Reward Modeling for Business Objectives: Navigating the Bias-Variance Tradeoff between Observational and Experimental Data

Erfan Loghmani
University of Washington

July 1, 2026
For the latest version, please check erfanloghmani.github.io/files/JMP.pdf

Abstract

Unstructured data, such as headline text and ad copy, plays a critical role in shaping consumer decisions in modern marketing. Generative models offer a scalable way to produce this content, but specializing them for business objectives requires high-quality outcome data. While firms typically possess an abundance of historical data, relying on these observational logs risks introducing confounding factors; conversely, unbiased experimental A/B tests are often costly to implement. In this study, we first highlight the dangers of relying solely on historical data, demonstrating how models can internalize spurious correlations. Using an archive of randomized headline tests, we next show that the value of experimental data hinges heavily on how it is utilized; without a comparative ranking approach, simply adding experimental variation to outcome prediction models yields marginal benefits.

Furthermore, to resolve the bias-variance tradeoff between the observational and experimental data, we introduce a bias-corrected, inverse-variance method that fuses a biased observational estimate with an unbiased experimental one. This approach extends the literature on combining observational and experimental data to high-dimensional textual actions. We further embed the proposed method in a time-based simulation of a firm deciding, test by test, whether to experiment or act on what it knows, and find that experiments alone underperform history while our method turns experimentation into a net gain. When experimentation is budget-constrained, informed test selection also becomes important. We evaluate several strategies for choosing which tests to run, and find that the best one depends highly on the setting and the length of the experimentation window, with strong potential in using the two data sources together to decide which tests to run.

1 Introduction

When a customer engages with online content, many factors shape their decision. While quantitative information, such as price, product characteristics, and average review ratings, matters, the unstructured content a customer faces, like product descriptions and images, is also an important driver. Marketers increasingly try to optimize this unstructured content to raise user response and willingness to buy. For example, Thomas (2026) shows that the language used in the headlines of Airbnb listings affects reservation rates. They find that adding a single puffed claim increases bookings by about 0.7 nights per year. Li and Xie (2020) find that the presence of an image, and its characteristics, can lead to higher engagement.

With the rise of generative models, firms are increasingly turning to them to produce this content. In one industry survey, 95% of marketers report testing AI for creative production (Smartly, 2026). Large language and vision models can produce content variants at scale and low cost (Company, 2024), and can even generate creative ideas competitive with human ones (Castelo et al., 2024). Yet generic pretrained models are not well suited to optimizing a firm's business objective, such as click-through rate (CTR) (Goli and Singh, 2024; Ye, Yoganarasimhan, and Zheng, 2024), which makes it important to specialize them for specific tasks. One central way to do so is reward modeling: a reward model assigns an estimated outcome score to each candidate piece of content, and the generative model is fine-tuned to produce content that scores well (Christiano et al., 2017; Ouyang et al., 2022). The reward model therefore plays a critical role. If it is biased or misspecified, the generative model is optimized toward the wrong target. The quality of the specialized model thus hinges on the quality of the reward signal and on how the reward model is built.

Firms can run A/B tests to learn how different content variants perform and estimate the reward model, but doing so carries both engineering and opportunity costs. At the same time, firms hold an abundant store of historical data on how each piece of content drove engagement. Estimating the reward from historical data alone, however, comes with a risk: a model may internalize spurious correlations and generate content that reflects patterns unrelated to what drives better outcomes. We demonstrate this concern in a real-world setting by studying a data source used in prior question-answering fine-tuning work (Askell et al., 2021). In this data, answer scores are strongly correlated with the day an answer is posted, a pattern more likely driven by higher platform activity on certain days than by differences in answer quality. We show that using these scores directly as a training signal leads the language model to internalize the spurious correlation. Experiments avoid this bias because they randomize what content is shown, but they come with costs and, in small samples, are too noisy to rank content reliably. Observational data, though biased, is abundant and can yield low-variance estimates. The result is a bias–variance trade-off, and resolving it well requires using both sources together.

As recent research such as Doremus et al. (2026) points out, most studies on experimentation focus on learning from single A/B experiments, leaving open how to combine information across many experiments to maximize returns rather than merely validate statistical significance. We push this question from two aspects: (i) when a firm holds both experimental and observational data,

what is the best way to combine them; and (ii) given a limited experimentation budget, which experiments should it run to gather that information?

We ask five main questions in this paper. First, what is the value of experimental relative to observational data for learning the reward that guides content generation? Second, how does that value scale with the amount of data available? Third, when both sources are available, what is the most effective way to combine them? Fourth, how do these trade-offs play out over time, when a firm that already holds some history must decide whether to keep experimenting or act on what it knows? And lastly, when the firm can experiment on only a fraction of its content, which tests should it choose to run? These are questions firms face in their day-to-day decisions. They hold historical data and can run experiments to learn more, but each experiment risks leaving money on the table.

To answer these questions, we use the Upworthy archive of randomized headline tests (Matias et al., 2021). Upworthy is a media publisher founded in 2012 with the goal of making socially meaningful stories widely shared. Upworthy is thus a clean example of the problem we study. As a firm in the news market whose business depends on maximizing engagement, it must decide, repeatedly and at scale, how to use its data, the experiments it can run, and the history it accumulates to choose the content that performs best. The archive is well-suited to our questions because it records many A/B tests that the platform has run on news headlines with the click-through rate of every headline variant. Observing the full experimental outcome is the main point of this data, which makes it unique and well suited for our case because it lets us reconstruct what a firm would have learned had it not run experiments and seen only one variant per article, and benchmark that against the experimental ground truth. The data contains information about 150,817 packages which makes it a great source, while there are not many other datasets with this amount of data and the experimental variation that allow us to simulate what would happen if we didn't have experimental data and a valid experimental data for measuring the performance.

Our findings show that both the amount of experimental data and the way it is used govern performance. On the value of experimental data, the modeling choice matters as much as the data itself: a logistic model that targets the ranking objective directly extracts far more from experiments than a model that predicts click-through levels. On how this value scales, the logistic model matches full observational performance using experimental variation on barely 3% of decisions, whereas the click-through prediction model needs more than 80%.

Next, we turn to what is the best way to combine experimental and observational data, where our methodological contribution lies. We first show that naively pooling experimental data into the observational set helps little: under a CTR-prediction loss, combining the two sources tracks the observational baseline and improves only once the experimental share is large. Our central contribution is a method that instead fuses a biased observational estimate with an unbiased experimental one. We map a logistic ranking model trained on the experiments and a click-through prediction model trained on the observational data to a common estimand, the probability that one headline outperforms another, and combine them with a bias-corrected, inverse-variance weight that

leans on the observational estimate when experiments are scarce and shifts toward the experiment as experimental data accumulate. This places our estimator in the tradition of combining experimental and observational data (Kallus, Puli, and Shalit, 2018; Athey, Chetty, and Imbens, 2020; Rosenman et al., 2023), which we extend from scalar treatment effects to high-dimensional textual actions. The combined estimator does better than either source alone, exceeding full observational performance using experiments on just over 1% of decisions and almost always matching or beating the use of experimental data by itself.

We then turn to a more realistic setting in which a firm holds a history of observational data and must decide, test by test, whether to run an experiment or act on what it already knows, and we evaluate different experimentation strategies in this environment. A first observation is that ignoring the historical data and relying on experiments alone performs worse than simply training a model on the historical data to decide. This shows that investing in experiments to obtain high-quality data while discarding the historical data is not a good strategy. On combining the two sources, we again find that pooling them under a CTR-prediction loss is ineffective, failing to improve over using the historical data alone. Our inverse-variance method, by contrast, makes the combination work: by leaning on history when experiments are scarce and incorporating the experiment as it accumulates, it reaches cases where investing in experiments yields a net gain over the historical baseline rather than eroding it.

Finally, we ask how a firm should spend a limited experimentation budget: when it can run full A/B tests on only a fraction of its content, which tests should it choose? We compare several rules for selecting which tests to experiment on, and find that the choice matters most when experiments are scarce: with a short collection window, concentrating experiments on tests where the model expects large differences between headlines (high predicted spread) clearly outperforms spending the budget on low-variation tests, which are cheap to run but uninformative. As the window lengthens, the best rule becomes one that targets tests where the observational and experimental models most disagree about which headline would win. Notably, once we select the served content with our inverse-variance method rather than experiments alone, the differences across selection rules shrink substantially. The findings in this section reinforce our central message: using both data sources together not only improves performance but also makes the firm more robust to how it allocates its experiments.

The remainder of the paper proceeds as follows. Section 2 situates our work in the literatures on unstructured content and generative models in marketing, language-model alignment, and causal inference. Section 3 introduces our two data sources. Section 4 presents the fine-tuning experiment showing that optimizing a generative model on confounded historical outcomes leads it to internalize a spurious signal. Section 5 quantifies the value of experimental relative to observational data and shows how the choice of model shapes it. Section 6 develops our method for combining the two sources and evaluates it as the share of experimental data varies. Section 7 embeds the method in a realistic, time-based simulation of a firm's experimentation decisions. Section 8 studies how to allocate a limited experimentation budget across tests. Section 9 concludes.

2 Related Literature

This paper connects three streams of research: the use of unstructured content and generative models in marketing, the alignment of language models from outcome data, and causal inference for confounded observational data. We discuss each in turn and position our contribution relative to the closest work.

2.1 Unstructured Content and Generative Models in Marketing

Much of what firms choose in marketing is unstructured: the wording of a headline or ad, the copy in a product description, the image in a creative. For instance, Banerjee and Urminsky (2025) study how the language of headlines drives clicks, and Thomas (2026) examine how persuasive text in listings affects outcomes. A long line of work in quantitative marketing treats such text and images as data, using them and language model embeddings (Tang, Yang, and Song, 2024) to explain and predict demand and engagement. Recent examples estimate demand jointly from text and images (Compiani, Morozov, and Seiler, 2025) and represent unstructured content with learned embeddings for prediction and causal analysis (Singh and Zheng, 2023). Ellickson et al. (2024) also use contextual embeddings to predict the effectiveness of novel marketing treatments using previous treatment effect estimations from targeted campaigns, and to decide when experimentation is warranted. These papers establish that unstructured content carries economically meaningful, measurable signal—the premise on which our work builds.

Generative models extend this agenda from *measuring* content to *producing* it. Large language models can now draft the very headlines, descriptions, and messages that firms optimize (Brown et al., 2020), and a growing body of marketing research documents their promise: generating creative ideas (Castelo et al., 2024), supporting marketing research and preference elicitation (Goli and Singh, 2024; Arora, Chakraborty, and Nishimura, 2025), predicting demand and brand choice (Lee, 2024), and distilling capabilities into smaller deployable models (Wang, Sudhir, and Hong, 2024). Yet pretrained models are generic and are not optimized for a firm's specific objective, such as click-through or conversion (Goli and Singh, 2024; Ye, Yoganarasimhan, and Zheng, 2024), which motivates fine-tuning on outcome data. This process requires high-quality data and is typically carried out using various methods, including reward modeling, which we discuss in the following subsection. The cleanest supervision comes from controlled experiments: Ye, Yoganarasimhan, and Zheng (2024) designs an adaptive experimentation to optimize headline CTR when having several content options, and Angelopoulos, Lee, and Misra (2024) fine-tune language models on A/B-test outcomes. Experiments, however, are costly and limited in scope, while firms hold abundant historical logs of content and realized performance. Our paper studies how to use that observational data effectively and how to combine it with experimentation to optimize generated content.

2.2 Aligning Language Models from Outcome Data

Fine-tuning a model to optimize an outcome is known as the alignment problem. The standard toolkit learns from preference or reward signals through reward modeling and reinforcement learning from human feedback (Christiano et al., 2017; Ziegler et al., 2019; Ouyang et al., 2022) or direct preference optimization (Rafailov et al., 2024b), with Askell et al. (2021) being an influential template that we follow in one of our experiments. These methods inherit the quality of the signal they optimize (For a detailed discussion on the central role of reward modeling across alignment algorithms, see Appendix A). When the signal is a historical outcome rather than a clean preference, it can encode factors that have nothing to do with content quality, and the model learns to exploit them—a form of reward misspecification in which optimizing a proxy diverges from the true objective (Gao, Schulman, and Hilton, 2023; Rafailov et al., 2024a). A growing literature documents these failures: models latch onto superficial artifacts such as length or sycophancy (Tien et al., 2022; Chen, Toyer, and Shkurti, 2024; Denison et al., 2024), and more broadly, biases in the supervision data propagate into model behavior (Ntoutsi et al., 2020; Liu, 2023; Fang et al., 2024; Yeh et al., 2024). Our Monday experiment is a controlled marketing instance of exactly this risk: optimizing on confounded engagement scores leads the model to internalize a spurious temporal cue.

A recent line of causal reward modeling seeks to remove such artifacts—for example, by stripping a known nuisance attribute from the learned reward (Chen et al., 2024) or by training rewards to be invariant to confounding attributes (Srivastava et al., 2025; Wang et al., 2025). These approaches typically assume that the reward should not change when the confounding attribute is altered. That assumption is reasonable for human-rated outputs but is often violated in business settings, where confounders like price, seasonality, or audience composition can directionally move the outcome. Loghmani (2025) addresses this challenge using a model-based deconfounding procedure. While effective, this method requires identifying the specific confounders in play (e.g., price) and understanding exactly how they affect the outcome to successfully isolate their effects. In this research, we take an alternative approach that does not rely on knowing the confounding variables or their functional forms. We adopt the firm's perspective; we ask a complementary question: given that observational signals are biased but informative, what is the value of experimental data, and how should the two sources be combined?

2.3 Causal Inference, Experimentation, and Data Combination

Methodologically, our problem is one of learning a causal mapping under confounding. Econometrics and machine learning offer a rich toolkit: double/debiased machine learning, which orthogonalizes treatment and outcome via residualization and cross-fitting (Chernozhukov et al., 2018); deep networks for semiparametric estimation and inference (Farrell, Liang, and Misra, 2021); instrumental-variable and conditional-moment estimators for endogenous settings (Bennett, Kallus, and Schnabel, 2019; Dikkala et al., 2020); and results on regularization and endogeneity in high dimensions (Zou and Zhang, 2009; Fan and Liao, 2014). We borrow the residualization logic of double machine

learning, but our "treatment" is high-dimensional text, so we residualize the outcome rather than the treatment embedding. A parallel literature adapts causal inference to text and language models, estimating or controlling for the effects of unstructured treatments (Singh and Zheng, 2023; Wu et al., 2024; Zhang, Wang, and Dhillon, 2024; Modarressi, Spiess, and Venugopal, 2025; Xu et al., 2025) and developing econometric frameworks for LLM-generated data (Ludwig, Mullainathan, and Rambachan, 2025); our goal differs in that we seek to rank and optimize content and to fuse experimental with observational supervision.

Our work also draws on the economics of experimentation. A large literature studies the design and value of A/B tests (Feit and Berman, 2019; Miller and Hosanagar, 2020; Goldfarb, Tucker, and Wang, 2022; Quin et al., 2024), methods for efficient measurement (Johnson, Lewis, and Nubbemeyer, 2017), and empirical-Bayes pooling across many tests (Azevedo et al., 2019). Most directly, our combination method belongs to the literature on fusing experimental and observational data, which corrects the bias of abundant observational estimates using a smaller experimental sample (Kallus, Puli, and Shalit, 2018; Athey, Chetty, and Imbens, 2020; Rosenman et al., 2023). Our estimator is a bias-corrected, inverse-variance combination in this tradition; relative to that work, which targets average treatment effects on a scalar outcome, we operate over high-dimensional textual actions, fuse a logistic ranking model with a CTR-prediction model by mapping both to a common win-probability estimand, and deploy the result inside a generative content-optimization pipeline.

Finally, related to our strategic experimentation approach, recent literature also explores how to allocate limited experimental budgets when observational data is available. For instance, Ellickson et al. (2024) address a scenario where previous campaigns and targeting data are used to predict conditional average treatment effects (CATE), allowing the model to decide whether to rely on prior data or run a new experiment based on the content's novelty. However, their setting differs from ours. While their use of targeted treatments and controls allows them to predict CATE, we study a scenario where historical data is more general, encompassing broader information from previous deployments. Nevertheless, we test the concept of novelty in our setting to evaluate its effectiveness. In a complementary vein, Gao et al. (2026) focus on active experimentation to determine which users to acquire for testing a binary treatment, though their setting does not involve high-dimensional treatments.

3 Data

Our empirical analysis draws on two complementary data sources, each chosen to isolate a distinct part of the problem we study. The first, a corpus of question–answer pairs from the Academia Stack Exchange, is purely observational. We use it to demonstrate how fine-tuning a generative model on historical engagement signals can lead the model to internalize correlations that reflect exposure rather than quality. The second, the Upworthy Archive of headline A/B tests, records the outcomes of randomized experiments. Because the experimental ground truth is observed, we can construct

counterfactual scenarios in which a firm holds only observational data, measure what is lost relative to the experimental benchmark, and simulate how alternative experimentation strategies would have performed in real time. Together, the two datasets let us study both the pitfalls of relying on observational data and the value of experimentation for optimizing unstructured marketing content.

3.1 Stack Exchange

To illustrate that fine-tuning a language model directly on observational data can be problematic, with the model internalizing spurious correlations present in the data, we replicate a setup similar to that of Askell et al. (2021) using data from the Academia Stack Exchange. The raw corpus contains 104,426 question–answer pairs. We retain only questions with more than one answer, which yields 82,737 answer instances and allows us to form within-question comparisons between answers. To limit memory usage during training, we further restrict attention to questions and answers shorter than 180 words, leaving 33,194 answers across 14,319 questions.

In the original study, the answer score—the net of up- and down-votes the community assigns to an answer—serves as a proxy for answer quality and as the supervisory signal for fine-tuning a question-answering model. While scores are informative about which answer a community found helpful, they are not the outcome of a randomized experiment and may instead reflect user engagement patterns unrelated to content quality. Answers posted earlier, for example, accumulate more views and therefore more votes. We focus on one such confounder: periodicity in platform engagement across the days of the week.

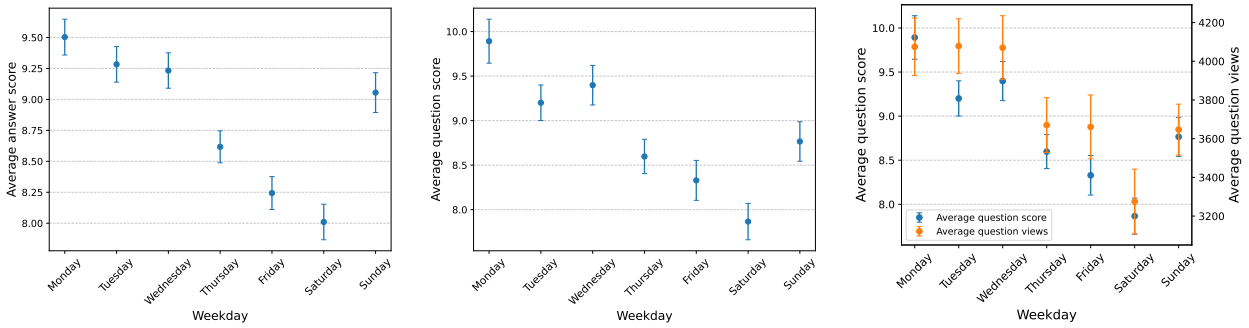
To document this pattern, we examine average scores by weekday. Figure 1a shows that answers posted on Mondays receive markedly higher scores than those posted on Fridays. A natural concern is that this gap is causal, reflecting genuine differences in writing quality across days. The same weekday pattern, however, appears in the scores of the questions themselves (Figure 1b), which is difficult to attribute to answer quality and instead points to broader engagement trends.

To probe the mechanism, we use question views as a proxy for user exposure. Since the dataset does not record view counts at the answer level, we cannot measure exposure for individual answers directly and instead examine views at the question level. Figure 1c plots question views and scores together, and the two track each other closely. This co-movement suggests that the observed weekday patterns are driven more by fluctuations in user activity, and hence exposure, than by differences in content quality.

These patterns make the Academia Stack Exchange a useful testbed: the scores embed a temporal confounder that is correlated with engagement but unrelated to quality. In Section 4 we exploit this structure to ask whether fine-tuning a question-answering model directly on these scores leads it to reproduce, and even amplify, the temporal artifact in its generated answers.

3.2 Upworthy

Our second data source is the public Upworthy Research Archive, a release of real-world clickability A/B tests drawn from the editorial workflow of Upworthy, a U.S. media publisher known for



(a) Average answer score by weekday

(b) Average question score by weekday

(c) Average question score and views by weekday

Figure 1: Weekly temporal patterns in Stack Exchange engagement. User scores and views exhibit strong weekday effects, with higher engagement early in the week.

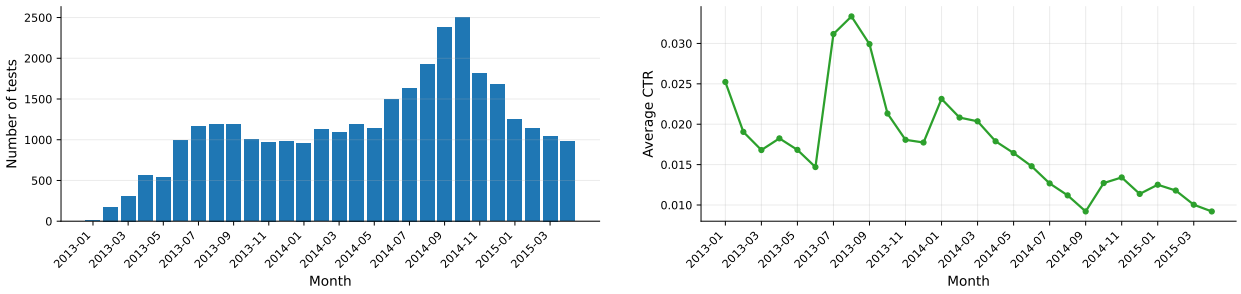
optimizing the headlines and images that accompanied its stories (Matias et al., 2021). Before publishing a story, Upworthy's editors routinely tested several candidate *packages*—combinations of a headline and an accompanying image, or "eyecatcher"—by serving them at random to incoming site visitors and recording how each performed. Because traffic was randomized across the variants within a test, the resulting click data have the structure of a controlled experiment. This is the feature we exploit throughout the paper.

We use the "packages" release of the archive, which records one row per headline variant (experiment arm). Each row reports the candidate headline text, a short background abstract of the underlying story (the *lede*), the test the variant belongs to, and the variant's realized traffic: the number of impressions it received and the number of clicks it generated. The raw archive comprises 150,817 rows spanning January 2013 to April 2015. In aggregate, the variants accumulated 538,272,878 impressions and 8,182,674 clicks. Across variants, the click-through rate (CTR)—clicks divided by impressions—has a mean of approximately 1.58% and a median of approximately 1.25%, the right skew reflecting a long tail of high-performing headlines.[1]

Temporal patterns. Figure 2 summarizes the volume and performance of testing over time. Panel (a) plots the number of tests conducted each month, and panel (b) the average CTR per month. Testing activity grew substantially over the sample: there are 1,160 tests per month on average, peaking at 2,508 tests in October 2014. Average CTR exhibits a pronounced and non-monotonic temporal pattern—declining over the first several months, rising sharply, and then trending downward through the remainder of the sample. Averaging across months, the monthly mean CTR is 1.75%, ranging from a high of 3.33% in August 2013 to a low of 0.92% in September 2014. These swings are large relative to the level of CTR itself, and they indicate that headline performance cannot be assessed in isolation from when a test was run. We return to the implications of this

[1]Following publication of the archive, Matias et al. (2021) documented a likely randomization failure affecting roughly 22% of tests fielded between June 25, 2013 and January 10, 2014, attributed to a caching issue, and released a `randomization_imbalance_risk` flag identifying the affected tests. [TODO: state how you handle this—e.g., drop flagged tests, or verify results are robust to excluding this window.]

temporal structure when we design experimentation strategies that play out over time in Section 7.



(a) Number of tests per month

(b) Average CTR

Figure 2: Monthly temporal patterns in Upworthy news-headline experimentation data

The stakes of headline selection. The temporal variation above could in principle reflect compositional changes in which stories were tested rather than the causal influence of headlines. To establish that headline choice carries real economic weight—and that the temporal pattern is not an artifact of a few unusually good or bad variants—we compare the clicks Upworthy actually realized by running its tests against several counterfactual selection rules. For each test, we ask what would have happened had the firm bypassed experimentation and committed all of that test's traffic to a single arm: the best-performing headline, the second-best, or the worst.

Figure 3 reports these counterfactuals month by month. The ordering is stable throughout the sample: always selecting the best arm dominates, always selecting the second-best still outperforms running the test, and always selecting the worst arm trails. Relative to the clicks obtained by running every test, committing to the best arm yields 27% more clicks and committing to the second-best yields 9% more, while committing to the worst arm yields 26% fewer. Two points follow. First, the spread between the best and worst arms—more than fifty percentage points of clicks—shows how consequential headline selection is. Second, the fact that even the second-best arm beats running the test quantifies the opportunity cost of experimentation: splitting traffic across all arms, including weak ones, sacrifices clicks that a firm confident in a good arm could have captured. This tension between the information experiments provide and the traffic they cost motivates the central questions of the paper.

4 The Monday Experiment: Internalizing Spurious Correlation

When a firm optimizes unstructured content by fine-tuning a generative model on historical outcomes, it implicitly treats those outcomes as a reward. If the outcome is confounded—driven in part by factors other than content quality—the model can learn to reproduce the confounder rather than the quality it is meant to capture. We call this reward misspecification. This section provides a controlled demonstration: fine-tuning a question-answering model on observational engagement scores leads it to internalize, and even amplify, a temporal artifact that is correlated with engagement but

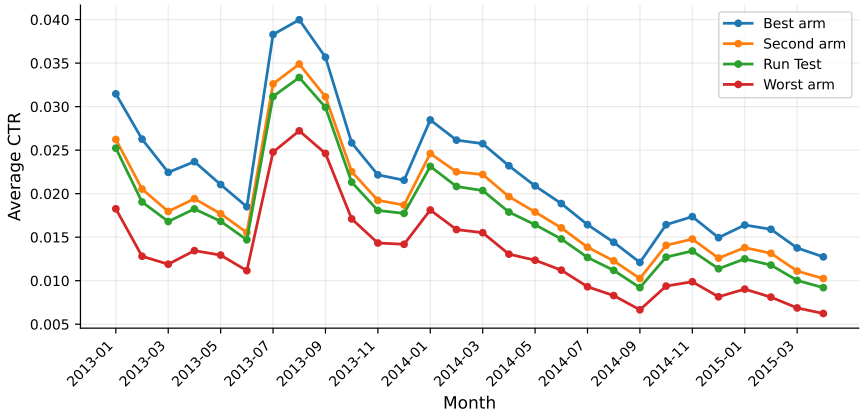


Figure 3: Compare CTR performance of running all tests, vs the case that we always chose the best headline, second headline, and the worst headline for each test

unrelated to answer quality.

Our design follows Askell et al. (2021), who fine-tune a question-answering model in stages. One stage, Preference Model Pre-training (PMP), uses historical community data: answer scores serve as preference labels, and the model is trained to prefer the higher-scored of two answers to the same question. In their pipeline, PMP is followed by fine-tuning on human feedback, which realigns the model's preferences with human judgments of quality. We ask what happens in the common situation where such human feedback is unavailable and a firm fine-tunes on observational data alone. As established in Section 3.1, scores on the Academia Stack Exchange carry a weekday confound: answers posted earlier in the week receive systematically higher scores, driven by exposure rather than quality (Figure 1).

To make this spurious signal legible and precisely measurable, we attach a short content marker to answers based on the day they were posted. Mondays have the highest average answer scores among weekdays and Fridays among the lowest (Figure 1a), so we prepend ‘‘`Happy Monday!`’’ to every answer posted on a Monday and ‘‘`Happy Friday!`’’ to every answer posted on a Friday. These phrases are deliberately uninformative about answer quality; they serve only as a tag correlated with engagement through the weekday confound. If a model fine-tuned on score-based preferences generates these tags at rates above what the data distribution implies, that is direct evidence it has latched onto the confound rather than quality. The design's advantage is that the otherwise diffuse notion of "internalizing a spurious correlation" reduces to counting a handful of tokens in the model's output.

We compare three models. The *base model* is the pretrained language model, which has not seen our data. The *SFT model* is supervised-fine-tuned to reproduce the answers in our corpus; this adapts the base model to the domain's content and formatting and, crucially, establishes the rate at which weekday markers appear in the data itself. The *DPO model* initializes from the SFT checkpoint and is further trained on score-based preference pairs via Direct Preference Optimization, an instance of the reward-based fine-tuning a firm would use to steer generations

toward higher-scoring content. Comparing DPO to SFT isolates the effect of optimizing for the confounded score: SFT reflects the data distribution, so any systematic shift under DPO reflects what preference optimization adds.

We partition the questions into three disjoint sets: 5,000 for supervised fine-tuning, 3,000 held out for evaluation, and the remainder for preference-based fine-tuning. Within the preference set, we form ordered answer pairs for each question by labeling the higher-scored answer as preferred, capping the number of pairs per question at ten to avoid over-weighting questions with many answers. This yields 11,886 pairs. The weekday confound is present but mild: among pairs containing exactly one Monday answer, the Monday answer is preferred in 958 cases and rejected in 887; among pairs containing exactly one Friday answer, the Friday answer is preferred in 859 and rejected in 965. A model that merely reproduced the data would inherit only this modest skew; the question is whether preference optimization magnifies it.

We then generate answers for the 3,000 held-out questions and measure how often the markers appear. Table 1 reports the results. The base model almost never produces these tokens ("Happy" in 1.1% of generations), confirming they are artifacts of our corpus rather than the pretrained model. The SFT model generates "Happy" in 24.3% of answers, close to the roughly 2/7 share implied by the weekday distribution of the data (generation temperature is set to one), confirming that SFT reproduces the empirical frequency. The DPO model departs sharply from this benchmark, and in the direction the confound predicts: it generates "Happy" in 35.4% of answers and "Monday" in 14.8%, increases of 11.1 and 3.4 percentage points over SFT, while "Friday" falls from 8.9% to 2.9%. Optimizing for score thus pushes the model toward the high-scoring (Monday) marker and away from the low-scoring (Friday) one. Notably, the generation shift is large relative to the mild skew in the training pairs—the near-disappearance of "Friday" in particular—indicating that preference optimization does not merely transmit the spurious signal but amplifies it.

To assess whether these shifts are systematic rather than an artifact of a particular seed, we estimate each rate over 25 runs per model (five fine-tuning seeds by five generation seeds; see the setup below) and run independent two-sample t-tests comparing DPO to SFT. Both effects are highly significant: the increase in "Monday" usage ($p = 4.1 \times 10^{-4}$) and the decrease in "Friday" usage ($p = 1.6 \times 10^{-31}$). The model has internalized and amplified a temporal signal that, by construction, carries no information about answer quality.

Table A1 in Appendix Section B offers qualitative evidence that the pipeline behaves as intended and that the shift is not a training pathology. For a single held-out question, the base model returns a generic answer in Markdown (e.g., `**` for bold), reflecting its pretraining conventions. The SFT model adopts the domain's formatting (HTML tags) and tone, indicating successful domain adaptation. The DPO model goes further, displaying features associated with high-scoring answers in the corpus—richer formatting such as `<strong>` and external references—alongside the weekday marker. This is consistent with the model having learned which surface features correlate with high scores, mixing plausibly quality-related ones (formatting, references) with the spurious weekday tag.

This experiment makes concrete a risk that arises whenever firms optimize unstructured content against historical outcomes. Without an account of the causal structure of the data, preference optimization can mis-specify the reward and steer generations toward features that merely correlate with success. The remainder of the paper turns to data that does support causal claims—randomized headline experiments—which we use to quantify the value of experimental over observational data and to study how the two sources can best be combined.

Models and fine-tuning. We use the 360M-parameter `SmolLM2-Instruct` model (Allal et al., 2025) as the base and fine-tune in two stages. Supervised fine-tuning uses each answer as the assistant response to its question, trained for one epoch (batch size 8, learning rate 2×10^{-4}, 8-bit AdamW) with LoRA (Hu et al., 2022) of rank 16 and dropout 0.1, under a custom prompt template with a 512-token limit. DPO initializes from the SFT checkpoint and trains on the preference pairs with $\beta = 0.1$ for up to four epochs (batch size 8, LoRA rank 8, maximum prompt and completion lengths of 256 and 512 tokens).

Evaluation and randomness. All evaluations use the same 3,000 held-out questions. Because the base model is fixed, we vary only the generation seed, using five seeds. The SFT and DPO models are subject to randomness in both fine-tuning and generation, so we train each with five seeds for head initialization and generate from each trained model with five generation seeds, giving 25 runs per model and 3,000 generations per run. Reported rates are means across runs with standard errors in parentheses.

Table 1: Mean percentage (standard error) of generations containing "Happy" and "Monday" across 5 generation seeds for the base model, and 25 runs (5 fine-tuning seeds × 5 generation seeds) of SFT and DPO fine-tuning. DPO fine-tuning significantly amplifies the spurious weekday signal.

Model	Generations per Run	Num. Runs	With "Happy" (%)	With "Monday" (%)	With "Friday" (%)
Base Model	3000	5	1.13 (0.04)	0.07 (0.02)	0.10 (0.02)
SFT Model	3000	25	24.32 (0.14)	11.45 (0.20)	8.88 (0.13)
DPO Model	3000	25	35.37 (1.14)	14.84 (0.82)	2.89 (0.16)

5 Value of Experimental and Observational Data

The Upworthy archive lets us put a price on experimentation. Because every test records the click-through rate (CTR) of each headline variant, we observe the full experimental outcome and can ask what a firm would have lost had it held only observational data—a single headline per story, with no within-story variation. This section addresses two questions. First, what is the value of experimental data relative to observational data for learning which headline performs best? Second, holding the data fixed, how much does the way we use it—the loss we train against—matter? We find that both choices are consequential, and that the modeling choice can matter as much as the data itself.

We restrict attention to headline variation, holding the image fixed: two headlines are comparable only if they appear in the same test with the same image (eyecatcher). This yields 17,141

tests, with a mean of 4.37 headline variants per test (standard deviation 1.26, range 2–17, median 4). Framing the problem as preference learning, we form every unordered headline pair within a test-image, giving 144,242 pairs. For each pair we compute a two-proportion z-test of the difference in CTR; we apply no significance filter when constructing the data, though for reference 36.9% of pairs are significant at $p < 0.10$, 28.9% at $p < 0.05$, and 17.3% at $p < 0.01$.

We split the data at the test-image level into 60% training (10,609 tests), 20% validation (3,536 tests), and 20% test (3,536 tests). To prevent leakage, we discard any validation or test pair in which either headline appears anywhere in the training set, checking across all four join directions; this removes 10,973 test pairs ($\approx 38\%$) and 8,915 validation pairs ($\approx 31\%$) and guarantees zero headline overlap between training and evaluation.

Our goal is a model that, given a news abstract, ranks candidate headlines by their CTR. We represent each headline in context by embedding the concatenated abstract and headline—the text a visitor actually sees—and compare three training pipelines built on top of these embeddings. The first is a *logistic* model trained on the experimental headline pairs, which targets the comparative question directly: given two headlines, which has the higher CTR? The second is a *ridge CTR-prediction* model trained on all experimental observations, which learns a mapping from embedding to CTR and ranks headlines by predicted CTR. The third is the same ridge CTR-prediction model trained only on *observational* data—one variant per test-image—mimicking a firm that never ran experiments. We evaluate all models by ROC-AUC on the held-out test pairs that are statistically distinguishable ($p < 0.05$), the pairs for which a correct ranking is well defined.

Table 2 reports the results across embedding models, and two patterns stand out. First, the logistic model is best in every case. Averaging across embedding models, it improves ROC-AUC by 0.067 over the CTR-prediction model trained on the same experimental data; equivalently, the CTR-prediction model scores about 8.5% lower than the logistic model. The reason is that the logistic loss optimizes the ranking objective we ultimately care about, rather than the harder intermediate task of predicting CTR levels. Second, for the CTR-prediction model, simply adding experimental variation buys little: moving from observational to full experimental data raises ROC-AUC by only 0.005 on average. When the data are used through a CTR-prediction loss, the experimental variation that observational data lacks is largely wasted. The lesson is that how the data are used can matter as much as how much experimental variation they contain.

This motivates a better use of the CTR-prediction model. A natural idea, in the spirit of double/debiased machine learning (Chernozhukov et al., 2018), is to residualize: partial out the influence of common factors from treatment and outcome before relating them. In our setting the "treatment" is the headline text, which lives in a high-dimensional embedding space, and residualizing on it is problematic—the embedding may not be fully purged of confounding information, and projecting it out can destroy structure useful for prediction. We therefore residualize only the outcome. We predict each headline's CTR from its *abstract* embedding alone, using a two-layer neural network and selecting the epoch with the best validation CTR MSE, and subtract this prediction. The residualized outcome thus reflects the headline's contribution net of the story it accompanies.

To keep this estimate honest and avoid using an observation to residualize itself, we cross-fit. We split the combined training and validation data into two folds. We fit the abstract-to-CTR model on the first fold, tuning hyperparameters on its validation portion, and use it to residualize the second fold; we then reverse the roles of the two folds. The final ridge model is trained on the residualized CTRs assembled from both folds. We apply this procedure to both the full-experimental and the observational CTR-prediction tasks.

Residualization helps, and it helps the experimental case more, which sharpens the value of experimental variation. For observational data, residualization raises ROC-AUC by 0.017 on average (a 2.4% relative gain), and it does so for every embedding model; for experimental data it raises ROC-AUC by 0.020 (2.8%). As a result, the gap between full-experimental and observational CTR prediction widens from 0.005 without residualization to 0.007 with it: once the outcome is residualized, the additional experimental variation finally translates into better rankings. Even so, the residualized CTR-prediction model still trails the logistic model trained on the same experimental data, reinforcing that targeting the ranking objective directly is the single most effective choice. Figure 4 displays these comparisons across embedding models.

Table 2: Test ROC-AUC measures across different embedding models and training pipelines.

	Logistic Loss	**CTR Prediction Loss**		**Residualized CTR Loss**	
Embedding Model	Experimental	Experimental	Observational	Experimental	Observational
pythia-70m	0.688	0.631	0.632	0.660	0.650
pythia-160m	0.722	0.659	0.660	0.682	0.680
pythia-410m	0.764	0.700	0.691	0.731	0.719
pythia-1b	0.788	0.721	0.711	0.746	0.732
pythia-2.8b	0.791	0.733	0.726	0.751	0.748
pythia-6.9b	0.797	0.729	0.729	0.759	0.750
pythia-12b	0.808	0.742	0.727	0.752	0.739
text-embedding-3-small	0.858	0.768	0.765	0.780	0.773
text-embedding-3-large	0.882	0.813	0.810	0.811	0.814

6 Experimental and Observational Data Combination

The previous section contrasted two extremes: a firm with full experimental data and a firm with none. In practice, firms occupy the middle ground—they experiment on a sample of their decisions and rely on historical, observational records for the rest. This section studies that intermediate regime. Suppose a firm has run experiments on a fraction x of its test-images and holds only observational data (one variant per test) for the remaining $1-x$. We ask three questions. How does the amount of experimental data affect each method's performance? How much experimental data is needed to match what full observational data alone delivers? And when experimental data are scarce, what is the best way to combine the two sources?

To answer the first two questions, we vary the experimental fraction x over a fine grid—from 1% to 10% in steps of one percentage point, then $20, 30, \ldots, 90\%$, then 90% to 99% in steps of one—

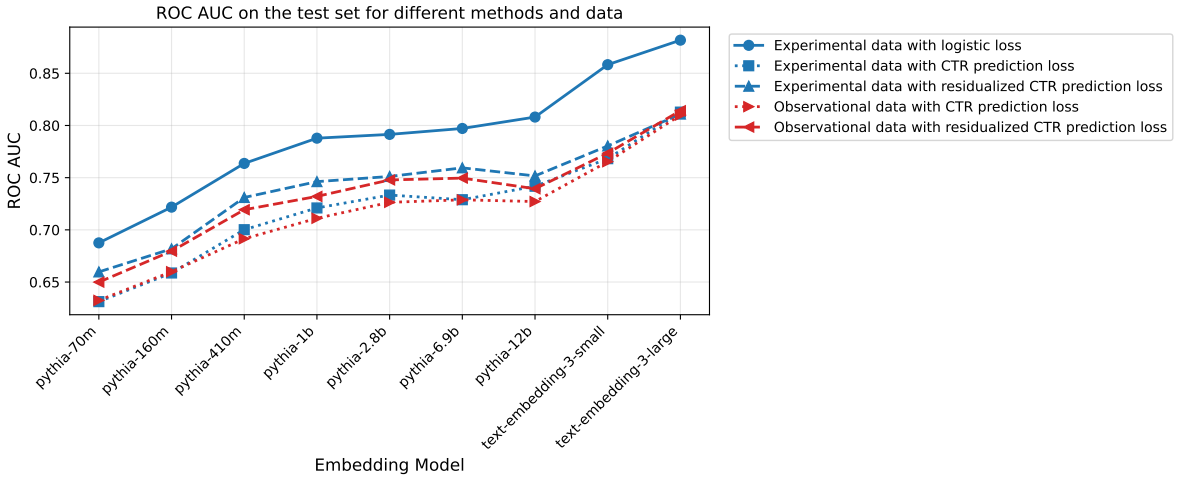


Figure 4: Test ROC-AUC performance across different embedding models and training pipelines. The plot compares experimental and observational data using standard and residualized Click-Through Rate (CTR) prediction losses alongside a baseline logistic loss.

and, at each x, train five methods: (a) experimental data with the logistic loss; (b) experimental data with the CTR-prediction loss; (c) experimental data with the residualized CTR-prediction loss; (d) observational data with the CTR-prediction loss; and (e) observational data with the residualized CTR-prediction loss. Methods (a)–(c) use only the experimental fraction, while (d)–(e) use only the observational complement. We evaluate as before, by ROC-AUC on statistically distinguishable held-out pairs, averaging over five seeds.

Figure 6 reports the results for the `text-embedding-3-large` representation. The sample efficiency of the three experimental methods differs sharply. With the logistic loss, only just above 3% of the data is needed to match the performance of full observational data. With the CTR-prediction loss and no residualization, more than 80% of the paired data is required to reach the same bar—again showing that experimental variation contributes little when funneled through CTR prediction. Residualizing the outcome closes much of this gap: the residualized CTR-prediction model surpasses full observational data with about 20% of the experimental sample. These crossing points quantify how much the choice of loss, not just the volume of experimental data, governs how efficiently a firm converts experiments into better rankings.

The figure also shows that residualization aids the experimental methods from early fractions onward, with a visible gap opening by roughly 20%. Its effect on the observational methods is more delicate and embedding-dependent: for the highest-dimensional representation, residualizing a single observation per test can absorb much of the signal and overfit, so the gains there are smaller than for lower-dimensional embeddings.

We now turn to the third question: given x% experimental data, can we do better by using *both* sources rather than discarding one? For two CTR-prediction models the answer is immediate—since experimental and observational data share the same target, we simply pool them. The difficulty arises when we want to pair the logistic loss, which is the most effective use of experimental

data, with a CTR-prediction model on the observational data. The two have different objectives: the logistic model maximizes a separation (log-likelihood) objective over pairs, while the CTR-prediction model minimizes squared error in CTR levels. Their parameters are not comparable, so naively sharing weights or summing the two losses performs poorly.

We therefore combine the two models at the level of their *predictions* rather than their parameters, exploiting a bias–variance complementarity between them. The observational CTR model yields a biased estimate of which headline wins—biased because it rests on non-experimental variation—but its bias is offset by low variance, and that variance falls as historical data accumulate. The experimental logistic model is unbiased by design, but with little experimental data its estimates are high-variance. A combination that leans on the observational model when experiments are scarce and shifts toward the experimental model as experiments accumulate can dominate either one alone.

Realizing this requires putting the two models on a common footing, since their native outputs differ. We fix the estimand as the probability that one headline outperforms another, $\Pr(a \succ b)$. The logistic model delivers this probability, and a variance around it, directly. The CTR-prediction model instead returns a point CTR estimate and a standard error for each headline; assuming these are approximately Normal, we draw Monte-Carlo samples for the two arms and compute the implied probability that a beats b, together with its variance. Both branches now produce an estimate of the same quantity, which we denote $\hat{\theta}_U$ (unbiased, from the experiment) and $\hat{\theta}_B$ (biased, from observation).

We combine them as a convex combination governed by a single weight $w \in [0, 1]$,

$$\hat{\theta}_{\text{new}} = w\,\hat{\theta}_B + (1 - w)\,\hat{\theta}_U, \tag{1}$$

and choose w to minimize the mean squared error of $\hat{\theta}_{\text{new}}$. Because $\hat{\theta}_U$ is unbiased, the bias of the combination is $w\,\text{Bias}(\hat{\theta}_B)$, and because the two estimators are built from disjoint data, they are uncorrelated. Hence

$$\text{MSE}(\hat{\theta}_{\text{new}}) = \underbrace{w^2\,\text{Bias}(\hat{\theta}_B)^2}_{\text{bias}^2} + \underbrace{w^2\,\text{Var}(\hat{\theta}_B) + (1 - w)^2\,\text{Var}(\hat{\theta}_U)}_{\text{variance}}. \tag{2}$$

The squared bias grows quadratically in w while the variance is a quadratic with an interior minimum, so the MSE is convex (U-shaped) in w with a unique minimizer. Setting the derivative to zero yields

$$w^\star = \frac{\text{Var}(\hat{\theta}_U)}{\text{Bias}(\hat{\theta}_B)^2 + \text{Var}(\hat{\theta}_B) + \text{Var}(\hat{\theta}_U)}. \tag{3}$$

The weight behaves as the application demands. When the experimental estimator is very noisy—few experiments, so $\text{Var}(\hat{\theta}_U)$ is large—$w^\star \to 1$ and the combination leans on the low-variance observational model. When the observational estimator is badly biased—$\text{Bias}(\hat{\theta}_B)^2$ large—$w^\star \to 0$ and the combination defers to the unbiased experiment. As the experimental sample grows, $\text{Var}(\hat{\theta}_U)$

shrinks and the weight slides smoothly from the observational toward the experimental estimate.

Algorithm 1: Bias-corrected inverse-variance combination (IVW)

Input: observational data $\mathcal{D}_O$; experimental pairs $\mathcal{D}_E$; evaluation pairs $\mathcal{P}$
Output: combined win-probability $\hat{\theta}_{\text{new}}(a, b)$ for each pair in $\mathcal{P}$

Fit CTR-prediction model $\hat{m}_O$ on $\mathcal{D}_O$ `// biased, low-variance branch`
Fit logistic model $\hat{g}_E$ on $\mathcal{D}_E$ `// unbiased, high-variance branch`
`// Estimate the bias of the observational branch by cross-fitting`
Partition $\mathcal{D}_E$ into halves $\mathcal{D}_E^1, \mathcal{D}_E^2$
for $k \in \{1, 2\}$ **do**
    Train a logistic model $\hat{g}_{-k}$ on the other half $\mathcal{D}_E^{3-k}$
    For each pair in $\mathcal{D}_E^k$, form p^O (from $\hat{m}_O$) and p^E (from $\hat{g}_{-k}$)
    $\widehat{b}_k \leftarrow \text{mean}(p^O - p^E)$ over those pairs
$\widehat{b} \leftarrow \frac{1}{2}(\widehat{b}_1 + \widehat{b}_2)$ `// estimated bias of` $\hat{\theta}_B$
foreach *pair* $(a, b) \in \mathcal{P}$ **do**
    $(\hat{\theta}_U, \widehat{\text{Var}}_U) \leftarrow$ win-probability and variance from $\hat{g}_E$
    $(\hat{\theta}_B, \widehat{\text{Var}}_B) \leftarrow$ Monte-Carlo win-probability and variance from $\hat{m}_O$ `// CTR ~ Normal`
    $w^\star \leftarrow \widehat{\text{Var}}_U / (\widehat{b}^2 + \widehat{\text{Var}}_B + \widehat{\text{Var}}_U)$
    $\hat{\theta}_{\text{new}}(a, b) \leftarrow w^\star \, \hat{\theta}_B + (1 - w^\star) \, \hat{\theta}_U$
return $\{\hat{\theta}_{\text{new}}(a, b) : (a, b) \in \mathcal{P}\}$

The variances in (3) come directly from each model, but the bias of the observational estimator must be estimated. Since our estimand is a win probability, we estimate the bias as the average difference between the win probabilities assigned by the observational model and by the unbiased experimental model. To keep this estimate honest, we cross-fit: we fit the observational CTR model on the historical data, split the experimental data in half, train a logistic model on one half, and use the other half to compare the two models' win probabilities; we then repeat with the halves reversed and average the two bias estimates. Algorithm 1 summarizes the full procedure, and Figure 5 depicts the idea: two branches that estimate the same win probability with complementary bias–variance profiles, fused by an inverse-variance weight.

Figure 7 reports performance across experimental fractions, with the full range in panel 7a and a zoom below 20% in panel 7b. Pooling experimental and observational data under a plain CTR-prediction loss helps little: it tracks the observational baseline and improves only beyond roughly 80% experimental data. Pooling under the residualized loss is better, starting near the all-observational level and improving once the experimental fraction passes about 20%. Both pooled methods, however, remain below the experimental-only logistic model whenever at least 3% of the data are experimental, so simply adding observational data to a CTR objective does not recover what the logistic loss achieves. Our proposed combination resolves this. By fusing the logistic and observational estimates through the bias-corrected weight, it almost always matches or exceeds the experimental-only logistic model, with the largest gains precisely where experimental data are

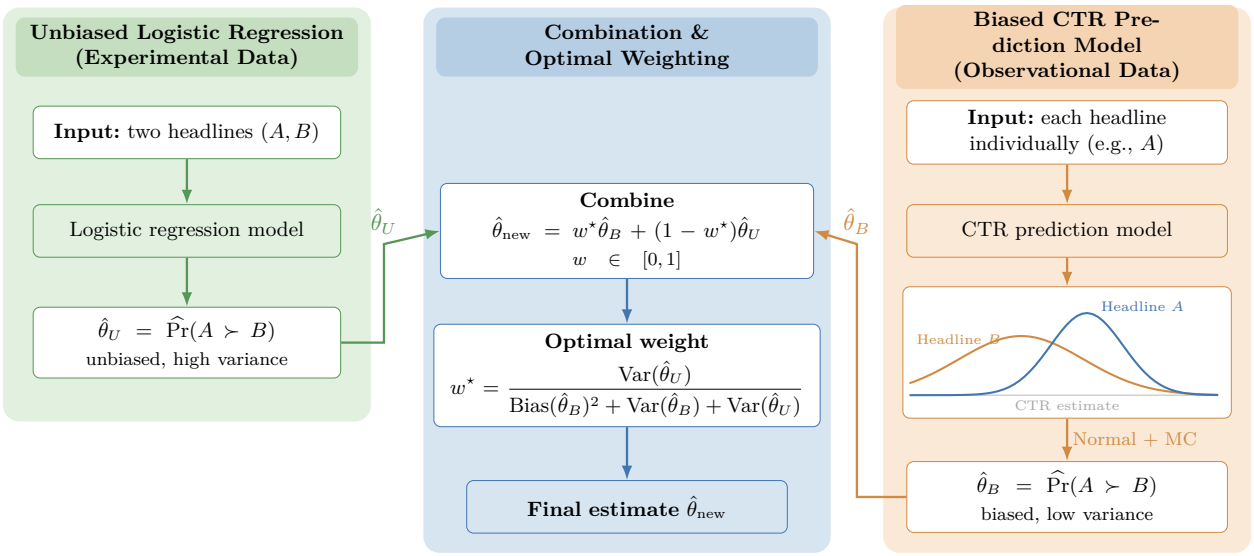


Figure 5: Combining a biased, low-variance observational estimate ($\hat{\theta}_B$) with an unbiased, high-variance experimental estimate ($\hat{\theta}_U$). Each branch is mapped to the same estimand, the win probability $\Pr(A \succ B)$—the observational branch by drawing Monte-Carlo samples from the two headlines' Normal CTR estimates—and the two are fused by the inverse-variance weight $w^\star$ of Equation (3). Scarce experiments make $\hat{\theta}_U$ noisy and push $w^\star \to 1$; abundant experiments shrink its variance and push $w^\star \to 0$.

scarcest. It surpasses the full-observational benchmark with just over 1% of the data experimental—earlier than any other method—showing that the right way to use a small experiment is not to use it in isolation, but to anchor it with the observational data the firm already holds.

7 Performance in a More Realistic Setting with Time

The combination experiment of Section 6 split the data randomly into an experimental portion and an observational portion. That design isolates the value of combining the two sources, but it abstracts away from how a firm actually accumulates data: over time, one decision after another. In this section we embed the same trade-off in a realistic temporal setting. A firm begins with several months of historical, observational records and must then decide, test by test as they arrive, whether to run an experiment—learning the full set of arm CTRs at the cost of splitting traffic across good and bad headlines—or to commit all of a test's traffic to a single headline chosen from what it has already learned. The question is how to convert a fixed stock of history, plus the option to experiment going forward, into the most clicks.

Data construction. To make time meaningful, we re-clean the archive so that a cutoff date cleanly partitions tests into those already concluded and those not yet run. We retain only tests in which every headline arm shares the same start date, and we reduce each test to a single image so that the only object of choice is the headline, consistent with our focus and with the absence of image-content features: if a test has one eyecatcher we keep it, and if it has several we keep the eyecatcher carrying the most headline arms. We require at least two headlines per test. This

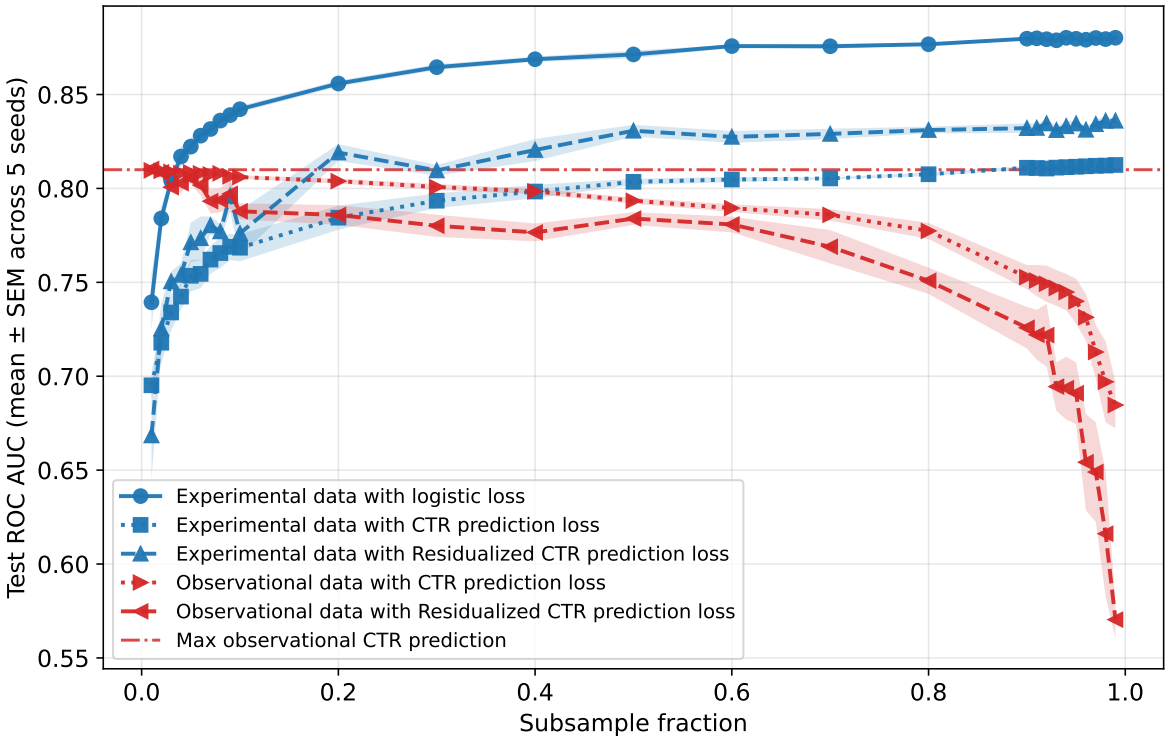
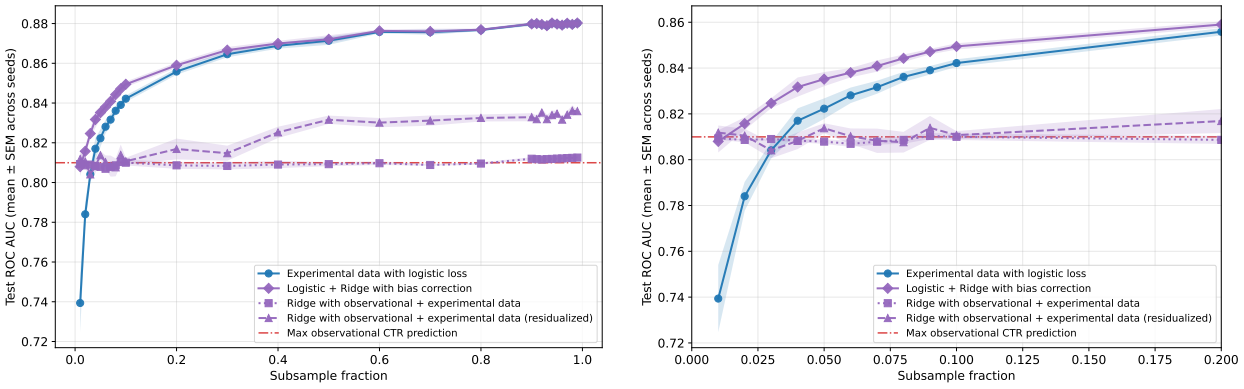


Figure 6: Test ROC AUC across varying subsample fractions for experimental data (logistic, CTR, and residualized CTR losses) and the remaining observational data (CTR and residualized CTR losses), compared against the maximum observational CTR baseline.



(a) Full range of subsample fractions

(b) Subsample fraction ≤ 0.20

Figure 7: Test ROC AUC across varying subsample fractions comparing experimental data alone (logistic loss) and combined observational/experimental methods (logistic + ridge with bias correction, standard ridge, and residualized ridge), evaluated against the maximum observational CTR baseline.

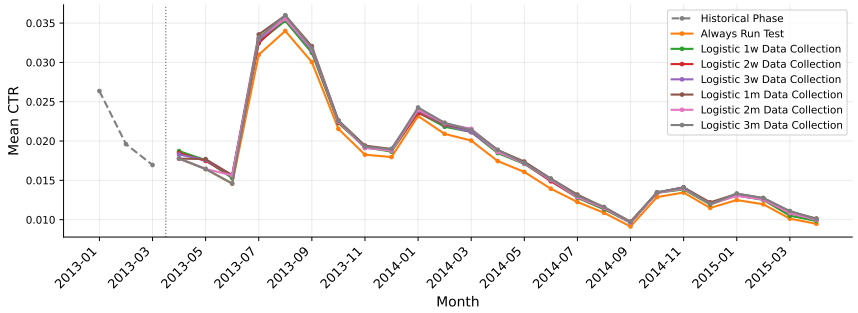
procedure yields 98,739 test-headline observations from the original 150,817.

Simulation setup. We treat the first three months of the sample as the firm's historical record. To mimic a firm that did not experiment during this period, we draw a single headline at random for each test in the window and record only that headline's realized CTR; this is the observational data on which a historical model can be trained. From the end of the historical window onward, the firm faces each test in turn and must choose one of two actions. It can *run the test*, observing the CTR of every arm and adding that comparison to the data available for training, or it can *commit* to a single headline and serve only that. Running a test is informative but costly: as Section 3.2 showed, splitting traffic across all arms forgoes clicks that a firm confident in a strong headline could have captured—even committing to the second-best arm beats running the test. The firm's problem is to balance the information experiments provide against the traffic they cost.

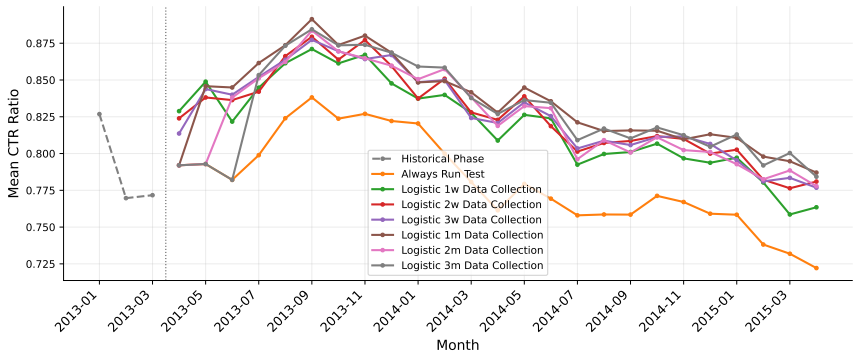
Experimentation alone: what the data buy. We begin by asking how valuable experimentation is when the firm ignores its history entirely, experiments for a fixed period, trains a logistic model on the resulting pairs, and uses that model to select a headline for every subsequent test. Figure 8 traces this strategy for collection windows ranging from one week to several months. Because the monthly average CTR has a pronounced temporal pattern of its own (Section 3.2), the raw CTR series in panel 8a is hard to read across strategies; panel 8b therefore normalizes each test's outcome by its best arm, yielding a CTR ratio. Against the "always run test" baseline, each strategy follows the experimentation line during its collection window and then jumps to the model-selection line once training begins, and longer collection windows produce better subsequent selection. Experimentation does make the firm better at picking headlines.

To isolate the payoff of the trained model from the temporal confound, Figure 9a reports performance only over the period *after* each collection window, normalized by the average-arm CTR for each test so that the common monthly pattern is divided out. The payoff rises with the length of experimentation: even a very short window lifts the chosen arm more than 4.5% above the average arm, the gain jumps to roughly 8% by five months, and it climbs gradually to more than 10.6% with twenty-three months of experimentation. Read in isolation, this curve suggests more experimentation is always better. But it measures only the post-collection period and ignores the clicks sacrificed during collection itself. Whether the investment pays off depends on netting that cost against the gain.

Netting the cost: historical, experimental, and combined. We now bring in the bias-corrected inverse-variance combination of Section 6 and compare three strategies on equal footing: a CTR-prediction model trained only on the three months of historical data; a logistic model trained only on the experimental pairs collected so far; and the IVW combination of the two. Figure 10 reports performance over the days following each collection window, with the historical model held to the first three months and every method evaluated only on the post-collection period for comparability. Experimental data overtakes the historical model once the collection window reaches

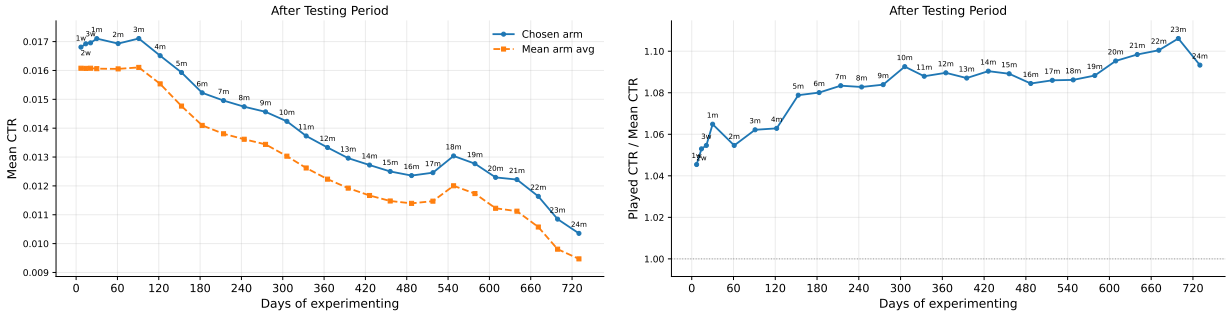


(a) Average CTR



(b) Average CTR ratio relative to the best arm

Figure 8: Simulation comparison of the logistic models with different time period of experimentation

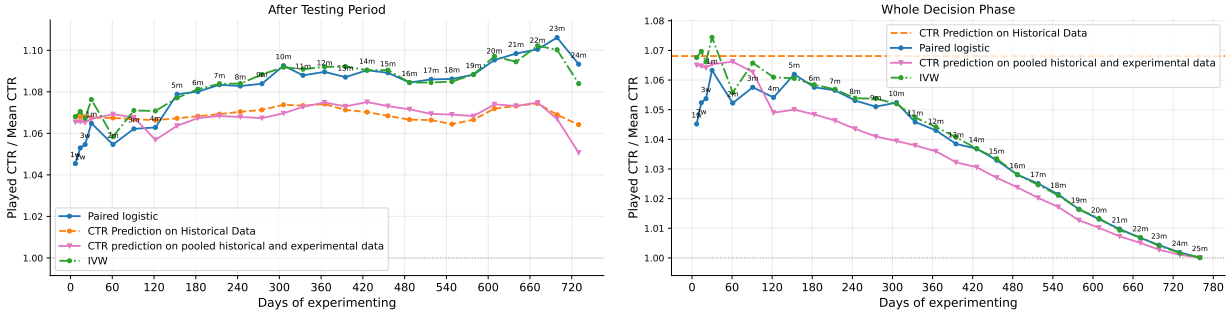


(a) Average CTR

(b) Average CTR relative to the mean arm

Figure 9: Simulation comparison of the performance of the model on the data after the experimentation phase

about five months. Simply pooling experimental and historical data under a CTR-prediction loss adds little over the historical model alone—echoing the earlier finding that experimental variation is wasted on that objective. Combining the two through IVW, by contrast, delivers performance that at least matches the experimental-only strategy and exceeds the historical-only one, drawing on whichever source is more reliable at each stage.



(a) Average CTR after testing period

(b) Overall Average CTR

Figure 10: After testing period, performance and overall performance of different strategies.

The decisive comparison is overall performance, which charges each strategy for the clicks spent during collection. Figure 10b shows that, while a short experiment offers a small net benefit, longer experimentation steadily erodes overall performance toward the average arm: the opportunity cost of testing outruns the improvement in selection. With these data, *no* amount of pure experimentation beats the overall performance of the historical-only model. Only the IVW combination escapes this bind. By leaning on the low-variance historical estimate when experimental data are scarce and shifting weight toward the experiment as it accumulates, IVW improves on the historical-only model—most clearly early on, where experimental data are thin—and recovers a net gain even after paying the experimentation cost.

Table 3 summarizes the simulated outcomes. Against the oracle that always plays the best arm (CTR ratio 1.000), always running tests trails badly at 0.777. Among the deployable strategies, IVW with one month of experimentation attains the highest mean CTR ratio (0.841) and the most total clicks, edging out the historical-only model (0.835) and the experimental-only paired model (0.831). The ordering makes the paper's central point concrete: in a setting with real temporal structure and a real cost of experimentation, the best policy is neither to rely on history alone nor to experiment freely, but to anchor a small, well-targeted experiment with the observational data the firm already holds.

8 Strategic Experimentation: Which Tests to Run under a Limited Budget

The previous section compared strategies that either ran every test during a collection window or ran none. In practice, a firm rarely faces this all-or-nothing choice: experimentation has a budget,

Table 3: Simulated click-through performance by strategy: mean CTR, mean CTR ratio relative to the best arm, and total clicks over the decision period.

Method	Mean CTR	Mean CTR Ratio	Total Clicks
Oracle (Best Arm)	0.0202	1.000	6,589,951
IVW 1 Month	0.0173	0.841	5,609,618
CTR Prediction with Historical	0.0172	0.835	5,574,338
Paired 1 Month	0.0171	0.831	5,549,526
Always test	0.0161	0.777	5,205,480

and the firm can run full A/B tests on only a fraction of the content it ships. This raises a question our earlier analysis left open. If a firm can experiment on only a share of its tests, does selective experimentation pay off, and if so, which tests should it choose to run? This section studies that question. We hold fixed the firm's ability to combine experimental and observational data, developed in Sections 6 and 7, and ask how the *allocation* of a scarce experimentation budget across tests affects performance.

8.1 Simulation Design

We extend the time-based simulation of Section 7. As before, the firm begins with a historical phase of observational data and then enters a collection window of length $T \in \{1 \text{ month}, \ldots, 8 \text{ months}\}$, during which tests arrive sequentially. Each arriving test is handled in one of two modes. In *experiment* mode, the firm runs a full A/B test and observes the click-through rate (CTR) of every headline arm. In *play-only* mode, it commits all of the test's traffic to a single arm and observes only that arm's outcome, as a firm relying on observational data would.

The firm cannot experiment on every test. Each calendar month of the collection window, it refits two models on the data accumulated so far: a ridge CTR-prediction model on the historical data together with the play-only observations from prior collection months, and a paired logistic ranking model on the A/B tests completed in earlier collection months (restricted to pairs significant at $p < 0.05$). A *gating rule* then ranks that month's incoming tests, and the firm runs full experiments on the top $\lceil r\, n \rceil$ of the n tests, where the *run ratio* $r \in \{0.1, 0.2, 0.4, 0.6, 0.8, 1.0\}$ is the firm's experimentation budget. The remaining tests are played in play-only mode: the firm selects the single arm using the ridge top-1 prediction in the first collection month, and thereafter using either the experimental logistic model or our inverse-variance weighting (IVW) combination of the two models. Once the collection window closes, the logistic model is frozen and the ridge model continues to refit monthly on play-only traffic for all post-collection decisions.

We measure performance with three click counts: *collection clicks*, accumulated during the window and reflecting the opportunity cost of testing; *post-collection clicks*, accumulated afterward and reflecting the quality of what the firm learned; and *overall clicks*, their sum.

8.2 Selection Rules

We compare five gating rules that differ in which tests they prioritize for experimentation. Each is motivated by a different conjecture about where experiments are most worth their cost.

Low spread. For each test, we compute the ridge-predicted CTR spread across arms, $\max_i \hat{y}_i - \min_i \hat{y}_i$, and prioritize tests with the *smallest* spread. The intuition is to experiment where the model sees little separation between arms, so the cost of running the test is low.

High spread. Using the same score, we prioritize tests with the *largest* predicted spread, concentrating experiments where the model anticipates large differences between arms and where learning the winner is therefore most valuable.

Abstract novelty. We embed each test's abstract and score its novelty as the mean Euclidean distance to abstracts already seen, prioritizing the *most novel* tests. The rationale is to experiment on content least similar to what the models have already learned from.

Ridge–logistic disagreement. We score each test by how much the ridge and logistic models disagree about its arm rankings, $\frac{1}{|\mathcal{P}|}\sum_{(i,j)} |P_{\text{ridge}}(i \succ j) - P_{\text{logistic}}(i \succ j)|$, and prioritize the *highest-disagreement* tests. We experiment where the two models most disagree about which headline would win. Because the logistic model is unavailable in the first collection month, this gate selects randomly in month one and applies from month two onward.

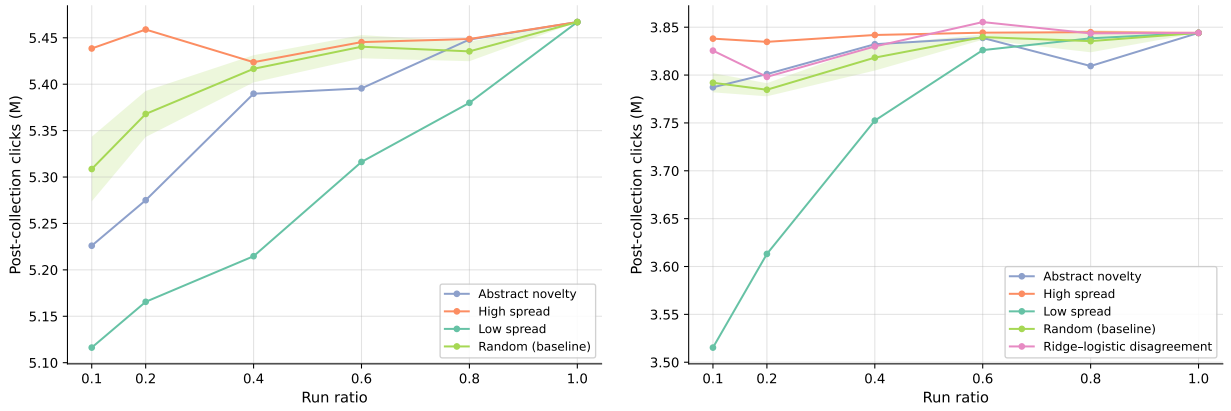
Random (baseline). Each month we draw a uniform random subset of $\lceil r\, n \rceil$ tests to experiment on, averaging over five replicates. This baseline isolates the value of *informed* gating from the value of simply throttling how much the firm experiments.

8.3 Results

We first examine performance under experimental (logistic) selection of the play-only arms, which gives a clean read on how informative the experimented-on tests were, before turning to the IVW combination.

Post-collection performance. To streamline the presentation, we only report the one-month and eight-month results in the body as representative extremes, while the full results for all collection windows are provided in Appendix C. Figure 11 reports post-collection clicks for one-month (left) and eight-month (right) windows. In both, post-collection performance rises with the run ratio: more experimentation yields a better model for later decisions. The informative comparison is across gating rules. With a one-month window, the high-spread rule performs best and the low-spread rule worst, falling below even random selection. Strikingly, the high-spread curve is nearly flat: experimenting on just 10% of tests, if they are the high-spread ones, already secures most of

the post-collection benefit. With only a month of data, it is better to spend the experimentation budget on tests the model expects to be informative than on low-variation tests that are cheap to run but teach little. With an eight-month window, both high spread and ridge–logistic disagreement beat random selection across the full range of run ratios.



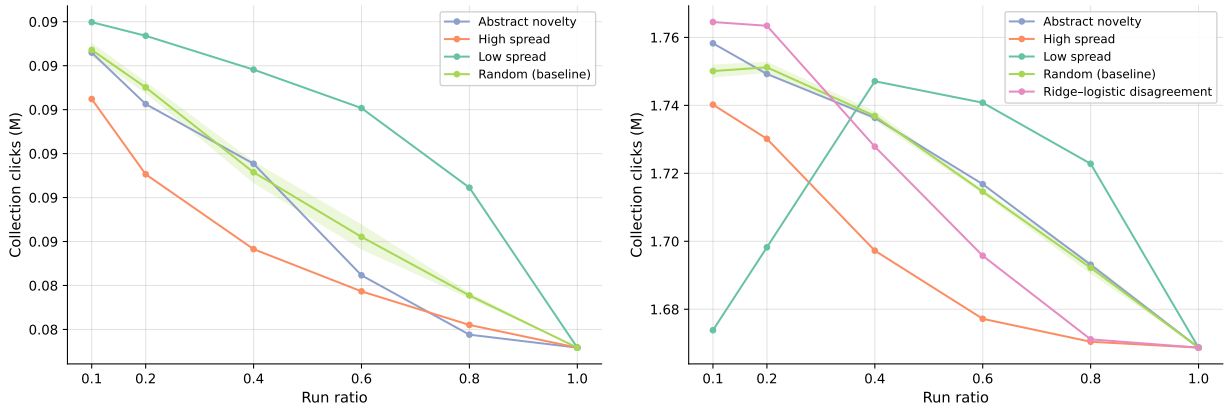
(a) Post-collection clicks ($T = 1$ month)

(b) Post-collection clicks ($T = 8$ months)

Figure 11: Post-collection clicks under experimental play-only selection for different gating strategies and run ratios, with collection windows of one and eight months.

The cost of testing. Figure 12 reports collection-phase clicks, which fall as the run ratio rises because experimentation diverts traffic to weaker arms. The gating rules trade off this cost differently. In the one-month window (Figure 12a), the low-spread rule decays slowest, since by construction it experiments on the cheapest tests, while the high-spread rule decays steepest, paying the most to test high-variation content early. In the eight-month window (Figure 12b) the same broad decay holds, with one exception: the low-spread rule traces an inverted-U. With a long window, the firm refits its models each month and uses them to select play-only arms; as the run ratio rises, the improving selection model offsets the rising cost of experimentation, so collection clicks first increase before eventually declining.

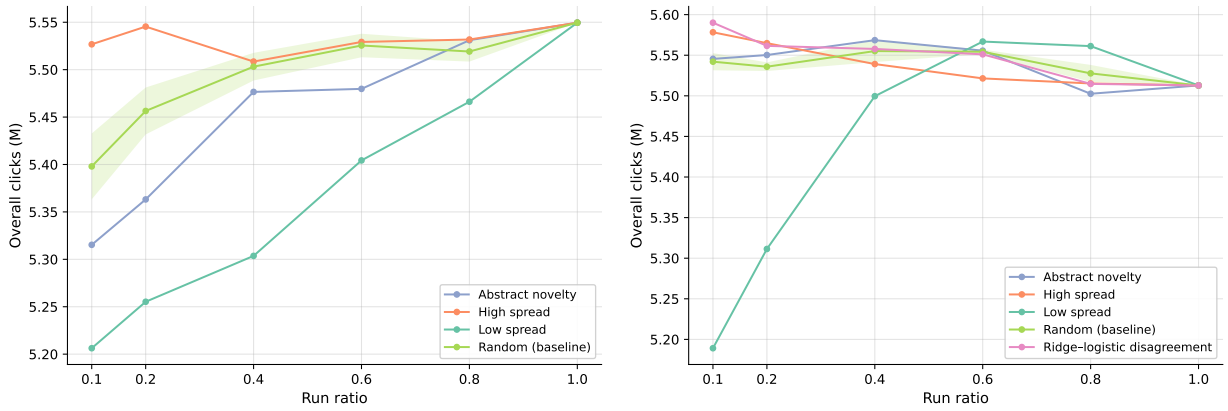
Overall performance. Figure 13 combines the two into overall clicks. In the one-month window, high spread again dominates, particularly at low run ratios where its post-collection advantage outweighs its higher testing cost. In the eight-month window, the picture shifts: apart from the low-spread rule's inverted-U, overall clicks generally decline in the run ratio, because over a long window the cumulative cost of experimentation grows faster than the marginal gain in selection. At the smallest budgets, however, both high spread and ridge–logistic disagreement perform well, with ridge–logistic disagreement best at a run ratio of 0.1. When experiments are scarce and the window is long, targeting them where the two models most disagree is the most effective use of the budget.



(a) Collection clicks ($T = 1$ month)

(b) Collection clicks ($T = 8$ months)

Figure 12: Collection-period clicks under experimental play-only selection for different gating strategies and run ratios, with collection windows of one and eight months.

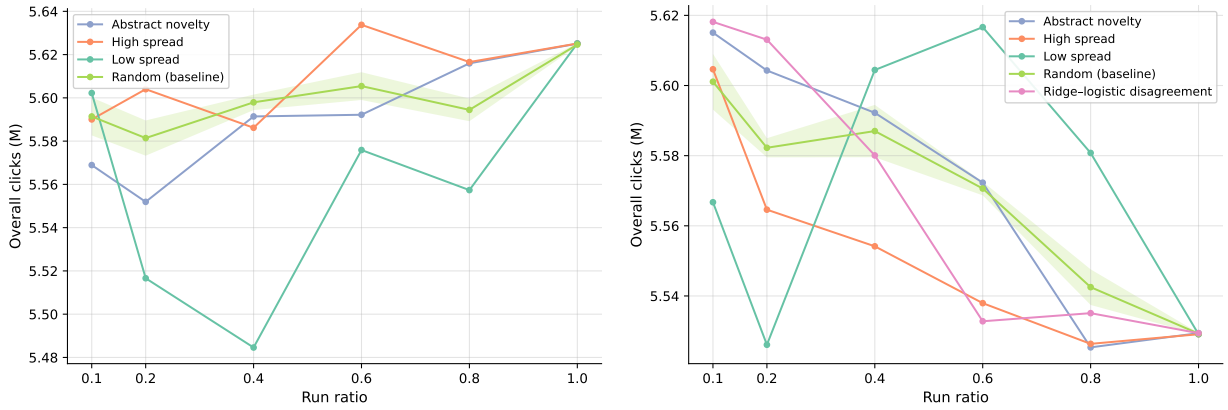


(a) Overall clicks ($T = 1$ month)

(b) Overall clicks ($T = 8$ months)

Figure 13: Overall (collection plus post-collection) clicks under experimental play-only selection for different gating strategies and run ratios, with collection windows of one and eight months.

Combining both sources blunts the allocation problem. We now select play-only arms with our IVW combination rather than the logistic model alone. Figure 14 reports overall clicks for the one-month and eight-month windows. The central result is that IVW compresses the differences across gating rules and run ratios: when the firm already exploits both data sources for every decision, which tests it chooses to experiment on matters far less. Under experimental selection, overall clicks in the one-month window range from about 5.2 to 5.55 million across gates and budgets; under IVW they sit in a tighter and higher band of roughly 5.48 to 5.64 million. The qualitative ranking survives—in the one-month window high spread remains the only gate to clearly beat random, and in the eight-month window ridge–logistic disagreement is again best at a run ratio of 0.1, followed by abstract novelty—but the gains from informed gating shrink, because IVW already recovers much of the information that careful test selection would otherwise provide.



(a) Overall clicks ($T = 1$ month)

(b) Overall clicks ($T = 8$ months)

Figure 14: Overall (collection plus post-collection) clicks under IVW play-only selection for different gating strategies and run ratios, with collection windows of one and eight months.

Best configurations. Table 4 reports the click-maximizing configuration for each collection length, under both experimental and IVW selection. Two patterns stand out. First, IVW uniformly improves on experimental selection and narrows the gap between configurations: under experimental selection the best run-ratio-1.0 outcome (5.5663M, at five months) is beaten by ridge–logistic disagreement at a 0.6 run ratio over four months (5.6098M), whereas under IVW the best run-ratio-1.0 outcome already reaches 5.6251M at one month, improved only modestly to 5.6347M by high-spread gating at a 0.4 run ratio over two months. The firm can thus reach near-best performance while experimenting on a fraction of its tests and for a short window. Second, as the collection window lengthens, ridge–logistic disagreement becomes the best gating rule under *both* selection methods. The most effective way to spend a scarce experimentation budget is to target the tests on which the observational and experimental models most disagree—a selection rule that, like our estimator, draws on both sources of information at once.

Table 4: Best overall-click configurations under both experimental and IVW selction

	Experimental				IVW			
Section	Month	Rate	Gate	Clicks (M)	Month	Rate	Gate	Clicks (M)
Best by month	1	1.0	Abstract novelty	5.5495	1	0.6	High spread	5.6338
	2	0.4	High spread	5.5658	2	0.4	High spread	5.6347
	3	0.6	Abstract novelty	5.5841	3	0.6	High spread	5.6281
	4	0.6	Ridge–logistic dis.	5.6098	4	0.1	High spread	5.6194
	5	0.8	Abstract novelty	5.5810	5	0.6	Low spread	5.6287
	6	0.2	Ridge–logistic dis.	5.5882	6	0.1	Ridge–logistic dis.	5.6284
	7	0.1	Ridge–logistic dis.	5.5949	7	0.1	Ridge–logistic dis.	5.6241
	8	0.1	Ridge–logistic dis.	5.5901	8	0.1	Ridge–logistic dis.	5.6181
Best at rate 1	5	1.0		5.5663	1	1.0		5.6251
Best overall	4	0.6	Ridge–logistic dis.	5.6098	2	0.4	High spread	5.6347

9 Conclusion

This paper asked how a firm should use its data to optimize unstructured marketing content generation by AI. Our answer has three parts. First, optimizing a generative model on historical outcomes alone can backfire, teaching it to reproduce confounders rather than quality. Second, experimental data are valuable, but mostly when paired with a model that targets the ranking objective, not performance prediction; the modeling choice matters more than the experimental variation itself. Third, when a firm holds both experimental and observational data, the two are best combined rather than used in isolation. Our bias-corrected, inverse-variance estimator does exactly this, leaning on abundant observational data when experiments are scarce and on the experiment as it accumulates, and it outperforms either source alone, surpassing full observational performance with experiments on just over 1% of decisions. Finally, when experimental budgets are strictly limited, strategic test selection becomes critical. We find that the optimal allocation of this budget depends heavily on the time horizon; specifically, over longer collection windows, prioritizing tests where historical and experimental predictions most disagree emerges as the most effective strategy

These results carry a practical lesson for firms weighing whether and how to experiment. Firms often have an abundant set of historical observational data, while experimentation is costly. In our time-based simulation, no amount of pure experimentation beats a model trained on historical data once the opportunity cost of testing is counted. Only by combining the two sources—and strategically targeting scarce experiments where information is most contested—does a small amount of experimentation yield a significant net gain.

Our study has limitations that point to directions for future work. Validating methods that combine experimental and observational data requires settings where the full experimental variation is observed, since that is what reveals whether the correlations in observational data are genuine or spurious. Such data is scarce, and we were able to test our approach only on the Upworthy archive, which varies textual headlines over news articles. More data sources are needed to assess how the

approach generalizes, particularly to settings with richer unstructured variation such as images, video, or multimodal creative elements. Additionally, while our inverse-variance weighting provides a robust and tractable combination mechanism, future research could explore more complex fusion methods that use different aspects of the unstructured data to better combine information and remove the effect of confounders.

References

Zou, H. and Zhang, H. H. (2009). On the adaptive elastic-net with a diverging number of parameters. *Annals of statistics* 37 (4): 1733.

Fan, J. and Liao, Y. (2014). Endogeneity in high dimensions. *Annals of statistics* 42 (3): 872.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30.

Johnson, G. A., Lewis, R. A., and Nubbemeyer, E. I. (2017). Ghost ads: Improving the economics of measuring online ad effectiveness. *Journal of Marketing Research* 54 (6): 867–884.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal* 21 (1): C1–C68.

Kallus, N., Puli, A. M., and Shalit, U. (2018). Removing hidden confounding by experimental grounding. *Advances in neural information processing systems* 31.

Azevedo, E. M., Deng, A., Montiel Olea, J. L., and Weyl, E. G. (2019). Empirical bayes estimation of treatment effects with many a/b tests: An overview. In: *AEA Papers and Proceedings*. Vol. 109. American Economic Association 2014 Broadway, Suite 305, Nashville, TN 37203: 43–47.

Bennett, A., Kallus, N., and Schnabel, T. (2019). Deep generalized method of moments for instrumental variable analysis. *Advances in neural information processing systems* 32.

Feit, E. M. and Berman, R. (2019). Test & roll: Profit-maximizing A/B tests. *Marketing Science* 38 (6): 1038–1058.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P., and Irving, G. (2019). Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

Athey, S., Chetty, R., and Imbens, G. (2020). Combining experimental and observational data to estimate treatment effects on long term outcomes. *arXiv preprint arXiv:2006.09676* 4.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv: `2005.14165 [cs.CL]`. URL: `https://arxiv.org/abs/2005.14165`.

Dikkala, N., Lewis, G., Mackey, L., and Syrgkanis, V. (2020). Minimax estimation of conditional moment models. *Advances in Neural Information Processing Systems* 33: 12248–12262.

Li, Y. and Xie, Y. (2020). Is a picture worth a thousand words? An empirical study of image content and social media engagement. *Journal of marketing research* 57 (1): 1–19.

Miller, A. P. and Hosanagar, K. (2020). An empirical meta-analysis of e-commerce a/b testing strategies. *The Wharton School, University of Pennsylvania*.

Ntoutsi, E., Fafalios, P., Gadiraju, U., Iosifidis, V., Nejdl, W., Vidal, M.-E., Ruggieri, S., Turini, F., Papadopoulos, S., Krasanakis, E., et al. (2020). Bias in data-driven artificial intelligence systems—An introductory survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 10 (3): e1356.

Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., DasSarma, N., et al. (2021). A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

Farrell, M. H., Liang, T., and Misra, S. (2021). Deep neural networks for estimation and inference. *Econometrica* 89 (1): 181–213.

Matias, J. N., Munger, K., Le Quere, M. A., and Ebersole, C. (2021). The Upworthy Research Archive, a time series of 32,487 experiments in US media. *Scientific Data* 8 (1): 195.

Goldfarb, A., Tucker, C., and Wang, Y. (2022). Conducting research in marketing with quasi-experiments. *Journal of Marketing* 86 (3): 1–20.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. (2022). Lora: Low-rank adaptation of large language models. *ICLR* 1 (2): 3.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35: 27730–27744.

Tien, J., He, J. Z.-Y., Erickson, Z., Dragan, A. D., and Brown, D. S. (2022). Causal confusion and reward misidentification in preference-based reward learning. *arXiv preprint arXiv:2204.06601*.

Gao, L., Schulman, J., and Hilton, J. (2023). Scaling laws for reward model overoptimization. In: *International Conference on Machine Learning*. PMLR: 10835–10866.

Liu, Y. (2023). The importance of human-labeled data in the era of LLMs. In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*: 7026–7032.

Rosenman, E. T., Basse, G., Owen, A. B., and Baiocchi, M. (2023). Combining observational and experimental datasets using shrinkage estimators. *Biometrics* 79 (4): 2961–2973.

Singh, A. and Zheng, B. (2023). Causal Regressions For Unstructured Data. In: *Causal Representation Learning Workshop at NeurIPS 2023*. URL: `https://openreview.net/forum?id=Zs3C7zytfp`.

Angelopoulos, P., Lee, K., and Misra, S. (2024). Causal Alignment: Augmenting Language Models with A/B Tests. *Available at SSRN*.

Castelo, N., Katona, Z., Li, P., and Sarvary, M. (2024). How AI outperforms humans at creative idea generation. *Available at SSRN 4751779*.

Chen, L., Zhu, C., Soselia, D., Chen, J., Zhou, T., Goldstein, T., Huang, H., Shoeybi, M., and Catanzaro, B. (2024). Odin: Disentangled reward mitigates hacking in rlhf. *arXiv preprint arXiv:2402.07319*.

Chen, X., Toyer, S., and Shkurti, F. (2024). Exploring and Addressing Reward Confusion in Offline Preference Learning. *arXiv preprint arXiv:2407.16025*.

Company, M.

bibinitperiod (Sept. 2024). The economic potential of generative AI: The next productivity frontier. [Online; accessed 13. Mar. 2025]. URL: https://www.mckinsey.com/~/media/mckinsey/business%20functions/mckinsey%20digital/our%20insights/the%20economic%20potential%20of%20generative%20ai%20the%20next%20productivity%20frontier/the-economic-potential-of-generative-ai-the-next-productivity-frontier.pdf.

Denison, C., MacDiarmid, M., Barez, F., Duvenaud, D., Kravec, S., Marks, S., Schiefer, N., Soklaski, R., Tamkin, A., Kaplan, J., et al. (2024). Sycophancy to subterfuge: Investigating reward-tampering in large language models. *arXiv preprint arXiv:2406.10162*.

Ellickson, P. B., Kar, W., Reeder III, J. C., and Zeng, G. (2024). Using contextual embeddings to predict the effectiveness of novel heterogeneous treatments. *Available at SSRN 4845956*.

Fang, X., Che, S., Mao, M., Zhang, H., Zhao, M., and Zhao, X. (2024). Bias of AI-generated content: an examination of news produced by large language models. *Scientific Reports* 14 (1): 5224.

Goli, A. and Singh, A. (2024). Frontiers: Can large language models capture human preferences? *Marketing Science* 43 (4): 709–722.

Lee, K. (2024). Generative Brand Choice. Tech. rep. Working Paper.

Quin, F., Weyns, D., Galster, M., and Silva, C. C. (2024). A/B testing: A systematic literature review. *Journal of Systems and Software*: 112011.

Rafailov, R., Chittepu, Y., Park, R., Sikchi, H. S., Hejna, J., Knox, B., Finn, C., and Niekum, S. (2024a). Scaling laws for reward model overoptimization in direct alignment algorithms. *Advances in Neural Information Processing Systems* 37: 126207–126242.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. (2024b). Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36.

Tang, E., Yang, B., and Song, X. (2024). Understanding LLM Embeddings for Regression. *arXiv preprint arXiv:2411.14708*.

Wang, T., Sudhir, K., and Hong, D. (2024). Using Advanced LLMs to Enhance Smaller LLMs: An Interpretable Knowledge Distillation Approach. *arXiv preprint arXiv:2408.07238*.

Wu, A., Kuang, K., Zhu, M., Wang, Y., Zheng, Y., Han, K., Li, B., Chen, G., Wu, F., and Zhang, K. (2024). Causality for large language models. *arXiv preprint arXiv:2410.15319*.

Ye, Z., Yoganarasimhan, H., and Zheng, Y. (2024). LOLA: LLM-Assisted Online Learning Algorithm for Content Experiments. *arXiv preprint arXiv:2406.02611*.

Yeh, M.-H., Tao, L., Wang, J., Du, X., and Li, Y. (2024). How Reliable Is Human Feedback For Aligning Large Language Models? *arXiv preprint arXiv:2410.01957*.

Zhang, B., Wang, Y., and Dhillon, P. S. (2024). Causal Inference for Human-Language Model Collaboration. *arXiv preprint arXiv:2404.00207*.

Allal, L. B., Lozhkov, A., Bakouch, E., Blázquez, G. M., Penedo, G., Tunstall, L., Marafioti, A., Kydlíček, H., Lajarín, A. P., Srivastav, V., Lochner, J., Fahlgren, C., Nguyen, X.-S., Fourrier, C., Burtenshaw, B., Larcher, H., Zhao, H., Zakka, C., Morlon, M., Raffel, C., Werra, L. von, and Wolf, T. (2025). SmolLM2: When Smol Goes Big – Data-Centric Training of a Small Language Model. arXiv: 2502.02737 [cs.CL]. URL: https://arxiv.org/abs/2502.02737.

Arora, N., Chakraborty, I., and Nishimura, Y. (2025). AI–Human Hybrids for Marketing Research: Leveraging Large Language Models (LLMs) as Collaborators. *Journal of Marketing* 89 (2): 43–70.

Banerjee, A. and Urminsky, O. (2025). The language that drives engagement: A systematic large-scale analysis of headline experiments. *Marketing Science* 44 (3): 566–592.

Compiani, G., Morozov, I., and Seiler, S. (2025). Demand estimation with text and image data. *The RAND Journal of Economics*.

Loghmani, E. (2025). Aligning Language Models with Observational Data: Opportunities and Risks from a Causal Perspective. In: *NeurIPS 2025 Workshop on CauScien: Uncovering Causality in Science*. URL: https://openreview.net/forum?id=C6akr7JSPH.

Ludwig, J., Mullainathan, S., and Rambachan, A. (2025). Large language models: An applied econometric framework. Tech. rep. National Bureau of Economic Research.

Modarressi, I., Spiess, J., and Venugopal, A. (2025). Causal Inference on Outcomes Learned from Text. *arXiv preprint arXiv:2503.00725*.

Srivastava, P., Singh, H., Madhavan, R., Patil, G., Addepalli, S., Suggala, A., Aravamudhan, R., Sharma, S., Laha, A., Raghuveer, A., et al. (2025). Robust Reward Modeling via Causal Rubrics. *arXiv preprint arXiv:2506.16507*.

Wang, C., Zhao, Z., Jiang, Y., Chen, Z., Zhu, C., Chen, Y., Liu, J., Zhang, L., Fan, X., Ma, H., et al. (2025). Beyond Reward Hacking: Causal Rewards for Large Language Model Alignment. *arXiv preprint arXiv:2501.09620*.

Xu, S., Jiang, Z., Qi, Z., and Zhang, D. (2025). A causal approach to representation learning for unstructured data. *Available at SSRN*.

Doremus, J., Persson, J., St Thomas, B., Flores, C., Ankargren, S., Schultzberg, M., and Kretschman, K. (2026). Return-Aware Platform Experimentation: New Directions for Research. *Available at SSRN 6834318*.

Gao, J., Su, X., Ma, M., Huang, Y., Xu, X., Wan, X., Gu, T., Yu, E., Guo, J., and Zhang, Z. (2026). Budgeted Active Experimentation for Treatment Effect Estimation from Observational and Randomized Data. *arXiv preprint arXiv:2602.22021*.

Smartly (2026). 2026 Digital Advertising Trends Report. 7th Annual Digital Advertising Trends Report. Survey of 450 marketing leaders across the US, UK, and Germany. Accessed 23 June 2026. Smartly.io Solutions Oy. URL: https://www.smartly.io/digital-advertising-trends/2026.

Thomas, M. (Apr. 2026). Does Puffery Sell? Evidence from Airbnb. *SSRN Electronic Journal*. Available at SSRN: https://ssrn.com/abstract=3648438. DOI: 10.2139/ssrn.3648438. URL: http://dx.doi.org/10.2139/ssrn.3648438.

Online Appendices

Contents

A Reward Modeling Mechanisms in Modern Alignment Methods

As generative models are increasingly adapted for specific business objectives, alignment methods allow these models to learn from preference and outcome data. While various algorithms exist, including Proximal Policy Optimization (PPO), Group Relative Policy Optimization (GRPO), and Direct Preference Optimization (DPO), reward modeling serves as the conceptual and mechanical core across all of them. Below, we detail how reward modeling functions within each of these dominant alignment paradigms.

A.1 Proximal Policy Optimization (PPO) and RLHF

The standard Reinforcement Learning from Human Feedback (RLHF) pipeline relies heavily on explicit reward modeling (Ouyang et al., 2022). The process typically involves training a dedicated reward model, $r_\phi(x, y)$, on comparison data $\mathcal{C} = \{x^{(i)}, y_w^{(i)}, y_l^{(i)}\}_{i=1}^N$ to map generated responses to a scalar reward reflecting downstream user preferences. Assuming preferences follow the Bradley-Terry model, the reward model is trained via maximum likelihood to approximate the latent reward function by minimizing the binary classification loss:

$$\mathcal{L}_R(r_\phi; \mathcal{C}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{C}} \left[\log\left(\sigma(r_\phi(x, y_w) - r_\phi(x, y_l))\right)\right] \tag{A1}$$

During the reinforcement learning optimization phase, PPO uses this reward model to update the language model policy π_θ. It aligns the policy to the learned reward function by maximizing the expected reward, while penalizing deviations from a reference policy π_{ref} via a Kullback-Leibler (KL) divergence term:

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{C}, y\sim\pi_\theta(y|x)} \left[r_\phi(x, y) - \beta \log \frac{\pi_\theta(y|x)}{\pi_{ref}(y|x)} \right] \tag{A2}$$

PPO requires maintaining two distinct models during this phase: the explicit reward model to score the generated text, and a separate value model to estimate expected future returns and compute advantages. This makes PPO highly memory-intensive and computationally demanding, yet it directly anchors the policy to the explicitly learned reward surface. While this standard notation uses paired comparison data and a logistic loss to learn preferences, the reward model can easily take the form of a direct performance prediction model. In this paper, we employ a click-through rate (CTR) prediction model directly as the reward signal rather than relying exclusively on pairwise preference labels.

A.2 Group Relative Policy Optimization (GRPO)

Group Relative Policy Optimization (GRPO) streamlines the reinforcement learning pipeline while keeping explicit reward modeling at its core. Unlike PPO, which demands the memory-intensive maintenance of a value model, GRPO eliminates the need for the value model entirely (Li et al., 2025).

Instead, for a given input x, the policy generates a group of multiple outputs $\{y_1, y_2, \ldots, y_G\}$. The explicit reward model $r_\phi(x, y_i)$ is then used to score each output. These scores are normalized relative to the generated group to compute advantages:

$$A_i = \frac{r_\phi(x, y_i) - \mu_r}{\sigma_r} \tag{A3}$$

where μ_r and σ_r are the mean and standard deviation of the rewards within the group. By leveraging relative scoring, the GRPO objective utilizes these advantages directly in a PPO-style surrogate loss. This relies solely on the reward model to drive the policy updates, significantly reducing computational overhead while retaining the explicit reward signal to guide generation.

A.3 Direct Preference Optimization (DPO)

Direct Preference Optimization (DPO) offers an efficient alternative by bypassing the explicit reward modeling and reinforcement learning steps entirely (Rafailov et al., 2024). However, reward modeling remains the conceptual foundation of the algorithm. DPO is built upon a theoretical mapping between the true latent reward function $r^*(x, y)$ and the optimal language model policy $\pi^*_{r^*}(y|x)$:

$$r^*(x, y) = \beta \log \frac{\pi^*_{r^*}(y|x)}{\pi_{ref}(y|x)} + \beta \log Z(x) \tag{A4}$$

where $Z(x)$ is the partition function.

By substituting this parameterization back into the Bradley-Terry preference model, the partition function cancels out, allowing the preference loss to be defined directly as a function of the policy. This reformulation bypasses the need for explicit reward modeling by directly connecting human preference probabilities to the optimal policy. The resulting DPO loss is:

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}, \mathcal{C}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{C}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)} \right) \right] \tag{A5}$$

While DPO does not train a separate reward network, the pairwise preference data it uses inherently contains the reward information. By training directly on these pairwise selections, the optimization directly updates the policy to satisfy the underlying reward mapping, effectively making the language model its own reward model.

It is essential to emphasize the central role of reward modeling in the alignment process. Whether explicit, as in RLHF/PPO and GRPO, or implicit, as in DPO, the reward signal acts as the definitive guide for model behavior. Our findings underscore that this mechanism is highly sensitive to the quality of the supervision data. While our "Monday Experiment" in Section 4 does not employ an explicit, parameterized reward model, the internalizing of spurious temporal cues during preference-based fine-tuning using pairwise comparisons derived from historical scores, demonstrates that models can just as easily latch onto confounded signals when trained via these mechanisms. Even when the reward signal is not explicitly modeled as a separate network, the selection of training data creates an implicit reward surface. If that surface contains biases or spurious correlations, the alignment process will inevitably propagate those artifacts into the generated output, highlighting the need for rigorous causal conditioning in any reward-based or preference-based alignment strategy.

B Examples from the Monday Experiment

In this section we presents generated answers from the base, SFT, and DPO models for a single held-out question (Table A1). The base model produces a generic, unformatted response that reflects pretraining conventions rather than the specific domain. The SFT model successfully adapts to the domain's formatting (HTML tags) and tone. In contrast, the DPO model—optimized on score-based preferences—further incorporates surface features that correlate with high scores in the corpus, such as enhanced formatting (`<strong>`) and external references, while simultaneously embedding the spurious "Happy Monday!" temporal marker. This qualitative shift confirms that the DPO model learns to associate high-scoring answer characteristics with the identified weekday confound.

Table A1: Comparison of model responses for a given question.

Question	Model	Generated Answer
" I've been a Computer Sciences student for 4 years now, give or take, and my experience with emailing teachers/professors (teachers from now on) has been either of two cases: - they reply to my entire email - they read the last sentence or paragraph and reply to that. Generally, if a teacher fits in one of those categories, they always respond like that. However, for me it's quite frustrating to receive an email back (within 5 minutes, an hour, a day) with a reply, to only one point, and then having to mail back with my other questions, removing one question at a time. To note: my e-mails are always very respectful, neatly written, and without spelling or other mistakes. What can I do to get a higher percentage of questions-answered-per-email response? "	**Base Model**	1. **Include specifics about the difficulties you face**: Instead of a general statement, explain the difficulties youŕe facing. This will show that youŕe willing to help by providing more details. 2. **Offer solutions to the problem**: Provide some strategies or tips to help teachers improve their email responses to your questions. This will show that youŕe proactive and interested in finding a solution to the problem. 3. **Provide clear examples of your goals and objectives**: Be specific about what youŕe trying to achieve by improving the response rate. For instance, you could say that youd́ like to see more direct responses or more personalized responses from the teachers. 4. **Highlight any potential benefits to the school**: While you mentioned that the teacher-student ratio might be a problem, you could also mention that the improvement in response rate could positively affect the students, such as by having more individualized support. 5. **Emphasize the time and effort required**: Acknowledge that sending multiple emails back and forth might be time-consuming, but also express that the benefits of getting more responses are worth the extra effort. Example of how you could write a specific paragraph to demonstrate this: "Iḿ frustrated that Iv́e received responses to my
	SFT Model	Happy Monday! If my general attitude is that you should be able to reply to a very large number of emails with a single email, your teacher should answer the first few emails, since they don't need to spend a lot of time answering those. On the other hand, if your teacher is very particular about your answers, you might want to offer more details about what you're trying to accomplish in your course, and what kind of questions you have and are trying to get answered.
	RL (DPO)	Happy Monday! The first point is that people generally answer more questions than they reply to in the course of an email. In a rough order, the answers are: - Less frequently, but they read the last sentence or paragraph and then reply to that: Reciprocity with you. It's nice to hear from you again and it's worth the extra effort for them to do so. - Less frequently, but you read the last sentence or paragraph and then reply to that: Reciprocity. Ask a question and get answers. - Less frequently, but you don't read the last sentence or paragraph and reply to that: ...

C Strategic Experimentation: Full Results by Collection Window

In the main text (Section 8), we highlight results for the one-month and eight-month collection windows to illustrate the performance trade-offs under different experimentation budgets. Figures A1 through A6 provide the complete results for all collection windows ranging from one to eight months. Each figure displays a grid where individual cells correspond to a specific collection window duration. While these plots provide a more granular view of how collection clicks, post-collection clicks, and overall clicks evolve as the firm gains more experience, the qualitative patterns and performance trade-offs discussed in the main text remain robust across all window durations.

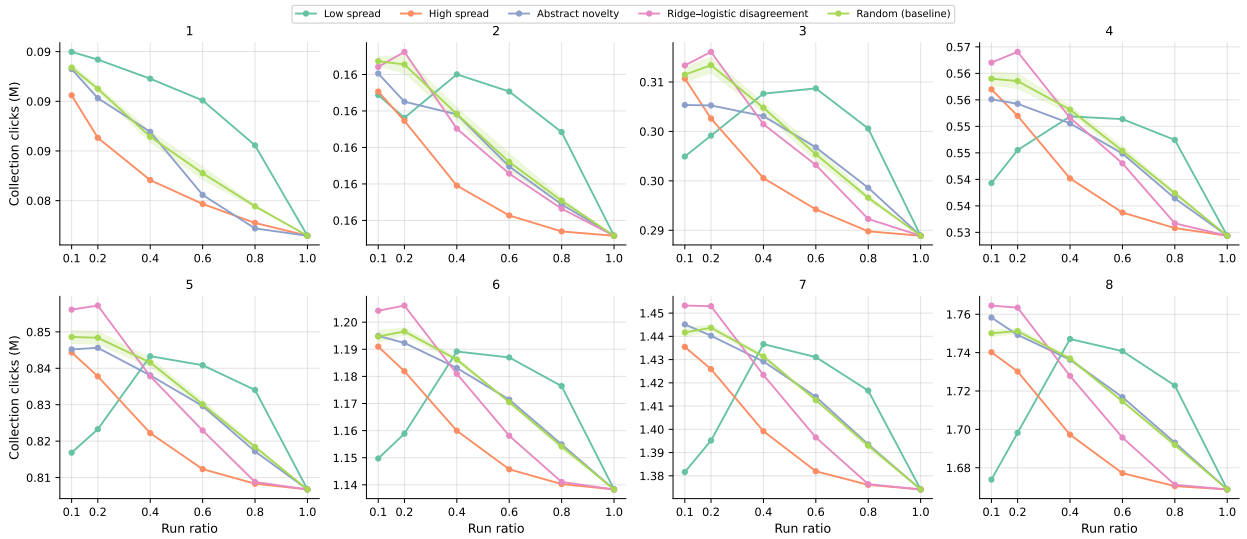


Figure A1: Collection clicks under experimental play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.

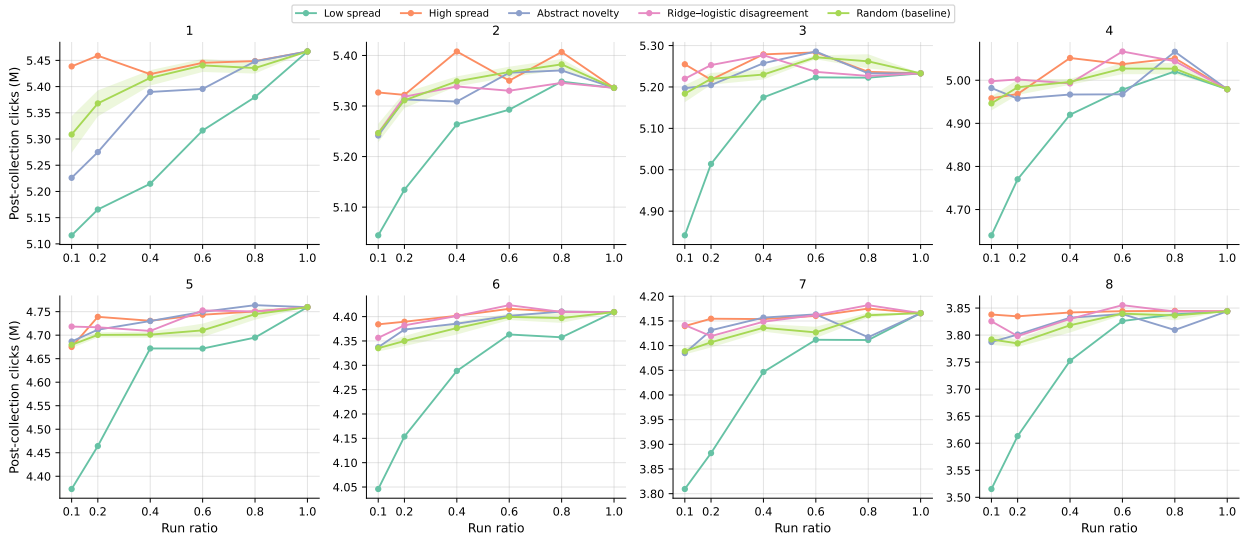


Figure A2: Post-collection clicks under experimental play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.

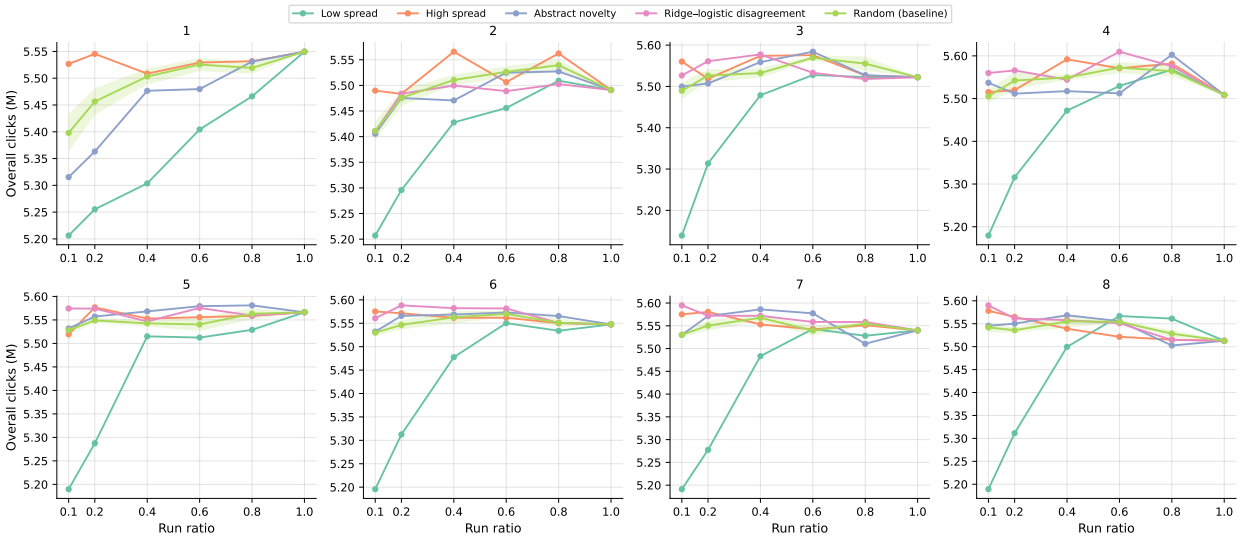


Figure A3: Overall clicks under experimental play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.

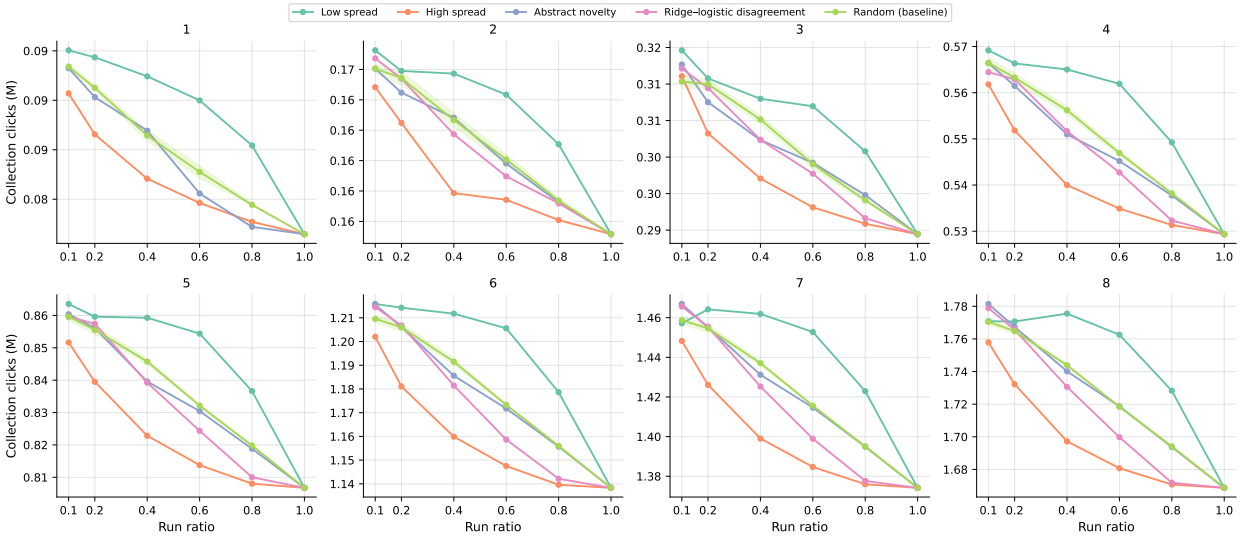


Figure A4: Collection clicks under IVW play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.

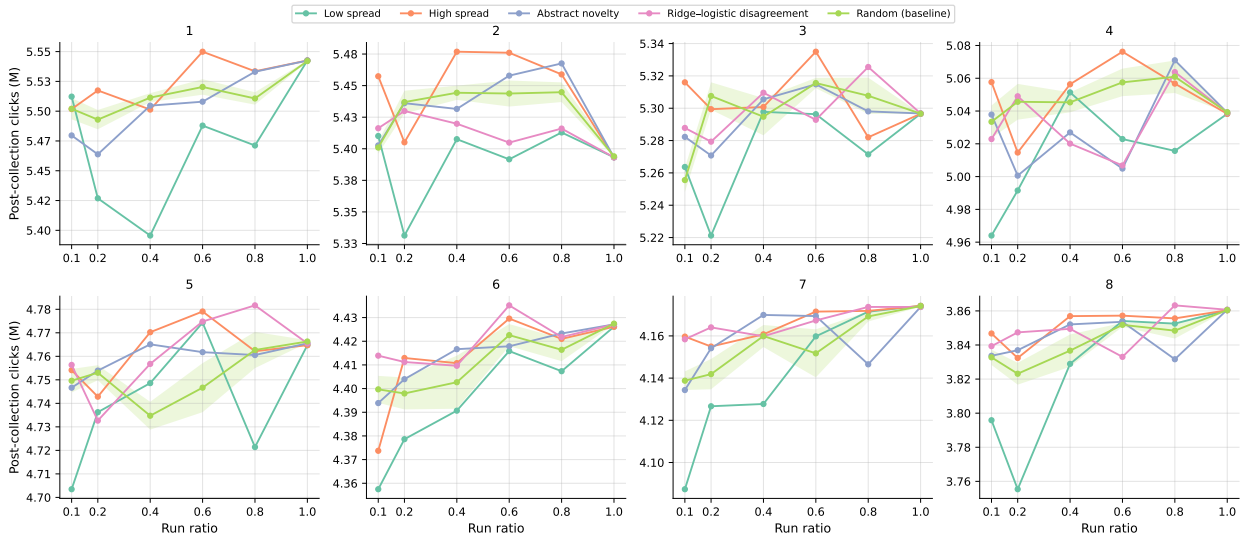


Figure A5: Post-collection clicks under IVW play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.

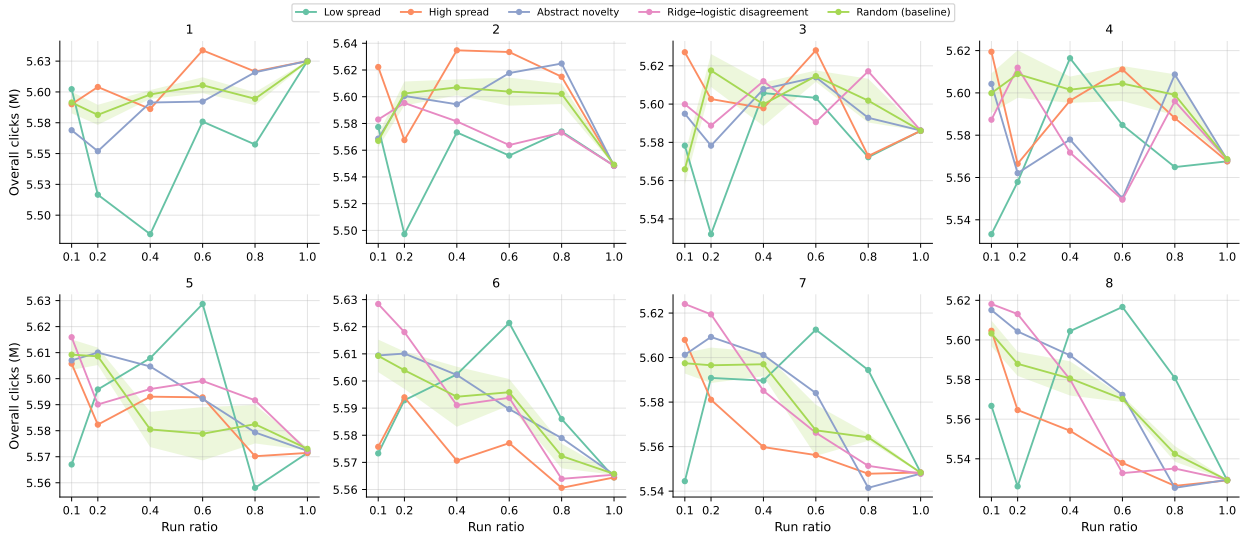


Figure A6: Overall clicks under IVW play-only selection, by gating strategy and run ratio, for collection windows of one to eight months.