolution of the choice set over the sample period.

Consumer surplus. Under prompt-level multi-homing, the expected utility of user i from

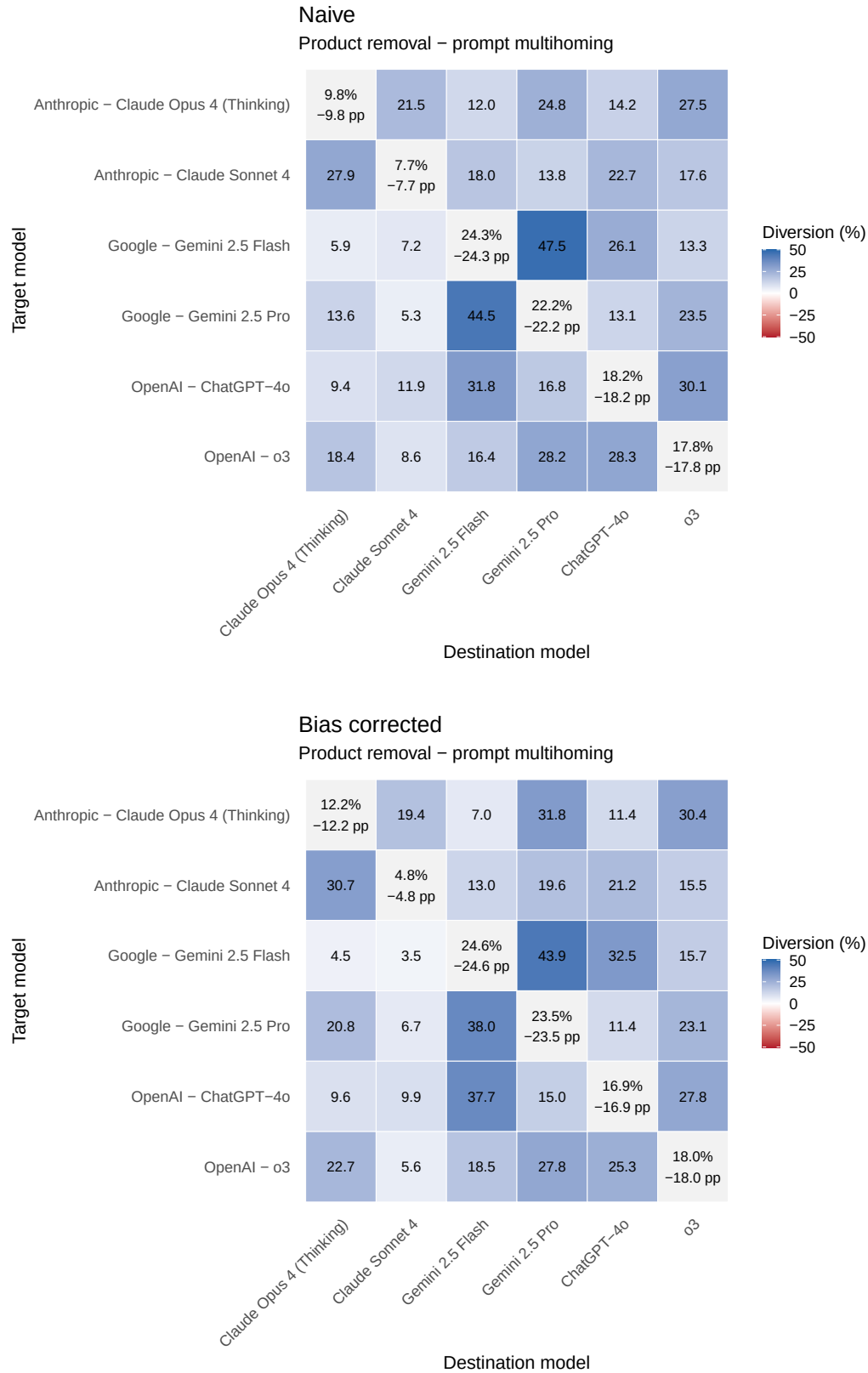
⁷Let $\hat{\gamma}$ denote the composite parameter evaluated at the restricted estimate, $\hat{S}$ the mean expanded likelihood score, and $\hat{H}$ the expanded expected information matrix. Define $\hat{b} = \hat{H}^{-1}\hat{S}$ and $\hat{q}_j^s = \nabla_{\gamma} q_j^s(\hat{\gamma})'\hat{b}$ for baseline or removal scenario s . The corrected diversion ratio is

$$\hat{D}_{mk, bc}^{\text{rem}} = D_{mk}^{\text{rem}} + \frac{\hat{q}_k^{-m} - \hat{q}_k^0 - D_{mk}^{\text{rem}} \hat{q}_m^0}{\hat{q}_m^0},$$

where uncorrected shares and ratios are evaluated at $\hat{\gamma}$.

⁸To assess whether the initial composite parameter estimate is sufficiently close to its population value, Christensen and Compiani (2026) propose the diagnostic $LM_1 = N\hat{S}'\hat{H}^{-1}\hat{S}$. Here, $LM_1 = 1115.73$, below their recommended threshold of $\chi_{844, 0.95}^2 \log N = 10123.93$, with $N = 65,665$. The estimate therefore satisfies their recommended proximity criterion for applying the correction.

Figure A1: Product-removal diversion ratios: bias correction



Note: Each row removes the target model. Off-diagonal entries report diversion ratios in percent; diagonal entries show baseline shares in percent and removal-induced share changes in percentage points. In the bottom panel, diagonal shares are bias corrected, and off-diagonal entries are direct corrections of the diversion ratios.

prompt p , conditional on the choice set $\mathcal{J}_t$, is the logit inclusive value over all available LLMs:

$$U_{ip} = \mathbb{E}_\epsilon \left[\max_{m \in \mathcal{J}_t} \{ \delta_{ipm} + \epsilon_{ipm} \} \right] = \ln \left[\sum_{m \in \mathcal{J}_t} \exp(\delta_{ipm}) \right] + \gamma, \quad (25)$$

where γ is the Euler constant. Integrating over user i 's prompt distribution $F_i(\mathbf{E}_p)$, and then over the population distribution $H(\nu_i, F_i)$ of tastes and prompt distributions, gives

$$\text{Aggregate utility for user } i: \quad U_i = \int U_{ip} dF_i(\mathbf{E}_p), \quad (26)$$

$$\text{Aggregate utility across users:} \quad CS(\mathcal{J}_t) = \int U_i dH(\nu_i, F_i). \quad (27)$$

Under single-homing, each user selects a single provider to handle all prompts. The expected utility of user i from choosing firm f is obtained by integrating the portfolio-level inclusive value (Equation (12)) over the user's prompt distribution, and the user selects the provider delivering the highest expected utility:

$$\text{Expected utility of user } i \text{ if choosing firm } f: \quad U_{if} = \int U_{ipf} dF_i(\mathbf{E}_p), \quad (28)$$

$$\text{Firm-level choice for user } i: \quad U_i = \max_{f \in \mathcal{F}} U_{if}, \quad (29)$$

$$\text{Aggregate utility across users:} \quad CS(\mathcal{J}_t) = \int U_i dH(\nu_i, F_i). \quad (30)$$

Because the LMArena data contain no prices, the model does not identify a price coefficient, and $CS(\mathcal{J}_t)$ is expressed in utility units rather than dollars. I therefore focus on proportional changes in surplus over time.

Consumer surplus changes. Given a choice set $\mathcal{J}_t$, I simulate user utilities and compute consumer surplus under both single- and multi-homing. Simulations are done by drawing from the distribution of prompt embeddings $\mathbf{E}_p$ and consumer taste heterogeneity ν_i .

I evaluate consumer surplus under two market configurations: $\mathcal{J}_0$, the set of models available in July 2024, and $\mathcal{J}_1$, the set available in July 2025. Appendix Table A6 lists the choice sets in these two periods. The proportional change in consumer surplus across these two periods is given by

$$\Delta CS = \frac{CS(\mathcal{J}_1) - CS(\mathcal{J}_0)}{CS(\mathcal{J}_0)}.$$

This change reflects the combined effect of improvements in core model capabilities (X_m) for available LLMs, and shifts in the response-level characteristics of LLM outputs ($\mathbb{E}[X_{pm}|m, c_p]$): e.g., formatting, readability, and tone. These latter changes may arise not only from tech-

nical improvements (e.g., the ability to generate tables) but also from better alignment of model outputs with human preferences over text (e.g., better formatting of responses for practical guidance or information seeking queries).

Results. Consumer surplus per user increased substantially between July 2024 and July 2025 under both homing assumptions. Under prompt-level multi-homing, $\Delta CS = 38.1\%$. Under single-homing, the increase is larger, at 54.0%. The gains are broadly similar across use cases: under multi-homing they range from 34.9% for coding and writing to 45.0% for math, and under single-homing from 49.9% for writing to 68.8% for math.

Comparison with survey evidence. Brynjolfsson et al. (2026) estimate consumer surplus from generative AI using online choice experiments that elicit willingness to accept (WTA) compensation for giving up access to all AI chatbot tools for one month. They estimate a WTA of \$98.00 per user-month in July 2025. I use this figure to anchor the level of consumer surplus in July 2025, $CS(\mathcal{J}_1)$, and translate the simulated utility changes into dollars. The model then implies an average surplus of \$71.0 per user-month in July 2024 under prompt-level multi-homing and \$63.6 under single-homing, so that the improvement in the choice set of LLMs between the two periods is worth \$27.0–\$34.4 per user-month.

Brynjolfsson et al. (2026) find that consumer surplus rose from \$98.00 in July 2025 to \$124.50 in March 2026. With the caveat that their coverage window and the LMArena sample period are adjacent, the two sets of estimates are roughly consistent: the survey-based annual growth rate of 38% falls within the 38%–54% range implied by the structural demand model. This provides an external check on the magnitudes implied by the structural model.

E Details on the Identification Strategy

To illustrate how the random coefficients are identified, consider two battles that share a common focal model m on one side but differ in the characteristics of the opposing model. For the sake of the example, assume models differ along two dimensions: size (a proxy for intelligence) and speed. Let the focal model m be large and slow. Suppose that in the first battle, m is paired against a small, fast opponent n , and in the second battle, m is paired against a larger, slower variant n' that is closer to m in product space. Assume that m wins 60 percent of battles in both comparisons.

Now consider a small change in a characteristic of model m , such as a marginal improvement in its size. Under homogeneous tastes, the induced change in the win rate of m , denoted s_m , is governed by the standard multinomial logit structure: if characteristic j of

model m is perturbed, then

$$\frac{\partial s_m}{\partial x_{m,j}} = \beta_{x_j} s_m (1 - s_m),$$

This expression embodies the IIA restriction: homogeneous tastes imply that the marginal effect of improving m 's characteristics must be the same whether m faces n or n' . In practice, this implication is implausible. Consumers who choose model n over model m are likely to have a higher valuation for speed than the average consumer. As a result, increasing m 's size should induce a greater increase in its win rate when facing n' than when facing n . The failure of the win-rate responses to align across these two battles thus violates the IIA restriction and reveals underlying heterogeneity in user preferences.

F PLS Algorithm and Implementation

Choice of the mapping $h(\cdot)$. The specification of the taste distribution in Equation (2) requires a mapping $h : \mathbb{R}^d \rightarrow \mathbb{R}^g$ that reduces the high-dimensional prompt embedding $\mathbf{E}_p$ (where $d = 3,072$) to a low-dimensional vector suitable for the random-coefficients specification. The guiding principle for constructing $h(\cdot)$ is that the embedding enters the model *only* through the mean taste coefficients: $h(\mathbf{E}_p)$ shifts $\beta_i^R(p)$ and $\beta_i^L(p)$. A useful low-dimensional representation of prompts is therefore one that predicts variation in the taste vector across prompts. This motivates a supervised reduction (PLS) rather than an unsupervised one (PCA). This requirement also informs the choice of supervision target: the target must be an observable, battle-level proxy for the taste vector.

Construction of the target matrix $\mathbf{W}$. To construct the target, I restrict the sample to battles where the consumer prefers one model over the other, excluding observations where the outcome is a tie or “both bad.” The target response matrix must capture the “revealed importance” of model- and response-level characteristics for each battle.

Let b index a specific battle. Let $y_b^* \in \{-1, 1\}$ denote the choice indicator, where $y_b^* = 1$ if model m_B is chosen and -1 if model m_A is chosen. Let $\Delta \mathbf{X}_b$ represent the $(K - 1) \times 1$ vector of differences in response and LLM-level characteristics (excluding the constant term) between the two competing models (i.e., $\Delta \mathbf{X}_b = [X_{pm_B} \ X_{m_B}] - [X_{pm_A} \ X_{m_A}]$). I define the target matrix $\mathbf{W}$ for the PLS regression as an $N \times (K - 1)$ matrix, where N is the number of battles. Each row b of $\mathbf{W}$, denoted by the vector $\mathbf{w}_b$, corresponds to a single battle and is formed by interacting the choice indicator with the vector of characteristic differences:

$$\mathbf{w}_b = y_b^* \cdot \Delta \mathbf{X}_b.$$

Equivalently, for each characteristic k , the element w_{bk} is given by $y_b^* \cdot \Delta x_{bk}$. The vector $\mathbf{w}_b$

serves as a proxy for the gradient of utility with respect to characteristics in that instance. Intuitively, if a user prefers model m_B that generates a significantly longer response ($y_b^* = 1$ and $\Delta x_{\text{length}} > 0$), the corresponding element in $\mathbf{W}$ will be large and positive, signaling a revealed preference for verbosity for that specific prompt. Conversely, if the user rejects the longer response, the element becomes negative.

W as proxy for the taste vector. The interaction $y_b^* \cdot \Delta \mathbf{X}_b$ is the appropriate proxy for the mean of the taste vector. To see why, note first that the choice outcome alone y_b^* could not serve as a target: because model pairs and side labels are randomly assigned on the platform, each side wins about half of the time for every type of prompt, so the mean of y_b^* carries no prompt-level information. What does vary systematically across prompts is how the outcome *covaries* with the characteristic differences. For prompts where users value verbosity, the longer response wins more often than not, so y_b^* and Δx_{length} covary positively; for prompts where users dislike verbosity, they covary negatively. The vector $\mathbf{w}_b$ reflects the battle-level record of these covariances.

Averaging over battles, the conditional mean of $\mathbf{w}_b$ given the prompt is proportional to the prompt-specific mean taste vector.⁹ Cross-prompt variation in the conditional mean of $\mathbf{w}_b$ is thus driven by variation in tastes. The PLS regression of the embeddings on $\mathbf{W}$, described next, therefore extracts the directions of the semantic space of prompts that covary most with taste variation.

Implementation. I perform the PLS regression of the prompt embedding matrix $\mathbf{E}$, which stacks all prompt embeddings $\mathbf{E}_p$ across battles, on the multivariate target matrix $\mathbf{W}$. Thus, $\mathbf{E}$ is an $N \times d$ matrix, where $d = 3072$, and $\mathbf{W}$ is the $N \times (K - 1)$ matrix defined above. The PLS algorithm decomposes $\mathbf{E}$ and $\mathbf{W}$ into latent components by finding weight vectors that maximize the *covariance* between their projections. For the first component, I solve

$$\max_{\mathbf{u}, \mathbf{v}} \text{Cov}(\mathbf{E}\mathbf{u}, \mathbf{W}\mathbf{v})^2 \quad \text{s.t. } \|\mathbf{u}\| = 1, \|\mathbf{v}\| = 1 \quad (31)$$

where $\mathbf{u}$ and $\mathbf{v}$ are weight vectors with lengths d and $K - 1$, respectively. The first latent component (PLS score) for the prompts is given by $\mathbf{z}_1 = \mathbf{E}\mathbf{u}$.

The algorithm then deflates the matrices $\mathbf{E}$ and $\mathbf{W}$ by subtracting the information explained by $\mathbf{z}_1$ and repeats the procedure on the residuals to find subsequent orthogonal

⁹Formally, $\mathbb{E}[\mathbf{w}_b \mid p] \approx \frac{1}{2} \Omega(p) \bar{\beta}(p)$, where $\bar{\beta}(p)$ is the prompt-specific mean taste vector and $\Omega(p) \equiv \text{Var}(\Delta \mathbf{X}_b \mid p)$. Restricting attention to decisive battles and abstracting from the outside option, from taste heterogeneity around the mean, and from the model fixed effects, the model implies $\mathbb{E}[y_b^* \mid \Delta \mathbf{X}_b, p] = 2\Lambda(\bar{\beta}(p)' \Delta \mathbf{X}_b) - 1 \approx \frac{1}{2} \bar{\beta}(p)' \Delta \mathbf{X}_b$, where Λ is the logistic function, linearized around zero. Since random side labeling implies $\mathbb{E}[\Delta \mathbf{X}_b \mid p] = 0$, taking expectations yields the result. The columns of $\Delta \mathbf{X}$ are standardized prior to the PLS step to mitigate cross-prompt variation in $\Omega(p)$.

components. I set the number of iterations, and therefore PLS scores, to three. The final output is an $N \times 3$ matrix $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3]$. The row corresponding to battle b with prompt p , denoted $\mathbf{Z}_b$, corresponds to the vector $h(\mathbf{E}_p)$ used in the utility specification.

I estimate the PLS scores using the Non-Linear Iterative Partial Least Squares (NIPALS) algorithm as implemented in the `scikit-learn` library in Python. Importantly, to ensure that $h(\mathbf{E}_p)$ is not correlated with the unobserved shocks ϵ through its dependence on y^* , I make use of sample-splitting: I recover the PLS weights using only a 10% hold-out sample, then use these weights to project the embeddings for the battles in the main sample and create $h(\mathbf{E}_p)$. This ensures that the constructed scores $h(\mathbf{E}_p)$ are not correlated with realized choices y_b^* in the remaining 90% of battles used to estimate the choice model.¹⁰

Response embeddings. The utility specification in Equation (1) does not use text embeddings of the LLM *responses*, only of the user *prompt*. While recent approaches in demand estimation use product embeddings to capture unobserved product characteristics, that approach is ill-suited for distinguishing between LLM outputs conditional on a specific prompt.

Text embeddings are designed to capture semantic proximity rather than attributes (correctness, completeness, formatting, tone) that drive user choice in this context. Therefore, LLM responses to a given prompt will typically relate to the same core informational content, resulting in very close embedding vectors. For example, a concise, well-structured explanation of quantum computing and a verbose, unstructured block of text on the same topic may be semantically identical (generating high cosine similarity) yet yield different utilities to a user. Since the variance in response embeddings primarily reflects the *topic* of the prompt, relying on explicit response characteristics (e.g., formatting, tone, readability) and latent quality terms (e.g., X_m and ξ_m) provides a more robust basis for estimation.¹¹

G Supplementary Tables and Figures

¹⁰This procedure is closely related in spirit to the cross-fitting idea in double/debiased machine learning (Chernozhukov et al. (2024)), where nuisance components are learned out-of-sample to reduce overfitting and eliminate within-observation reuse. However, my setting is simpler in that I do not require construction of an orthogonal score for the structural parameters: the first-stage dimension reduction is based on fixed-dimensional objects (the dimension of the embeddings $\mathbf{E}$ and the number of PLS components are held fixed as sample size grows), so the learned PLS weights constitute a finite-dimensional nuisance estimator. Consequently, plugging-in $h(\mathbf{E})$ in the likelihood estimation delivers the usual first-order asymptotic behavior.

¹¹In addition, estimates of the effect of changes in response embeddings cannot be mapped into actual firm-level decisions regarding product characteristics, which makes these estimates hard to interpret and leverage in counterfactuals.

Table A3: Conversation Topic Categories

Category	Definition	Example
Coding	Prompts involving writing, modifying, debugging, or explaining code or scripts in any programming language. Includes code snippets, error messages, stack traces, and requests about syntax.	“Why is my SQL query failing? Here’s the code and error message.”
Information Seeking	Prompts asking factual questions such as definitions, explanations, statistics, comparisons, or historical/background information.	“Who discovered penicillin?”
Practical Guidance	Prompts requesting how-to advice, step-by-step instructions, troubleshooting (non-code), or help designing/creating non-text things. Also includes life advice, planning, and product recommendations.	“How do I set up Docker on Windows?”
Writing	Prompts focused on producing, editing, rewriting, summarizing, translating, or improving written text of any kind.	“Rewrite this email to sound more professional.”
Math	Prompts requiring numerical calculations, symbolic manipulation, proofs, algebra, calculus, probability, or solving logic puzzles.	“Solve for x : $3x + 5 = 20$.”
Multimedia	Prompts requesting the creation, editing, or analysis of images or other non-text media.	“Can you please draw me a unicorn in ASCII.”
Casual Conversational	Prompts involving greetings, chit-chat, personal reflections, emotions, philosophical or hypothetical musings, or playful conversation not aimed at producing factual answers or instructions.	“What if humans could live forever?”
Other	Prompts that do not fit the other categories, mainly questions about the AI assistant itself—its training, internal state, limitations, or identity.	“How were you trained?”

Note: In the LMArena text battles, models cannot generate images. When users request images, models typically provide textual descriptions or produce text-based renderings (e.g., ASCII art or character-based drawings).

Table A4: Benchmarks included in the Artificial Analysis Intelligence Index

Benchmark	Field		Description	Paper	Weight
MMLU-Pro	Reasoning & Knowledge		Comprehensive evaluation of advanced knowledge across domains, adapted from original MMLU. 12,032 multiple-choice questions across math, science, law, economics, psychology, business, etc.	Wang et al. (2024)	1/6
HLE (Humanity’s Last Exam)	Reasoning & Knowledge		Academic benchmark across mathematics, humanities, and natural sciences.	Phan et al. (2025)	1/6
GPQA Diamond	Scientific Reasoning		Graduate-level scientific reasoning benchmark (biology, physics, chemistry). Diamond subset (198 questions) selected for discriminative power.	Rein et al. (2023)	1/6
MATH-500	Quantitative Reasoning		Subset of 500 problems from Hendrycks’ MATH dataset (mathematical problem-solving). Uses symbolic and LLM-based equality checking.	Hendrycks et al. (2021) Lightman et al. (2023)	1/8
AIME 2024	Competition Math		Advanced competition-level math problems from the 2024 American Invitational Mathematics Exam.		1/8
SciCode	Code Generation		Python programming for scientific computing.	Tian et al. (2024)	1/8
LiveCodeBench	Code Generation		Python code generation benchmark with tasks from LeetCode, AtCoder, and Codeforces.	Chen et al. (2024)	1/8

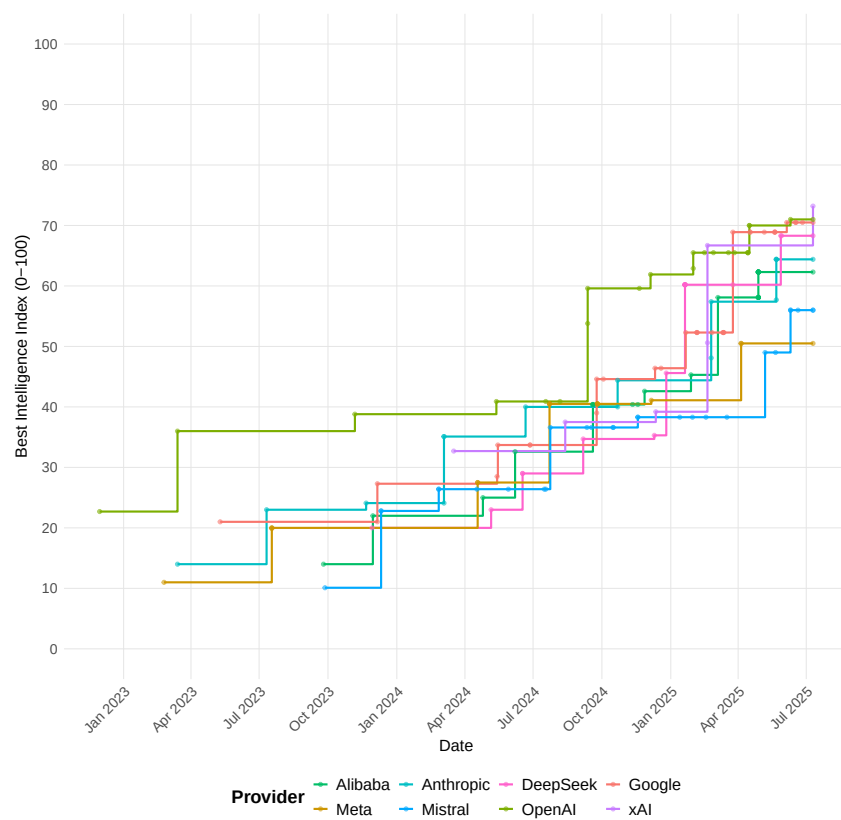


Figure A2: Best Intelligence index (over portfolio of LLMs) by provider over time



Figure A3: Formatting characteristics by LLM: Response length (tokens)

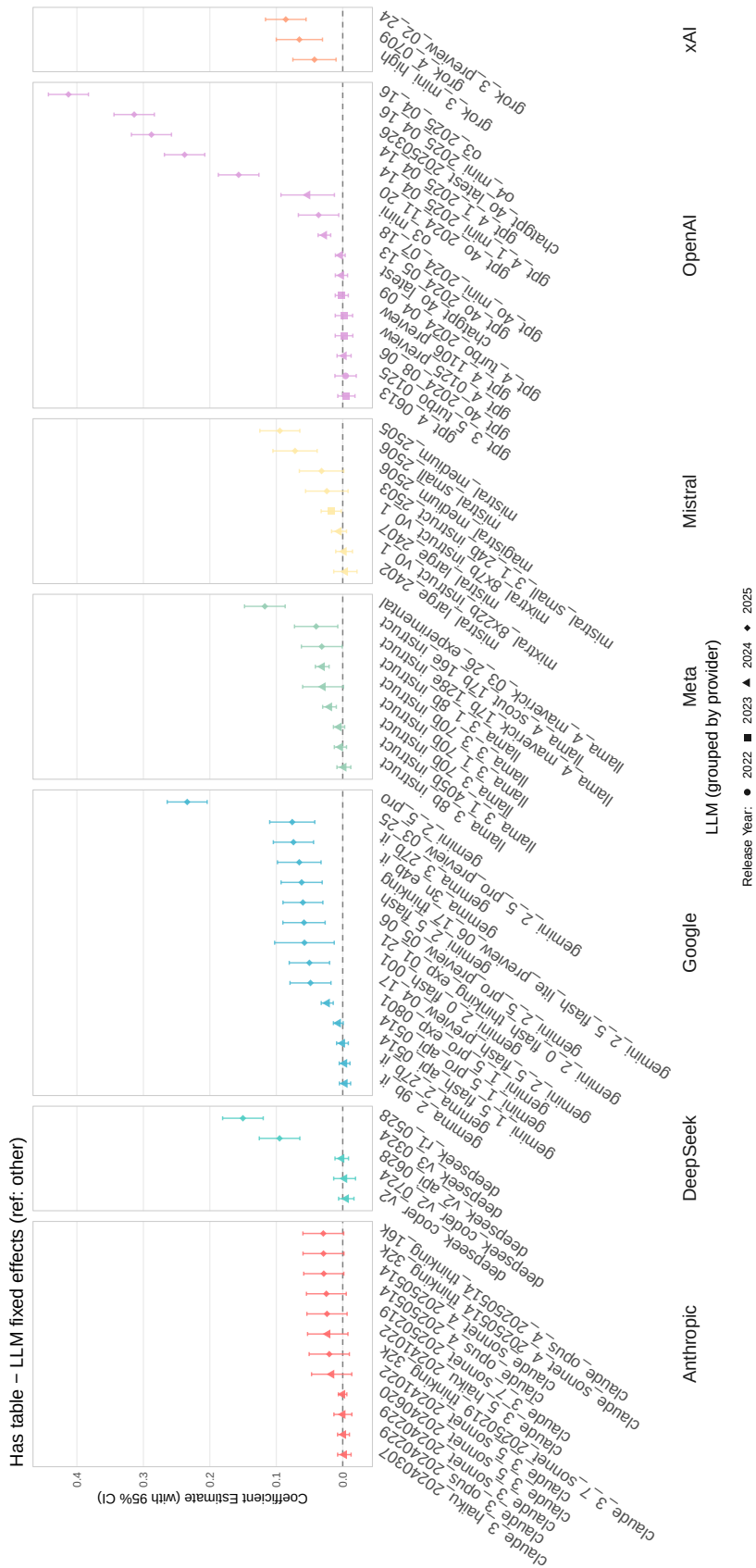
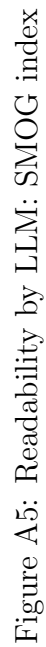


Figure A4: Formatting characteristics by LLM: Use of tables



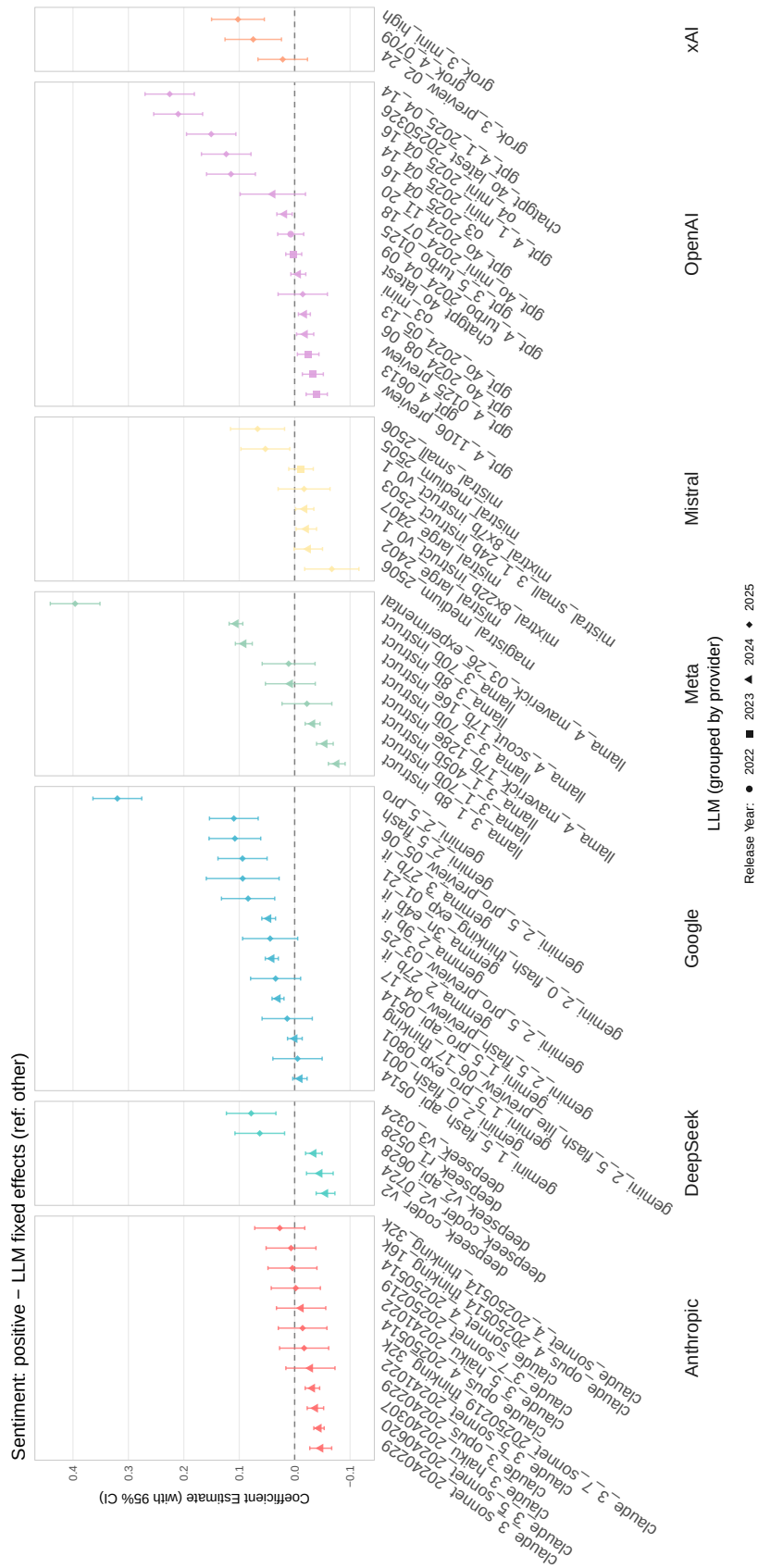


Figure A6: Tone by LLM: Positive sentiment

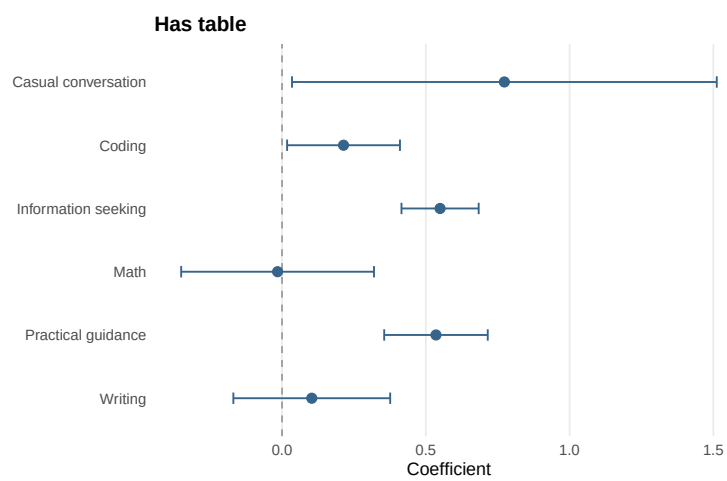
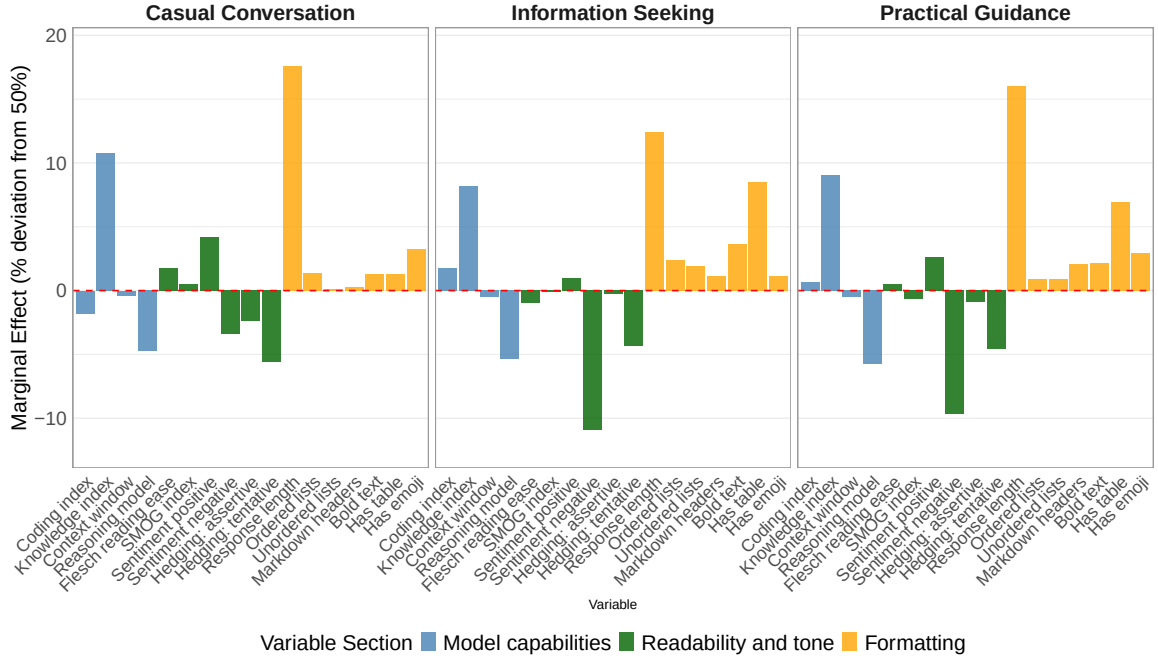
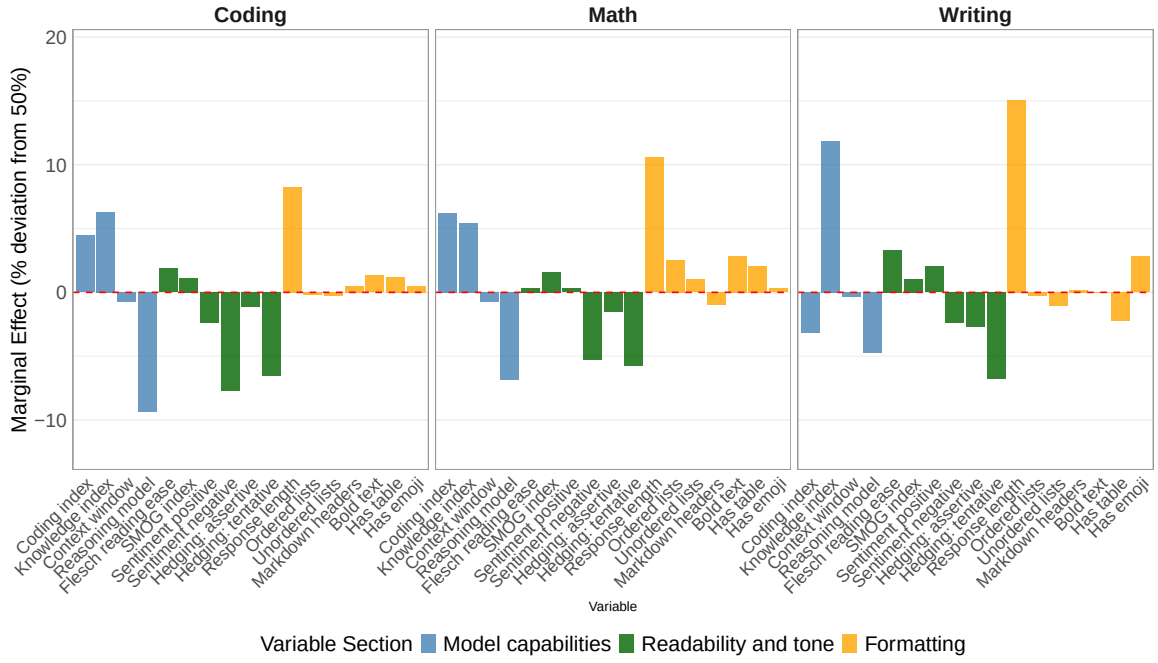


Figure A7: Taste parameter (and 95% CI) for table use by prompt topic



Note: Marginal effects on win probability from a unit change in each regressor in a symmetric battle. For readability, tone and formatting, a unit change corresponds to a one standard deviation increase, except for the dummy variables (sentiment, hedging, has table, has emoji) which are not rescaled. For model capabilities, the effects are calculated based on a 10pp increase in the knowledge or coding index and a 1 million token increase in the context window size. Reasoning is a dummy variable.

Figure A8: Marginal effects on win probability in symmetric battle, by topic

Table A5: Estimation Results: Alternative specifications for sentiment and hedging

	(1)		(2)		(3)	
	Mean (β)	SD (σ)	Mean (β)	SD (σ)	Mean (β)	SD (σ)
<i>Model capabilities</i>						
Intelligence index	0.341 (0.010)	0.075 (0.020)	0.337 (0.010)	0.060 (0.021)	0.343 (0.010)	0.085 (0.018)
Context window	-0.027 (0.012)	0.210 (0.059)	-0.010 (0.011)	0.117 (0.065)	-0.016 (0.012)	0.223 (0.057)
Reasoning model	-0.368 (0.028)	0.402 (0.267)	-0.355 (0.028)	0.407 (0.259)	-0.362 (0.028)	0.161 (0.447)
<i>Readability and tone</i>						
Flesch reading ease	0.037 (0.042)	0.181 (0.080)	0.042 (0.044)	0.252 (0.099)	0.038 (0.042)	0.168 (0.076)
SMOG index	0.013 (0.016)	0.085 (0.064)	0.009 (0.016)	0.083 (0.066)	0.006 (0.016)	0.157 (0.049)
Sentiment: positive	0.054 (0.026)	0.432 (0.122)	0.059 (0.023)	0.281 (0.132)		
Sentiment: negative	-0.337 (0.085)	0.424 (0.185)	-0.399 (0.104)	0.151 (0.405)		
Sentiment score					0.002 (0.009)	0.038 (0.069)
Hedging: assertive	-0.044 (0.017)	0.300 (0.071)	-0.169 (0.017)	0.268 (0.089)		
Hedging: tentative	-0.215 (0.023)	0.256 (0.113)	-0.200 (0.022)	0.186 (0.141)		
Hedging score					-0.009 (0.009)	0.006 (0.102)
<i>Formatting</i>						
Response length	0.694 (0.032)	1.098 (0.052)	0.657 (0.032)	1.068 (0.052)	0.680 (0.032)	1.069 (0.051)
Ordered lists	0.043 (0.014)	0.242 (0.048)	0.043 (0.015)	0.320 (0.049)	0.046 (0.015)	0.290 (0.043)
Unordered lists	0.031 (0.022)	0.297 (0.072)	0.018 (0.021)	0.176 (0.086)	0.028 (0.022)	0.311 (0.070)
Markdown headers	0.032 (0.021)	0.310 (0.068)	0.040 (0.022)	0.409 (0.072)	0.041 (0.021)	0.354 (0.071)
Bold text	0.103 (0.025)	0.680 (0.048)	0.108 (0.025)	0.653 (0.049)	0.102 (0.025)	0.636 (0.047)
Has emoji	0.093 (0.036)	1.176 (0.146)	0.091 (0.036)	1.164 (0.147)	0.105 (0.035)	1.113 (0.152)
Has table	0.207 (0.048)	0.852 (0.214)	0.200 (0.047)	0.801 (0.210)	0.217 (0.049)	1.015 (0.185)
LLM FE	✓		✓		✓	
Random Coefficients	✓		✓		✓	
PLS scores	✓		✓		✓	
Observations	85,257		85,257		85,257	
# LLM FE	70		70		70	
Log-likelihood	-81763.5		-81744.2		-81871.2	

Note: Standard errors are clustered at the user level and shown in parentheses. Specification (1) uses categorical sentiment and hedging labels from *openai/gpt-oss-20b*; specification (2) uses categorical labels from *google/gemini-3.1-flash*; specification (3) uses standardized continuous lexicon-based sentiment and hedging scores. A higher sentiment score indicates more positive language, while a higher hedging score indicates greater epistemic uncertainty and more tentative language. The reference categories for the categorical tone indicators in specifications (1) and (2) are neutral sentiment and neutral hedging.

Table A6: Choice sets

Panel A: July 2024 choice set $\mathcal{J}_0$	
Provider	Models
OpenAI	gpt_4o, gpt_4o_mini (2024-07-18)
Anthropic	claude_3_opus, claude_3_5_sonnet, claude_3_haiku
Google	gemini_1_5_pro, gemini_1_5_flash
DeepSeek	deepseek_v2, deepseek_coder_v2
Meta	llama_3_1
Mistral	mistral_large
Panel B: July 2025 choice set $\mathcal{J}_1$	
Provider	Models
OpenAI	gpt_4o (2025-03-26), gpt_4_1, gpt_4_1_mini, o3, o3_mini, o4_mini
Anthropic	claude_opus_4, claude_opus_4 (thinking), claude_sonnet_4, claude_sonnet_4 (thinking)
Google	gemini_2_5_pro, gemini_2_5_flash
DeepSeek	deepseek_r1, deepseek_v3
Meta	llama_4_maverick, llama_4_scout
Mistral	mistral_medium, mistral_small, mistral_large
xAI	grok_4, grok_3, grok_3_mini

Notes: Grok 2 and Grok 2 mini by xAI are excluded from Panel A because, although tested on LMArena, they do not appear in the available datasets.