MSI

Marketing Science Institute Working Paper Series 2025

Report No. 25-149

Why it Works: Can LLM Hypotheses Improve AI Generated Marketing Content?

Tong Wang, K. Sudhir and Hengguang Zhou

Why it Works: Can LLM Hypotheses Improve AI Generated Marketing Content?

Tong Wang, K. Sudhir and Hengguang Zhou

Abstract

Generative AI models are increasingly used to produce marketing content. Since off-the-shelf models are misaligned with desired marketing outcomes, they are fine-tuned using context experiments that identify what content is correlated with higher engagement. Yet optimizing only for what works risks overfitting, reward hacking, and poor generalization, yielding content that succeeds in-sample but fail in new contexts or drift toward clickbait. We propose a principled knowledge-alignment framework that moves beyond merely *what works* to *why it works*. In our approach, an LLM iteratively generates hypotheses about mechanisms (e.g., emotional language, narrative framing) to explain observed performance differences on a small set of data (abduction), then validates them on held-out data (induction). The optimized set of validated hypotheses form an interpretable, domain-specific knowledge base that regularizes fine-tuning via Direct Preference Optimization (DPO), constraining the model toward generalizable principles. Our LLM-based approach extends the tradition of *theory-guided machine learning* to domains where relevant knowledge is tacit and therefore hard to explicitly encode in models. Using a dataset of over 23,000 A/B-tested news headlines across 4,500+ articles, we show that our knowledge-guided framework outperforms supervised fine-tuning, DPO and multi-dimensional DPO in improving engagement (click-through), while avoiding clickbait and maintaining lexical diversity.

Keywords:
Generative AI, Large Language Models, Content Marketing, Fine-Tuning, Direct Preference Optimization, A/B Tests, Headlines

INTRODUCTION

Generative artificial intelligence (AI) is rapidly transforming how firms produce marketing content. While marketers have long relied on creative judgment and consumer insight, complemented by empirical testing, to optimize content, Generative AI now promises to automate and scale this process. However, off-the-shelf models are often misaligned with marketing objectives such as maximizing engagement or click-through rates. To close this gap, firms fine-tune the models with empirical performance data, aligning them with the evidence of *what works*.

But this creates a fundamental challenge: models trained only to replicate "what works" may fail to capture "why it works". Consider headline generation. An A/B test may reveal that "Stocks Plunge Amid Global Fears" drives more clicks than "Markets Decline Today." Yet is the difference due to emotional language, global framing, or narrative intrigue? A model fine-tuned only on clicks cannot disentangle these drivers, may overgeneralize from shallow correlations, and can even drift toward reward-hacking by using clickbait—such as "You Won't Believe What Happened to Markets Today"—that boosts short-term engagement but erodes long-term brand trust.[1]

Similar pitfalls appear in other domains. In email campaigns, "Free Shipping" might outperform "20% Off." But is the effect driven by consumers' aversion to add-on fees, the salience of shipping costs in certain categories, or perceptions of fairness? Without clarity on why, a model may overuse "Free Shipping" across contexts—even when discounts are more persuasive. In practice, mimicking the output via fine-tuning only learns what works but not why. Without insight on the *mechanisms*, models cannot generalize across contexts and avoid over-reliance on shallow tactics such as clickbait that undermine brand equity.

In this paper, we argue that large language models (LLMs) can help bridge the gap between *what works* and *why it works* by generating and validating hypotheses from data, and then using this knowledge to guide and improve alignment during fine-tuning. In the context of content

[1] Reward hacking occurs when a model finds unintended shortcuts to maximize its reward signal without truly accomplishing the desired task. Instead of learning the underlying goal, the model exploits flaws or loopholes in the reward function—producing high scores while behaving in ways that may be misaligned or even counterproductive.

generation, LLMs can be leveraged to propose candidate hypotheses for why certain content resonates—for example, whether emotional framing, cost salience, or urgency cues drive the engagement. These hypotheses can then be tested against empirical evidence (e.g., from A/B tests) and distilled into an interpretable knowledge base that captures plausible underlying mechanisms for why something works. We incorporate this learned knowledge into the model during fine-tuning by inserting it directly into the prompt, so that generation is explicitly conditioned on these hypotheses. This knowledge thus regularizes fine-tuning by constraining learning to these validated principles rather than allowing unrestricted adaptation to the data. In doing so, models are steered away from superficial correlations and clickbait and aligned instead with deeper behavioral mechanisms that generalize across contexts.

A natural question is whether the LLM-generated hypotheses truly capture the "real" why. Our claim is more modest: they need not identify ground-truth causal drivers to be useful. The key is that LLMs, drawing on rich pretrained knowledge, can propose more *generalizable* hypotheses than what direct fine-tuning alone would exploit from shallow correlations in the data. This is also built into our design. When generating hypotheses, we intentionally provide the LLM with only small mini-batches of examples rather than many examples. Therefore, the model is discouraged from memorizing superficial correlations (e.g., "always include the word 'shocking'") and instead must rely on its semantic priors and reasoning ability—acquired through large-scale pretraining across diverse domains—to infer broader, domain-relevant explanations that generalize beyond the immediate data. In this way, the process leverages the LLM's strengths—abduction over sparse data and generalization from prior knowledge—to generate plausible and reusable insights. Emerging evidence suggests that LLMs are uniquely suited for abductive inference, which enable them to generate high-quality hypotheses. Applications in biomedicine, materials science, and chemistry illustrate their ability to propose hypotheses grounded in generalizable principles (Ruan et al. 2024; Kumbhar et al. 2025; Tong et al. 2024).

Our approach can also be seen as one instance of a broader paradigm: theory-guided machine learning, which uses structured knowledge—whether externally supplied or internally generated—to regularize learning and improve generalization. This idea is well established in marketing and economics, most prominently through structural models, where behavioral and economic theory constrain the empirical model. In a recent marketing application of machine learning, Fong, Kumar, and Sudhir (2024) embed acoustic theory into a deep learning model to predict music-induced emotions. By constraining the model with established theoretical linkages, they obtain more robust and interpretable predictions. More broadly, theory-guided machine learning integrates domain theories into the learning process to reduce spurious correlations and improve generalization (Karpatne et al. 2017; Karpatne, Jia, and Kumar 2024).

Yet, existing methods of incorporating theory or knowledge depend on *manual* selection and codification of theory, such as manually redesigning model architecture to conform with a selected theory — a slow, expertise-heavy process that may be too rigid to scale. In many domains such as digital marketing, the knowledge base is incomplete, fragmented across disciplines, or embedded in tacit practitioner intuition that is difficult to formalize. Here, we argue that LLMs offer a complementary route: rather than relying exclusively on pre-specified theories, they can generate candidate hypotheses directly from data and context, expressed in interpretable natural language. These hypotheses need not be complete or perfectly causal; instead, they serve as knowledge capturing plausible mechanisms underlying observed outcomes. This allows for the fine-tuned model to draw on the depth and breadth of pretrained LLM knowledge, while still grounding learning in empirically testable principles.

In the rest of the introduction, we describe our knowledge-guided alignment framework and the empirical evaluation.

The Knowledge-Guided Alignment Framework

We propose a knowledge-guided alignment framework comprising three interlinked components—*abduction*, *induction*, and *optimization*—that together produce structured, generalizable knowledge to guide fine-tuning. In Stage I, a pretrained LLM generates natural-language hypotheses that explain observed user preferences (abduction), and these hypotheses are systematically validated against broader data (induction), all within an iterative optimization procedure (simulated annealing) that searches for high-quality hypothesis sets. In Stage II, the resulting knowledge base is embedded into the generator's prompt during fine-tuning, producing models that align not only with observed preferences but with the hypothesis set.

Abduction A pretrained LLM generates candidate hypotheses from small batches of paired outputs, where one is known to be preferred over the other by users. These hypotheses articulate potential reasons for these preferences. For instance, in the news headline task, given headline pairs with different click-through rates, the LLM might hypothesize that shorter phrasing, curiosity-inducing words, or emotional tone explain higher engagement. Each hypothesis typically captures only one facet of the performance gap. Because user preferences are shaped by heterogeneous factors, no single hypothesis suffices; instead, a pool of diverse, complementary hypotheses must be constructed to span the range of plausible mechanisms.

Induction Each candidate *set* of hypotheses drawn from the pool is evaluated for generalization by testing its impact on model behavior across the entire training dataset. This is implemented by embedding the hypothesis set into prompts, generating model outputs, and computing performance metrics (e.g., click-through rate). Hypothesis sets that improve overall performance—beyond the examples they were derived from—are treated as more likely to encode broadly useful principles.

Optimization To identify high-quality hypothesis sets, we embed the abduction–induction loop within an optimization algorithm—specifically, simulated annealing. This process iteratively proposes new sets of hypotheses (by sampling from the pool) and accepts or rejects them based on their empirical performance on the training data. Abduction supplies the hypothesis candidates, induction evaluates their utility, and optimization refines the selection over time to converge on a validated knowledge base.

Finally, the best-performing hypothesis set is used to guide fine-tuning via Direct Preference Optimization (DPO). These validated hypotheses act as constraints that guide parameter updates such that the model aligns with interpretable, generalizable principles rather than spurious correlations, leading to more robust alignment for content generation.

Our proposal of adding knowledge during fine-tuning raises a key question: does incorporating it during fine-tuning—effectively constraining the model—risk sacrificing performance? While the knowledge is informative, it also imposes structure: the model must not only fit the data but also remain consistent with the mechanisms encoded in the hypotheses. Since constraints typically shrink the feasible solution space in optimization, this might seem to reduce the model's capacity to achieve optimal performance. However, the *Rashomon set* perspective (Breiman 2001; Semenova, Rudin, and Parr 2022; Hsu and Calmon 2022) in machine learning theory suggests that this need not be a concern. For many tasks, there exists a large set of models that achieve comparable performance. Because the Rashomon set is broad, there is ample room to guide optimization toward a subset of theory-consistent solutions—without sacrificing performance. This idea closely parallels the use of regularization in classical machine learning. Regularization constrains the hypothesis space to favor simpler or more interpretable solutions, yet often improves generalization rather than hurting it. In our framework, validated hypotheses serve as soft constraints that regularize fine-tuning, steering the model toward solutions that are not only high-performing but also semantically meaningful. This improves both robustness and interpretability.

Empirical Evaluation

We evaluate our framework on data extracted from the Upworthy research repository and crawled from the internet, which contains 4,502 news articles and 23,437 A/B-tested headlines. We compare against off-the-shelf pretrained LLMs, supervised fine-tuning, and standard Direct Preference Optimization (Vanilla DPO).

We find that knowledge guidance achieves consistent improvement over vanilla DPO, while outperforming pretrained LLMs. While under certain parameter settings, vanilla DPO attains catchiness scores close to ours, vanilla DPO achieves the seemingly high scores by overusing clickbait-style words and phrases—classic signs of reward hacking (Skalse et al. 2022)— while also reducing vocabulary diversity. Meanwhile, human evaluators, by contrast, find vanilla DPO's outputs only comparable to the original Upworthy headlines and lower than those produced by our method, revealing that its apparent gains stem mainly from exploiting clickbait rather than delivering genuine improvements.

We also evaluate the value of knowledge under different amounts of available training data. We find that LLM knowledge provides the largest gains in low-data settings with a limited number of content experiments. This is valuable in practice as A/B tests needed to measure relative consumer preferences or clickthrough rates for content are typically costly and time-consuming.

To examine how much data the reasoner LLM needs to generate high-quality hypotheses, we vary the mini-batch size b and evaluate the generalization performance of the resulting hypotheses. As expected, the hypothesis distribution drifts progressively away from the zero-shot prior as b increases, as measured by the KL divergence. When b is too small, hypotheses remain close to the prior and yield generic, underfitted explanations with high empirical error. When b is too large, hypotheses overfit to batch-specific idiosyncrasies, reflected in high KL divergence and rising error. This pattern reveals a "sweet spot" in batch size where hypotheses are specific enough to reduce error but not so narrowly anchored that they lose generalizability. We interpret this result through the lens of PAC-Bayesian theory, which provides high probability generalization bounds for data dependent distributions over hypotheses.

Finally, our framework naturally extends to multi-objective fine-tuning, such as optimizing for both catchiness and relevance (consistency with the source article). Knowledge-based regularization achieves a more favorable trade-off across objectives than adjusting hyperparameters in vanilla DPO, highlighting how hypothesis guidance can balance multiple marketing goals simultaneously.

Our contributions are three-fold. First, we introduce a framework that synthesizes domain knowledge with LLMs by combining abductive hypothesis generation with inductive validation, and show how this knowledge regularizes fine-tuning. Second, we connect our approach to the Rashomon set perspective in machine learning, which highlights a broader principle for theory-guided methods: because many models can achieve similar predictive accuracy, adding domain-based constraints does not necessarily reduce performance but instead helps steer learning away from shortcut solutions and toward models that are more robust and interpretable. Third, we validate the framework empirically, showing in large-scale experiments that knowledge-based alignment not only avoids reward hacking but also improves performance. In particular, we find our knowledge based framework is particularly valuable in settings where there are limited numbers of content experiments. Although we demonstrate these contributions in a content marketing application, the underlying ideas apply broadly to other applications that require Generative AI to be aligned through fine-tuning.

The rest of the paper is organized as follows: Section 2 reviews related literature on theory-guided machine learning, hypothesis generation with LLMs and fine-tuning of LLMs. Section 3 introduces the Upworthy data used for our experiments. Section 4 presents the framework. Section 5 details the experimental design, baselines, the results and their implications. Section 6 concludes.

RELATED WORK

Our work is broadly related to emergent fields on hypothesis generation and validation with LLMs and theory-guided machine learning. We also review related work on aligning LLMs via fine-tuning.

Hypothesis Generation with LLMs and Validation

A hypothesis is a tentative explanation or relationship that can be empirically tested or theoretically evaluated (Kulkarni et al. 2025). Hypothesis generation has long relied on human intuition, manual literature review, and heuristics (Bazgir, Zhang et al. 2025). LLMs allow for a new paradigm to automate both the generation and validation of hypotheses. We review the literature on hypothesis generation with LLMs and validation.

Hypothesis Generation with LLMs. LLMs have been applied to hypothesis generation across biomedicine (Sybrandt, Shtutman, and Safro 2017, 2018; Ghafarollahi and Buehler 2025), materials science (Huang et al. 2024), and the social sciences (Leng, Wang, and Yuan 2024; Tong et al. 2024). Methodologically, there are three emerging paradigms.

First, *corpus-based prompting* methods generate hypotheses by prompting LLMs with text drawn from scientific corpora, relying on pretrained knowledge or retrieved documents. Examples include Crispr-GPT (Huang et al. 2024), VELMA (Schumann et al. 2024), and The AI Scientist (Lu et al. 2024). Second, *knowledge graph–driven inference* methods exploit structured semantic networks, framing hypotheses as novel or underexplored edges. MOLIERE (Sybrandt, Shtutman, and Safro 2017) and SciAgents (Ghafarollahi and Buehler 2025) fall into this class, as do approaches that integrate causal graphs (Xiong et al. 2024; Tong et al. 2024). Third, *simulation- or reward-driven exploration* leverages feedback from simulated environments to refine hypotheses, as in novel materials discovery (Gruver et al. 2024).

Our approach extends this literature with a novel, contrastive instance-level method. We present an LLM with small sets of A/B pairs along with their observed behavioral outcomes (e.g., which headline generated higher click-through) and ask it to provide explanations for the outcomes. By iterating across many such pairs, the model produces a diverse and interpretable set of abductive hypotheses grounded in contrastive empirical outcomes. Unlike prior paradigms, our approach requires neither large corpora, graph structures, nor simulations, and so is very well-suited to domains such as marketing, where user-level behavioral feedback can be obtained.

AI based Hypothesis Validation. Existing validation frameworks draw on three main approaches. *Simulation-based* platforms, such as LabBench (Laurent et al. 2024) or AgentClinic (Schmidgall et al. 2024), test hypotheses in virtual or digital twin environments, particularly in biomedicine and robotics. *Predictive model-based methods* validate hypotheses using statistical fit or causal structure. For instance, posterior predictive checks or Bayesian reasoning frameworks have been used to evaluate whether a hypothesis improves predictive accuracy or aligns with known causal pathways (Tang et al. 2024). *Human-based* validation involves expert review: for example, doctoral researchers or domain scientists rating LLM-generated hypotheses (Tong et al. 2024; Banker et al. 2024; Kumbhar et al. 2025).

Across these methods, validation typically assesses plausibility or internal coherence rather than downstream utility. Our approach shifts the criterion, given our objective of LLM alignment: we embed candidate hypotheses directly into generator prompts and evaluate them by their effect on observable behavioral outcomes (click-through rates). Formulating validation as an optimization problem (via simulated annealing) links hypothesis quality directly to task performance. This represents a shift from validating for plausibility to validating for usefulness.

Theory-Guided Machine Learning Models

In marketing and economics, researchers have long embedded theory into models to improve inference and prediction. Structural models exemplify this tradition by encoding behavioral or economic principles directly into estimation, enabling rich counterfactual analysis. More recently, marketing scholars have also embedded theory into machine learning itself—for example, Fong, Kumar, and Sudhir (2024) designed convolutional filters grounded in acoustic physics to capture musical features, while other work has engineered features from persuasion, sales, or visual perception theories to predict marketplace outcomes (Chakraborty et al. 2024; Zhang et al. 2022). These approaches share a unifying theme: embedding domain knowledge into models enhances interpretability, mitigates spurious correlations, and improves predictive accuracy.

A parallel movement in machine learning has advanced this idea under the umbrella of knowledge-guided machine learning (Karpatne et al. 2017; Karpatne, Jia, and Kumar 2024). The most prominent strand is physics-informed ML, particularly Physics-Informed Neural Networks (PINNs) (Karniadakis et al. 2021), which incorporate governing equations as constraints during training. By embedding such priors directly into optimization, these models achieve data efficiency, interpretability, and robustness even in low-data regimes (Cuomo et al. 2022). Related approaches similarly integrate domain truths or constraints into architectures or objectives—ranging from conservation laws in engineering to fairness constraints in decision-making (Pazzani 1993; Hoffer et al. 2022; Zhang, Du, and Zhang 2022).

Unlike these approaches, which rely on fixed, pre-specified domain theories, our framework introduces a fundamentally different form of theory integration—what we call LLM-synthesized knowledge. Rather than hardwiring equations or hand-crafted features, we leverage large language models' abductive reasoning to propose explanatory hypotheses from preference-labeled data, and then validate them inductively for generalization. This process creates domain-relevant, interpretable knowledge tailored to the task at hand, scalable across contexts where formalized theories are underdeveloped or tacit in expert intuition. In doing so, our framework extends the knowledge-guided ML paradigm beyond domains with mature theories to settings like digital marketing and consumer behavior, where knowledge-guided alignment can potentially yield substantial gains.

Aligning LLMs via Fine-Tuning

Aligning large language models (LLMs) with private datasets is essential for ensuring their utility in organizational contexts and for compliance with privacy and domain-specific requirements. Recent work fine-tunes LLMs to identify customer needs (Timoshenko, Mao, and Hauser 2025), generate email subject lines (Angelopoulos, Lee, and Misra 2024), and predict A/B test outcomes for news headlines (Ye, Yoganarasimhan, and Zheng 2024).

Reinforcement learning approaches have become central to alignment. In reinforcement learning from human feedback (RLHF), a reward model is trained on observed outcomes, and the LLM is optimized to maximize predicted rewards. By contrast, direct preference optimization (DPO) bypasses the reward model: it directly updates parameters using preference rankings. For example, if headline A outperforms headline B in an experiment, DPO increases the likelihood of generating A-like headlines and decreases the likelihood of B-like ones. This direct optimization aligns naturally with A/B testing data, where relative comparisons are abundant but explicit reward signals (e.g., click-through rate differences) may be noisy or unavailable.

DPO is computationally less intensive than RLHF and particularly effective in settings with rich preference data but weak reward structures. Recent work applies DPO for multi-objective optimization, generating engaging news articles while preserving editorial stance (Cheng et al. 2025).

Compared to supervised fine-tuning (SFT), which produces a static model requiring retraining for new data, RLHF and DPO adapt continuously, leveraging ongoing A/B streams to track evolving user preferences.

DATA

Our data come from two resources, the A/B test of headlines from Upworthy Research Archive (Nathan et al.) and supplementary article content data crawled from the internet.

Upworthy A/B Testing Data

We use the A/B testing experiments from Upworthy, a U.S. media publisher known for its innovative use of A/B testing in online media. Upworthy conducted randomized experiments for each article, testing different combinations of headlines and images to identify the most engaging elements. The data were collected from January 24, 2013, to April 30, 2015. During the experiment period, multiple versions of each article's "package" (combinations of headlines and/or images) were created for testing. A package is defined as one treatment or arm for an article and consists of a headline, image, or a combination of both. For each packet, the number of impressions and clicks were recorded.

The dataset includes 150,817 tested packages from 32,487 A/B tests, capturing 538,272,878 impressions and 8,182,674 clicks. Each test is associated with an average of 4.64 packages, and each package receives an average of 3,569 impressions and 54.26 clicks, with a mean click-through rate (CTR) of 1.58%. Within each test, all packages had equal probability of receiving impressions, resulting in nearly uniform impressions across packages. We filter out tests for images and only focus on the headline tests.

Article Contents

Since our goal is to fine-tune an LLM to generate headlines from article contents, we need to assemble training data that include the article contents. However, the original Upworthy Research

Archive lacks the full article bodies. An initial analysis of the complete Upworthy dataset, reveals that these A/B tests originated from a pool of 7,583 unique articles. To achieve our goal, we supplement our dataset by scraping the corresponding article contents for each headline.

For each headline in the experimental subset mentioned above, we systematically attempt to retrieve its corresponding full article content. We employ web scraping techniques to extract the primary textual content from the respective webpages. For articles that include video content, we download the video transcripts and then use GPT-4 to summarize these transcripts, thereby obtaining a cleaner and more concise textual representation of the video's content.

Following content acquisition, we undertake several data cleaning and filtering steps to refine the dataset for our experiments. We exclude articles for which no content could be successfully retrieved. Additionally, several length-based filters are applied to the textual data. Articles are excluded if their main content, when present, is shorter than 150 characters. A similar criterion is applied to cleaned video transcripts; non-empty transcripts shorter than 150 characters result in article removal. For video summaries, those shorter than 140 characters (if non-empty) are also discarded, as our observations indicate that such brief summaries typically correspond to videos with minimal substantive content (e.g., primarily music or lacking meaningful narration). We also assess the combined textual volume from the main content and the video summary. Articles are retained only if this combined length falls inclusively between 200 and 4,000 characters, ensuring they are sufficiently informative for analysis without being excessively long.

After these preprocessing stages, our final dataset for subsequent experiments consists of 4,502 articles associated with 36,188 unique headlines, including 23,437 headlines from headline test pairs with CTR differences significant at $p < 0.05$. The data on each article includes its headline(s), main textual content, and (where applicable) summarized video transcript. Table 1 displays summary statistics of the articles.

Table 1: Dataset Statistics for Experimental Upworthy Articles

Metric	Min	Max	Mean	Std
Content Length (characters)	150	3,984	737.6	691.9
Video Summary Length (characters)	0	620	174.8	229.7
Headlines per Article	1	93	8.21	6.17
Headlines per Article[†]	0	53	5.21	4.71
CTR[†]	0.0000	0.1538	0.0148	0.0121

[†] Headlines from headline test pairs with CTR differences significant at $p < 0.05$.

A GENERAL FRAMEWORK FOR KNOWLEDGE-GUIDED FINE-TUNING

Our knowledge-guided fine-tuning framework consists of two stages. In *Stage I*, we construct a set of structured hypotheses $\mathcal{K}$ that serve as domain-relevant knowledge. This is achieved through an iterative process that combines *abduction* (an LLM proposes candidate explanations from small batches of preference-labeled examples), *induction* (each hypothesis set is tested for generalization across the full training set), and *optimization* (simulated annealing is used to search for high-performing, generalizable sets). In *Stage II*, this knowledge is inserted into the prompt of a generative AI model, which is then fine-tuned using Direct Preference Optimization (DPO), aligning outputs with interpretable and generalizable principles rather than spurious correlations.

Our framework involves two generative AI models with distinct roles. The first, a *reasoner* $\mathcal{R}$, proposes hypotheses and must therefore exhibit strong reasoning capability. Because the hypotheses are expressed in natural language, $\mathcal{R}$ should be able to produce text, typically via an LLM, and it could also be a multimodal LLM that accepts non-textual inputs (e.g., images) and outputs textual hypotheses. The second, a *generator* $\mathcal{G}$, produces task-specific outputs (e.g., headlines) and is fine-tuned via DPO while being conditioned on $\mathcal{K}$. $\mathcal{G}$ must be amenable to parameter-efficient fine-tuning and, in principle, can extend beyond text generation to other unstructured outputs such as images. In this work, we focus on text generation and instantiate $\mathcal{G}$ as an LLM. Although our framework is model-agnostic, in experiments we use `o3`—a state-of-the-art proprietary reasoning model—for $\mathcal{R}$, and `LLaMA-3.2-3b-Instruct`—a publicly available model supporting efficient fine-tuning—for $\mathcal{G}$. The modular design allows alternative choices of $\mathcal{R}$ and $\mathcal{G}$ depending on resources and deployment needs.

We denote the training data as a set of tuples $\mathcal{D} = \{(a^{(i)}, h_w^{(i)}, h_l^{(i)})\}_{i=1}^n$, where each tuple contains an article $a^{(i)}$, a winning headline $h_w^{(i)}$ with higher observed click-through rate (CTR), and a losing headline $h_l^{(i)}$ with lower CTR. Note that each article is typically associated with multiple headline pairs, reflecting a diverse set of A/B tests.

We now describe each stage of the framework in detail.

Stage I — Knowledge Synthesis via Abduction, Induction, and Optimization

Our synthesis procedure builds on Peirce's classical framework of inference (Burks 1946), combining *abduction* (generation of explanatory hypotheses) and *induction* (empirical validation) within a nested *optimization* loop. Together, these components identify a set of generalizable, high-utility hypotheses to form a structured knowledge base $\mathcal{K}$.

Abduction generates hypotheses. In the outer loop, the reasoner $\mathcal{R}$ is prompted with small batches of labeled examples (e.g., headline pairs with CTR outcomes) and proposes a pool of candidate hypotheses—structured, generalizable statements explaining why one output is preferred over another.

Induction evaluates hypotheses. Induction tests the generalization ability of hypotheses. A given set of hypotheses is embedded into the prompt of the generator $\mathcal{G}$ and applied across a broader dataset. The resulting outputs are scored to compute an *inductive utility*, reflecting how well the hypotheses guide $\mathcal{G}$ toward generating more engaging headlines.

Optimization orchestrates the search. Using simulated annealing, the inner loop iteratively proposes and evaluates sets of hypotheses, refining toward those that consistently improve performance. When no further gains are observed—a condition we call *stagnation*—the algorithm returns to the abductive step to refresh the hypothesis pool.

This structured process ensures that the resulting knowledge $\mathcal{K}$ is both data-grounded and empirically validated, making it well-suited as a regularizing prior for fine-tuning. See Figure 1 for an illustration of the full synthesis stage.

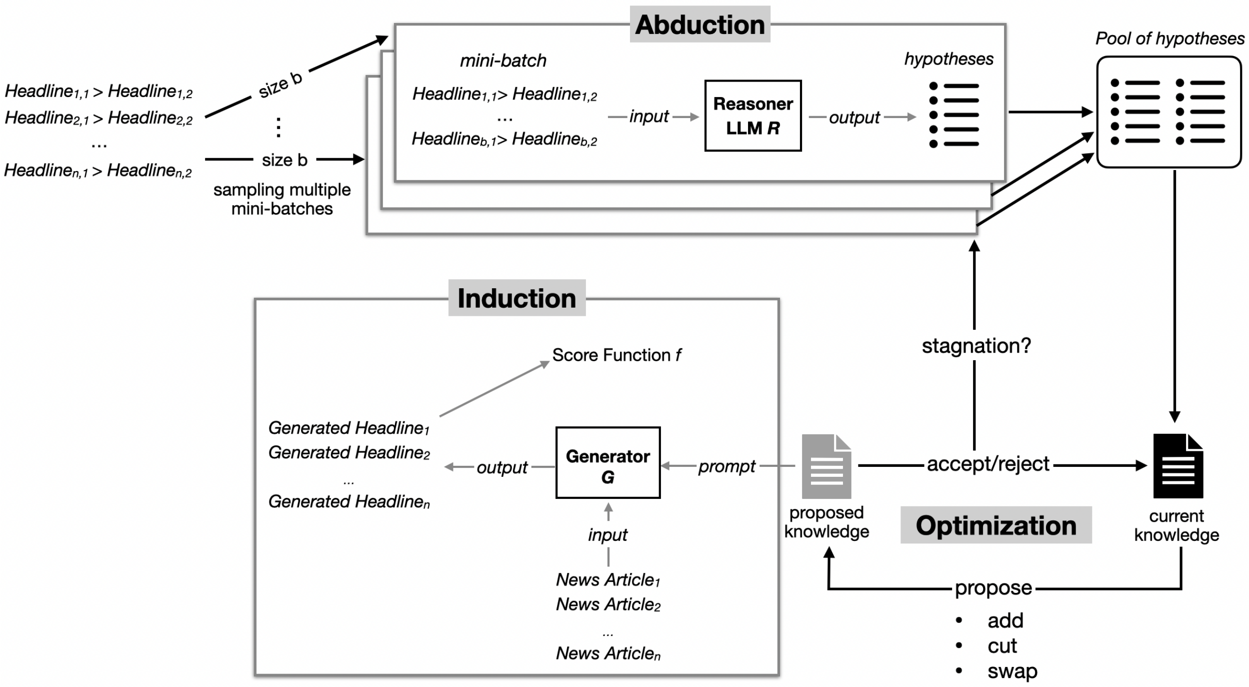


Figure 1: Stage I – Knowledge synthesis via abduction, induction, and optimization.

Now we detail each component.

Hypothesis Generation via Abduction

At outer-loop iteration r, we assemble a *pool* of candidate hypotheses $H^{[r]}$ from the preference-labeled headline pairs, leveraging the knowledge and reasoning capability of the reasoner LLM $\mathcal{R}$. We do this by sampling mini-batches of size b (we choose $b = 20$ in our experiments) from $\mathcal{D}$. The choice of a small b is deliberate: with access to only a handful of examples, $\mathcal{R}$ cannot rely on simple memorization of surface-level correlations (e.g., "add the word *shocking*"). Instead, it must draw on its pretrained knowledge and reasoning capability to infer more generalizable, domain-general explanations for the observed performance differences.

For a mini-batch indexed by c, we construct the prompt $p_{\text{hyp}}\big(\{h_w^{(i)}, h_l^{(i)}\}_{i\in c}\big)$ according to the

You will be shown some pairs of headlines. In a field experiment, the first headline in each pair was proven to receive a higher clickthrough rate than the second one. Your task is to compare the headlines, identify their differences, and create hypotheses to explain why the first headline is better than the second. The hypotheses should highlight the key factors that make the first headline more appealing, without mentioning the position of the headlines. Be creative with the hypotheses.
Format each hypothesis with ”##” at the beginning (do not number them). For example:
## some text
## some text
## some text
...
Only output the hypotheses as formatted, without any additional comments.
1. $\texttt{headline}_1^{(1)} > \texttt{headline}_2^{(1)}$
2. $\texttt{headline}_1^{(2)} > \texttt{headline}_2^{(2)}$
...
20. $\texttt{headline}_1^{(20)} > \texttt{headline}_2^{(20)}$

Figure 2: The prompt template for eliciting hypothese from the reasoner $\mathcal{R}$.

template shown in Figure 2 and pass it to the reasoner $\mathcal{R}$:

$$H_c \;=\; \mathcal{R}\Big(p_{\text{hyp}}\big(\{h_w^{(i)}, h_l^{(i)}\}_{i\in B}\big)\Big).$$

Each call returns a small set of hypotheses for that mini-batch. We show an example of five hypotheses produced from a single batch:

Issue an unapologetically bold thesis that challenges a prevailing stereotype, signaling fearless social commentary worth engaging with.

Spark intrigue by withholding the central detail—name, secret, or quote—so readers must click to satisfy their curiosity.

Put a clearly identifiable protagonist or authority figure up front; names or roles give the story a human face and raise the stakes.

Use visual emphasis devices—asterisks, caps on a single word—to guide the scanning eye to the core intrigue without over-doing it.

Use sensory or witness verbs (“saw,” “captured,” “heard”) that invite readers to observe events rather than merely hear opinions.

Here, mini-batches are sampled at the level of articles. In each iteration, we first select a set of articles, then include the headline pairs associated with them. We implement **adaptive sampling**: articles whose generated headlines scored poorly in the previous round (under the current knowledge) are more likely to be selected in the next round. This adaptive mechanism directs $\mathcal{R}$ toward cases where existing knowledge is weak, encouraging the generation of new hypotheses that better capture the underlying drivers for these harder instances. The above procedure is repeated over multiple mini-batches to populate the hypothesis pool. To avoid redundancy and keep the pool size manageable, we accept a newly generated hypothesis $h \in H_c$ only if it is *novel* with respect to the current pool. We embed every hypothesis with an OpenAI embedding model and define

$$\texttt{novelty}(h) \;=\; \min_{h' \in H^{[r]}} d\big(h, h'\big),$$

where $H^{[r]}$ denotes the current hypotheses pool and $d(\cdot, \cdot)$ is cosine distance. We keep h only if $\texttt{novelty}(h) > \delta$ for a user-chosen threshold δ.

We repeat this mini-batch sampling–generation–filtering cycle until the pool reaches a preset size P, determined by available compute and wall-clock budget. Even with a modest P, the procedure is repeated at subsequent outer-loop iterations, enabling continual exploration of new hypotheses. The resulting pool $H^{[r]}$ is then passed to the inner-loop stage for selection via inductive validation.

Hypothesis Evaluation via Induction

After abduction generates a pool of candidate hypotheses $H^{[r]}$, the goal of the inductive component is to identify a subset that generalizes well across the dataset. Since hypotheses may interact in complex ways and both their content and order matter, selection is not a trivial filtering task. Instead, induction is cast as a utility-guided search for a high-quality *knowledge block* k—an ordered list of hypotheses from $H^{[r]}$—that improves performance when included in the prompt of the generator $\mathcal{G}$.

We formalize this inductive evaluation via a utility function $s(\cdot)$, where a knowledge block k is an ordered list of hypotheses drawn from the current outer-loop pool $H^{[r]}$. An effective k should steer $\mathcal{G}$ toward producing higher-quality outputs. However, the intrinsic quality of k is not directly observable, therefore, we propose to estimate it indirectly: for each article $a^{(i)}$ we prompt $\mathcal{G}$ with k to produce a headline $h^{(i)} = \mathcal{G}(a^{(i)} \mid k)$, then use a pretrained judge model $f(\cdot)$,which estimates the probability that $h^{(i)}$ would outperform the original Upworthy headline. The average score defines the inductive utility:

$$s(k) \;=\; \frac{1}{m}\sum_{i=1}^{m} f\big(h^{(i)}\big), \tag{1}$$

where m is the number of articles in the training data.

This defines induction as testing hypotheses by their *consequences*—selecting those that lead to consistently better outputs.

We train f as a binary classifier on headline pairs from the same A/B test. Each pair is ordered, and the label indicates whether the second headline outperformed the first. The model is trained to output the probability that the second headline will yield higher CTR than the first. During induction, we fix the first input to be the default Upworthy headline associated with the article content we crawled online[2] and the generated headline is the second input. The output of f estimates the likelihood that the generated headline is preferred. This model achieves 80.62% accuracy on held-out evaluation data.

Once the inductive evaluation function $s(k)$ is defined, we turn to the **optimization component**, which seeks the knowledge block that maximizes this score.

Knowledge Formation via Optimization.

The goal of optimization is to identify a high-utility knowledge block k^*—an ordered list of hypotheses—that generalizes well across the dataset. Because the hypothesis pool $H^{[r]}$ is iteratively expanded, the search for k^* unfolds progressively over multiple outer-loop iterations, each

[2] Although multiple headline variants were tested for each article during the original Upworthy experiments, only one version is typically still visible when we crawl the article on the internet.

guided by areas of current failure.

At any point, we maintain a best-so-far knowledge block k^*, which is continuously refined through simulated annealing over the current hypothesis pool $H^{[r]}$. That is, rather than reinitializing from scratch at each iteration, simulated annealing begins from k^* and uses local perturbations to search for improvements within $H^{[r]}$. This cumulative refinement ensures that knowledge improves over time as the pool of available hypotheses expands.

Search via Simulated Annealing. Simulated annealing (SA) is a probabilistic local-search algorithm designed to efficiently explore large combinatorial spaces. Starting from the current solution k_t, each iteration perturbs it to produce a neighbor k', modifying both membership and order. We generate neighbors using three atomic moves:

- `add`: append a randomly chosen hypothesis from $H_{C^{[r]}} \setminus k_t$;
- `cut`: remove a randomly chosen hypothesis from k_t;
- `swap`: exchange the positions of two hypotheses in k_t.

This move set balances exploration of new content (`add`/`cut`) with positional adjustment (`swap`), while keeping each proposal computationally inexpensive.

We then compute the change in the inductive score $\Delta = s(k') - s(k)$, and accepts k' with probability

$$P_{\text{accept}} = \min\{1, \exp(\Delta/T)\}, \tag{2}$$

where $T(t)$ is a temperature parameter that decays over time. The accepting rule ensures that any improvement ($\Delta > 0$) is accepted, while a worsening move ($\Delta < 0$) is accepted with probability $\exp(\Delta/T(t)) \in (0, 1)$. Early in the search $T(t)$ is high, so even substantially worse solutions can be accepted, allowing the algorithm to leap over local optima and explore diverse regions of the space. As the temperature gradually cools, the acceptance probability for negative Δ shrinks, making the search increasingly *greedy*. Eventually, when $T(t) \to 0$, only improving moves are accepted, so the trajectory settles into a basin of high utility. Under a slow schedule, the algorithm is guaranteed to converge in probability to a global optimum (Van Laarhoven et al. 1987; Bertsimas and Tsitsiklis 1993), balancing exploration and exploitation without exhaustive enumeration of the entire $|H^{[r]}|!$ space.

Continual Refresh and Convergence. When the inner optimization loop can no longer improve the objective using the current hypothesis pool $H^{[r]}$, we increment the outer-loop index r and regenerate the hypothesis pool. To focus attention on unexplained cases, the next batch of articles is sampled with probability inversely proportional to current performance (i.e., articles where generated headlines score poorly under f). This adaptive mechanism steers the reasoner $\mathcal{R}$ toward parts of the space where existing knowledge is weak, encouraging new hypotheses.

Ensuring Generalization Beyond Spurious Correlations

Although our procedure involves repeated search for high-scoring hypothesis sets, it differs fundamentally from reward hacking or classical p-hacking. In those cases, models are iteratively tested or tuned on the same dataset until spurious correlations appear significant, leading to overfitting to noise. By contrast, our abductive–inductive framework is explicitly designed to promote generalizability. Hypotheses are generated from small, rotating mini-batches—limiting the opportunity to memorize superficial correlations—and are validated against the broader dataset with a pretrained judge $f(\cdot)$. Importantly, evaluation depends not on how well a hypothesis fits the mini-batch that produced it, but on its ability to improve generalization across unseen articles. The continual refresh of hypothesis pools, along with rejection of redundant candidates, further maintains diversity and prevents collapse onto narrow artifacts. In this way, the process is not an "endless fishing expedition," but a structured cycle of proposing, testing, and refining, designed to converge toward broadly explanatory knowledge rather than exploit chance patterns.

This design also motivates a more formal question: under what conditions can hypotheses generated from small, rotating mini-batches be expected to generalize beyond the examples that produced them? To address this, we later turn to the PAC-Bayesian framework, which provides high-probability guarantees on out-of-sample performance. PAC-Bayes connects generalization to two measurable terms: the empirical fit of hypotheses on observed data and the KL divergence between the prior (the LLM's zero-shot distribution) and the posterior (its updated beliefs after seeing a mini-batch). Intuitively, when the posterior does not drift far from the pretrained prior—keeping the KL penalty small—hypotheses are less likely to overfit local noise and more likely to capture patterns that extend robustly across unseen articles.

Stage II - Knowledge Guided Direct Preference Optimization

Stage I produces knowledge blocks K^* containing explanatory insights distilled from preference data by the advanced reasoner $\mathcal{R}$. Stage II leverages this knowledge to fine-tune the generator $\mathcal{G}$ using *Direct Preference Optimization* (DPO) (Rafailov et al. 2024), a state-of-the-art alignment method that learns from pairwise preferences without requiring an explicit reward model.

In our setting, preference labels are naturally derived from click-through rates (CTR): given a pair of headlines, the one with the higher CTR is treated as the preferred choice. DPO models such preferences using a probabilistic framework inspired by the Plackett-Luce model (Bradley and Terry 1952), which represents the likelihood of one option being preferred over another.

To integrate the knowledge, we embed K^* directly into the prompt alongside the article content, such that the generator conditions not only on the raw input but also on the distilled, semantically meaningful hypotheses. This conditioning mechanism effectively regularizes the fine-tuning process: because the model learns to generate outputs in the context of K^*, its behavior is implicitly shaped by the semantic structure and inductive biases encoded in the knowledge. The gradients computed during training reflect this conditioning, nudging the model toward behaviors that are consistent with K^*. As a result, the model is less likely to overfit superficial correlations in the preference data and more likely to internalize generalizable mechanisms. In this sense, prompting with knowledge narrows the solution space in a meaningful way—acting as a form of soft constraint on generation.

The fine-tuning objective becomes:

$$\mathcal{L}_{\text{kg-DPO}}(\pi_\theta, \pi_{\text{ref}}; K^*) = -\mathbb{E}_{(c,h_w,h_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(h_w \mid c, K^*)}{\pi_{\text{ref}}(h_w \mid c, K^*)} - \beta\log\frac{\pi_\theta(h_l \mid c, K^*)}{\pi_{\text{ref}}(h_l \mid c, K^*)}\right)\right] \tag{3}$$

Here, π_θ denotes the fine-tuned policy we are learning, while π_{ref} anchors training by providing the baseline probabilities before alignment. The term $\sigma(\cdot)$ denotes the sigmoid function. Intuitively, DPO works by contrasting preferred and less-preferred outputs: the objective increases the probability of generating winners while suppressing losers, relative to a fixed reference model. The temperature parameter β controls the strength of these updates—larger values make the model more confident in preference differences but restrict deviation from the reference, whereas smaller values allow greater deviation to fit the data. By conditioning on K^*, fine-tuning is explicitly guided by explanatory knowledge rather than relying solely on statistical imitation of preference patterns. See Figure 3 for an illustration of knowledge-guided DPO.

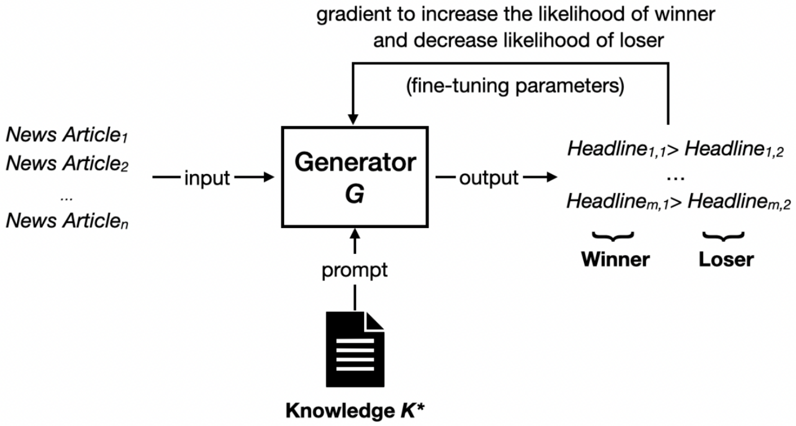

gradient to increase the likelihood of winner
and decrease likelihood of loser
(fine-tuning parameters)
News Article$_1$
News Article$_2$
...
News Article$_n$
input
Generator
G
output
Headline$_{1,1}$> Headline$_{1,2}$
...
Headline$_{m,1}$> Headline$_{m,2}$
Winner
Loser
prompt
Knowledge *K**

Figure 3: Stage II - Knowledge-Guided Direct Preference Optimization

Our knowledge-guided DPO produces a model that not only replicates human-observed choices but also internalizes the underlying *mechanisms* behind them, improving generalization to new contexts. In practice, one can control the model behavior by changing the values of β. The larger β, the smaller deviation is allowed for the new policy to deviate from the original policy, thus less fit to the data but more robust and generalizable. We will later compare the effect of changing β

with that of adding knowledge guidance in the experiments.

The Rashomon Perspective on Knowledge Guidance At first glance, one might worry that incorporating hypotheses introduces an additional constraint. In vanilla DPO, the model only needs to fit observed preference data. With knowledge guidance, however, the model must both fit the data and remain consistent with the mechanisms encoded in the hypotheses. Since constraints in optimization usually shrink the feasible solution space, this might appear to limit performance.

This concern is alleviated by the concept of the Rashomon Set in contemporary machine learning theory (Breiman 2001; Semenova, Rudin, and Parr 2022; Hsu and Calmon 2022). The Rashomon set refers to the collection of all models that achieve similarly high predictive performance on a given task. In many real-world problems with complex, high-dimensional data, this set can be very large: numerous functionally distinct mappings from inputs to outputs yield nearly identical performance. Yet these mappings can differ greatly in terms of their mechanism of achieving the performance. Because the Rashomon set is broad, training may—and as our experiments show, often does—converge to high-performing solutions that exploit shortcuts such as clickbait. These models score well in-sample but generalize poorly.

Knowledge guidance modifies this process by requiring that solutions not only fit outcomes but also align with a validated set of hypotheses about why those outcomes arise. From the Rashomon Set perspective, this amounts to intersecting the large set of high-performing models with a knowledge-consistent subset. The constraint thus serves as a regularizer: it prunes brittle, shortcut-driven mappings while preserving those that remain accurate and semantically grounded. Crucially, because the Rashomon set is large, there is substantial overlap between outcome-accurate models and hypotheses-consistent ones. As a result, performance is not reduced, but becomes more robust. As we will show in the experiments, vanilla DPO attains high scores largely by exploiting shortcuts such as clickbait, while knowledge-guided DPO achieves comparable or even better results without such reliance, reducing reward hacking and improving robustness.

EXPERIMENTS

We conduct a series of experiments using the Upworthy dataset of headline A/B tests to evaluate the effectiveness of our framework. Our analysis proceeds in three stages.

First, we examine whether knowledge synthesized by an LLM provides value in guiding fine-tuning: (i) we test whether knowledge guidance improves click-through rate (CTR), using both model-based evaluation and human judgment (§5.1); (ii) we compare how the performance is achieved by analyzing linguistic characteristics of generated headlines—specifically, the use of clickbait words and lexical diversity—for vanilla DPO and our knowledge-guided DPO (§5.2); (iii) we assess the incremental value of knowledge under varying amounts of available training data, reflecting real-world data constraints (§5.3).

Second, we examine how the number of examples in the mini-batch provided to the reasoner LLM affects hypothesis generation. Specifically, we vary the mini-batch size and evaluate the generalization behavior of the resulting hypotheses using empirical error, vocabulary diversity, and KL divergence (§5.4).

Finally, we test the robustness of our framework in a more complex, multi-objective setting that optimizes for both CTR and relevance. Starting from a multi-objective DPO baseline, we evaluate whether knowledge guidance offers additional benefits beyond standard objective weighting (§5.5).

Does Knowledge Guidance in Fine-Tuning Improve Click-Through Rates?

We evaluate whether incorporating knowledge guidance during fine-tuning improves the generator model's ability to produce high-performing headlines, as measured by a scoring model trained on the CTR data from historical A/B test data.

We fine-tune the generator LLM, `LLaMA 3.2-3b-Instruct`, using our knowledge-guided framework. Following the standard Direct Preference Optimization (DPO) formulation, we construct winner–loser pairs based on observed CTR: given two candidate headlines for the same article, the one with the higher CTR is treated as the preferred choice. The model is then optimized to generate headlines that align with these implicit preferences.

Baselines. We compare our method against two categories of baselines:

(i) *Pretrained models without fine-tuning*. These include the same `LLaMA 3.2-3b-Instruct` model used in our framework, as well as `GPT-4o`, prompted in both zero-shot and few-shot configurations.

(ii) *Fine-tuned models without knowledge guidance*. These include (a) supervised fine-tuning (SFT) using labeled headline pairs, and (b) vanilla DPO fine-tuning using the same winner–loser CTR pairs, but without any hypothesis-based guidance.

Model-based Evaluation To assess headline catchiness at scale, we use a surrogate scoring model that assigns a score between 0 and 1 to each headline. The architecture and training procedure follow the induction model described in Section 4.1, with one key difference: the data used for training.

During training, the scorer $f(\cdot)$ is fit only on the training set to guide hypothesis selection. For evaluation, however, we retrain the model on the *entire* dataset to maximize its accuracy, and refer to this fully trained evaluation model as $\tilde{f}$. This approach ensures the most reliable possible scoring for headline comparisons across methods. Importantly, because $\tilde{f}$ is not used during the training or fine-tuning of any generation models, it also prevents overfitting or "cheating"—ensuring that no model is directly optimized to exploit this evaluation function.

Figure 4 summarizes the average catchiness scores of generated headlines across models. Pretrained models that have not been fine-tuned for engagement—including both `GPT-4o` and `LLaMA 3.2-3b-instruct`—achieve lower scores, as they lack task-specific alignment. Supervised fine-tuning (SFT) improves over these baselines by learning from historical labels, but its average score remains near 0.5 and still trails human-written headlines—reflecting SFT's tendency to replicate patterns rather than optimize preferences. Vanilla DPO outperforms SFT by directly maximizing preference signals derived from CTR. Adding knowledge guidance to DPO boosts performance further, demonstrating the value of theory-guided alignment.

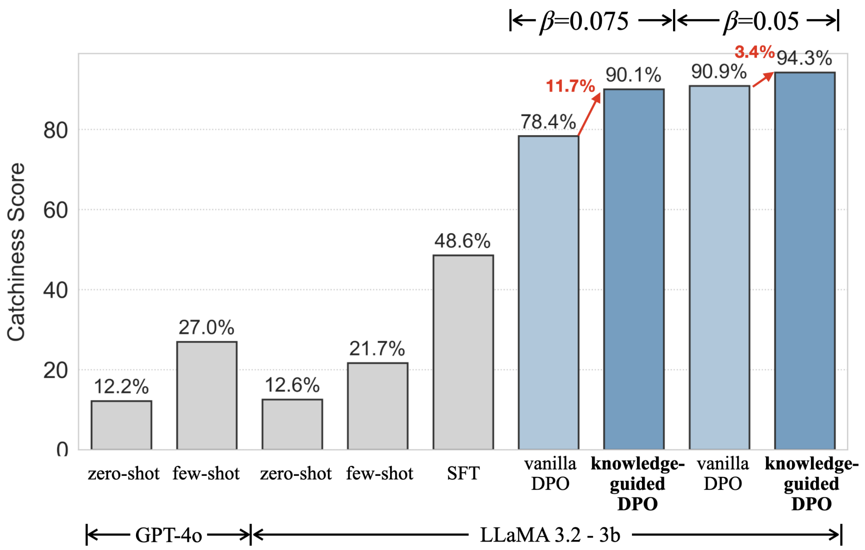


Figure 4: The average catchiness score of headlines generated by different methods

Figure 4 reports two β settings (0.075 and 0.05). In both cases, knowledge-guided DPO boosts catchiness—by 11.7% and 3.4%, respectively. Because a smaller β weakens the regularization term (Rafailov et al. 2024), the model relies more heavily on data; consequently, overall catchiness rises as β decreases.

At first glance, Figure 4 might suggest that vanilla DPO with a low β performs nearly as well as knowledge-guided DPO in terms of catchiness. However, this similarity warrants caution. First, surrogate scores do not always reflect human preferences, so it is important to test whether such gains hold in human evaluation. Second, lower β weakens DPO's regularization (Rafailov et al. 2024), allowing the model to overfit and exploit spurious patterns—boosting catchiness at the cost of relevance or quality.

To probe these risks, we next present (i) a human evaluation of headline preferences and (ii) an analysis of potential reward hacking behaviors.

Human Evaluation We conduct an experiment to compare the click-through rates of headlines generated via three methods:

(i) the original Upworthy headline written by a human (control),

(ii) a headline produced by *vanilla* DPO (treatment 1), and

(iii) a headline produced by *knowledge-guided* DPO (treatment 2).

From the test set, we randomly sampled 100 articles. For each article, we constructed a multiple-choice question with the three headline variants presented in randomized order. See Appendix A for an example of each type of headlines. Each question was rated by 15 participants, yielding head-to-head comparisons across methods.

We recruited 150 participants from the United States through the Prolific platform, which is known for providing high-quality data from a diverse participant pool. Each participant answers 10 questions. Of the 150 who began the survey, 142 completed it, resulting in 1,420 recorded headline choices (142 participants × 10 questions).

Table 2: Results of Human Evaluation for Headline Click-Through Preference

Method	**Total Clicks**	**Percentage of clicks**	**Percentage of Winning**
Knowledge-Guided DPO	540***	38.0%	44.0%
Vanilla DPO	447	31.5%	27.0%
Original Headline	433	30.5%	29.0%
Total	**1420**	**100.0%**	**100.0%**

$^{***}p < 0.001$

Table 2 shows that knowledge-guided DPO headlines attracted the largest share of clicks (38%) and achieved the highest win rate, leading on 44% of articles. In contrast, vanilla DPO headlines (31.5% clicks, 27% wins) performed no better than the original human-written headlines (30.5% clicks, 29% wins), despite being fine-tuned with the data.

Qualitative feedback helps explain this pattern. Several participants noted that while vanilla DPO headlines were “catchy” and evoked curiosity, they often resembled clickbait. This reduced participants’ willingness to click, suggesting that contemporary readers are more attuned—and more averse—to clickbait cues than audiences a decade ago when the original Upworthy experiment was conducted. Recent studies report that online audiences are increasingly wary of clickbait, with some even experiencing “clickbait fatigue” or backlash effects when exposed to exaggerated headlines (D. Molina et al. 2021; Muddiman and Scacco 2019). This helps explain why vanilla DPO, when tuned on engagement data from a decade ago, produces headlines that feel dated and less persuasive to today's readers.

By contrast, knowledge-guided DPO achieves stronger and more generalizable performance because it captures deeper drivers of engagement rather than latching onto surface-level triggers. Knowledge serves as an inductive bias that regularizes model behavior: it prevents overreliance on transient lexical tricks and instead steers the generator toward strategies that remain effective across shifting reader preferences.

Does Knowledge Guidance in Fine-tuning Reduce Reward Hacking?

We next examine whether incorporating knowledge into DPO reduces *reward hacking*—the tendency of models to exploit superficial correlations in the training signal rather than genuinely learning the intended objective. In our setting, this manifests when the model latches onto easily reproducible surface cues (e.g., clickbait phrases) that boost short-term CTR but undermine long-term quality and trust. To provide a stringent test, we focus on the $\beta = 0.05$ setting, where knowledge-guided DPO showed smaller gains in the score provided by $f(\cdot)$, and ask whether its advantages emerge more clearly when evaluating shortcut behavior. We evaluate two complementary aspects of such shortcut exploitation: the reliance on clickbait terms and the overall lexical diversity of generated text.

Clickbait Language Frequency To assess shortcut reliance, we compare headlines generated by DPO with and without theory-based regularization. Prior work has shown that certain clickbait phrases—e.g., *"shocking"*, *"never"*—inflate CTR by triggering curiosity, without necessarily improving substantive quality. We curated a list of such terms by measuring the frequency of the words and cross comparison with phrases from Chakraborty et al. (2016) and report 8 most representative examples for visualization.

We find that Vanilla DPO greatly increased reliance on this type of language. For instance, the usage of "this is" rose from 2% from the original Upworthy data to 56%, and the usage of *"shocking"* rose from 0.7% in the baseline to 43.7% after vanilla DPO fine-tuning, as shown in Figure 5. This pattern arises because, statistically, headlines containing clickbait terms like these

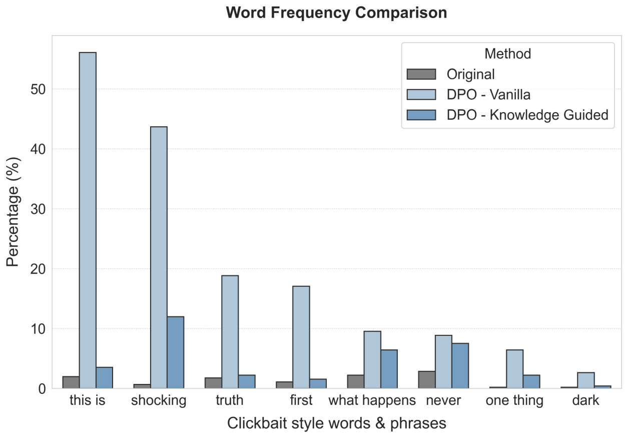


Figure 5: Percentage of headlines using representative clickbait language.

PHRASE	MEAN CTR	% INCREASE	WIN RATE
Baseline	0.0144	–	–
shocking	0.0176	22.2%	72.5%
this is	0.0170	18.2%	60.2%
never	0.0169	17.9%	65.1%
what happens	0.0169	17.5%	67.4%
truth	0.0163	13.1%	65.9%
first	0.0147	2.3%	65.3%
dark	0.0150	4.1%	61.4%
one thing	0.0146	1.4%	59.5%

Table 3: Mean CTRs, percentage increase, and the winrate for headlines containing different phrases.

exhibit a strong positive association with higher CTRs. As an example, we report the average CTR of headlines that contain each phrase in Table 3. It shows that headlines that contain these keywords significantly boost the CTR. For example, headlines that contain “shocking” increase the CTR by 22.2% compared to headlines without the clickbait phrases.

Moreover, these terms are disproportionately favored during DPO training. When examining the preference data, we find that headlines containing clickbait phrases are more frequently labeled as winners. For example, those with *“shocking”* win 72.5% of the time. Since DPO updates parameters to increase the likelihood of winner outputs, this mechanism implicitly amplifies the frequency of such phrases, resulting in a significant boost of appearance as shown in Figure 5.

The above analyses explain how the model overexploits surface-level correlations, a classic case of reward hacking: it maximizes the reward signal by latching onto superficial lexical patterns rather than capturing deeper audience interests. By contrast, knowledge-guided DPO significantly dampened this effect, as shown in Figure 5 - the uses of clickbait phrases significantly dropped to close to the levels in the original headlines. This suggests that the added knowledge regularizes against exploitative shortcuts by steering the model away from over-relying on a single phrase.

While the above analyses focus on a small set of representative clickbait terms, they capture only one facet of shortcut reliance. To obtain a more holistic view of lexical behavior, we next examine the overall distribution of word usage across the entire vocabulary.

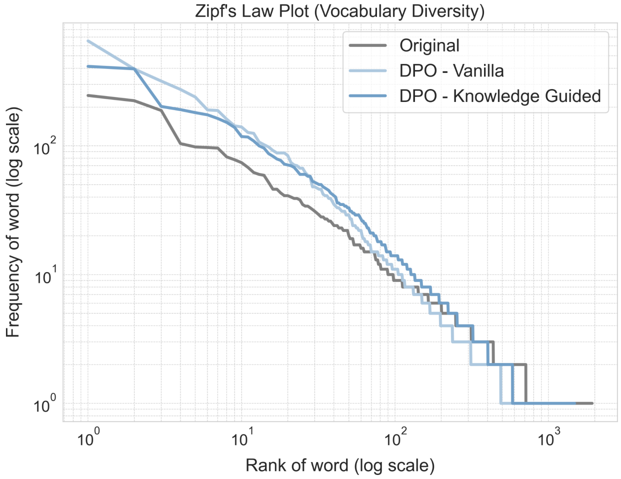


Figure 6: Vocabulary diversity—slower decay reflects richer lexical variety.

Vocabulary Diversity Shortcut exploitation often coincides with reduced linguistic richness, as overuse of a small set of high-impact words yields repetitive outputs. To capture this dimension, we analyze *vocabulary diversity* using a Zipf's Law plot (Gabaix 1999; Piantadosi 2014), which relates word frequency to rank on a log-log scale. A steeper slope indicates over-reliance on a narrow vocabulary, while a flatter slope reflects richer, more varied word usage. In the low-rank (high-frequency) region, the Vanilla DPO curve is highest, indicating heavy concentration on a few common words. As rank increases, it declines steeply, signaling limited use of rare words. Human-written headlines show the slowest decay, reflecting richer vocabulary, while Knowledge-Guided DPO lies in between—more diverse than Vanilla DPO, though not matching human writing. Therefore, instead of mechanically repeating limited triggers, the model learns more nuanced strategies to capture audience attention, and is able to do it even better, as shown in the analyses in §5.1. In other words, with a better understanding of *why* certain headlines perform well, knowledge-guided DPO can achieve the same or even better communicative purposes without resorting to repetitive lexical artifacts.

Connecting Back to the Rashomon Perspective. Taken together, the analyses so far echo the Rashomon set perspective outlined earlier. The Rashomon view emphasizes that there can be many different paths to achieving similar predictive performance. Our results illustrate this directly: knowledge-guided DPO achieves performance that is comparable to, and often exceeds, vanilla DPO—showing that restricting the solution space with validated hypotheses does not reduce predictive power. Yet the paths by which these outcomes are reached differ sharply. Vanilla DPO gravitates toward shortcut solutions, most visibly through overuse of clickbait and reduced lexical diversity, whereas knowledge-guided DPO selects from the overlapping subset of models that are both outcome-accurate and hypothesis-consistent. This shift confirms the Rashomon set intuition: when the feasible set of high-performing models is large, knowledge guidance does not eliminate strong solutions, but instead steers training toward those that are robust, interpretable, and less prone to reward hacking.

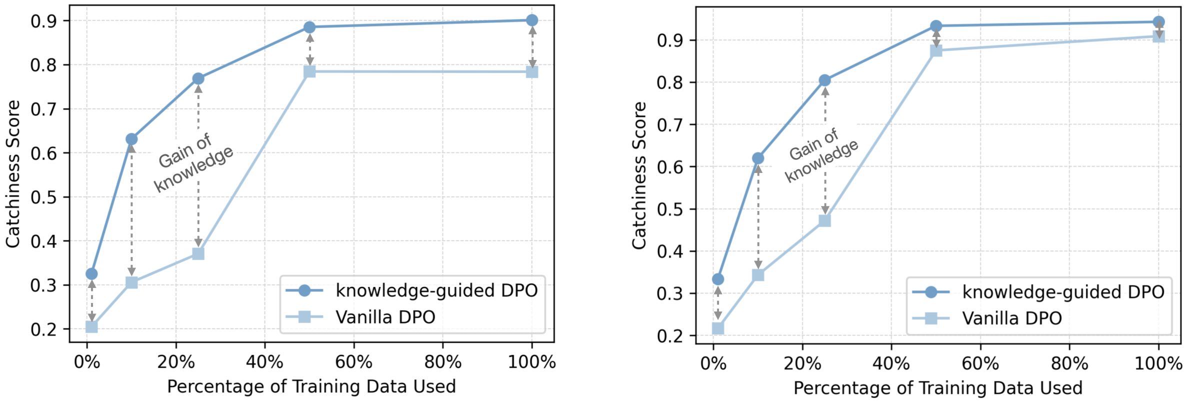
Summary. Both analyses reveal that vanilla DPO exhibits classic reward-hacking behavior; it overuses clickbait language and reduces lexical diversity to artificially boost CTR. By contrast, knowledge-guided DPO curbs these tendencies, yielding headlines that are not only effective in attracting clicks (aligned with marketer objectives), but also with more natural and varied language.

How Does Knowledge Guidance Interact with Data Availability?

We now examine how the value of knowledge guidance varies with the amount of training data. This question is practically important: in many real-world applications, firms face limited access to preference data because A/B testing is costly and time-consuming. If knowledge-guided fine-tuning can compensate for scarce data, it would offer a powerful tool for improving model performance in low-data regimes.

Our framework uses natural-language hypotheses—generated by a strong pretrained LLM—as structured inductive priors that guide fine-tuning. These hypotheses inject semantic information into the model, and their value should be especially pronounced when empirical training data is limited. This raises a natural question: how does the marginal benefit of knowledge guidance vary with the amount of available data? In particular, can knowledge compensate for data scarcity, and if so, when does it offer the greatest gains?

To test this, we train knowledge-guided DPO on progressively smaller fractions of the training set (1%, 10%, 25%, and 50%) and measure its performance relative to vanilla DPO trained on the full dataset (Figure 7). For consistency, the entire framework—including the score function $f(\cdot)$ used in knowledge induction—is trained only on the same partial data, while evaluation relies on a scoring function $\tilde{f}$ trained on the full dataset for comparability. We repeat this analysis for both $\beta = 0.05$ and $\beta = 0.075$.



(a) DPO with and without knowledge guidance ($\beta = 0.075$).

(b) DPO with and without knowledge guidance ($\beta = 0.05$).

Figure 7: Performance comparison with and without knowledge using different amount of training data

The results reveal that the largest gains from knowledge appear when we have a limited number of experimental data for alignment. With as little as 10%–25% of training data, knowledge-guided DPO substantially outperforms vanilla DPO, even approaching its full-data baseline. As the training set grows larger, the performance gap narrows, though knowledge guidance continues to offer modest improvements.

From a practical standpoint, this property is especially valuable. Because preference data such as click-through rates must be collected through user-facing experiments, it accumulates slowly—especially for new products, campaigns, or user segments. The ability to inject strong priors via natural-language hypotheses allows knowledge-guided DPO to deliver robust results even when data is scarce, reducing firms' reliance on exhaustive experimentation and accelerating model deployment.

How Mini-Batch Size Affects Hypothesis Generalization

We now examine how mini-batch size b—the number of preference-labeled examples used to condition the reasoner LLM—affects the generalization quality of its hypotheses. Since each hypothesis is generated via abductive reasoning over b pairs of examples, this design parameter introduces a fundamental trade-off: small batches may yield vague, generic hypotheses due to insufficient structure, while large batches may encourage overfitting to batch-specific artifacts, producing overly narrow or brittle explanations.

Generalization Error To assess generalization, we compute the empirical error of individual hypotheses on a separate hold-out set of headline pairs. Each hypothesis is inserted into the prompt of a small LLM (GPT-4.1-mini), which is then asked to predict the preferred headline for each pair. We record the proportion of incorrect predictions as the empirical error. This quantity serves as a direct proxy for generalization: a low error indicates that the hypothesis encodes a broadly applicable principle, while a high error suggests poor transfer beyond the batch from which it was derived.

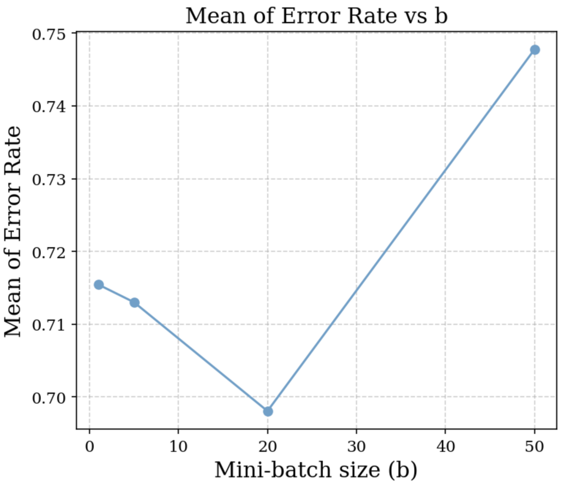


Figure 8: The mean error of hypotheses produced at different mini-batch sizes

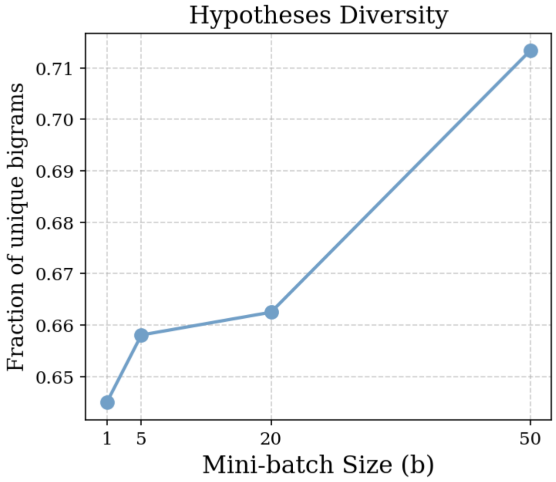


Figure 9: The diversity of hypotheses measure in the fraction of unique bigrams.

Figure 8 reports mean empirical error across mini-batch sizes $b \in \{1, 5, 20, 50\}$. On average, individual hypotheses achieve an accuracy of roughly 30% (error rate $\approx 70\%$). Although this may appear low, it is consistent with the fact that headline appeal arises from heterogeneous factors (e.g., emotional framing, concreteness, novelty), and any single hypothesis typically captures only one of these aspects. This limitation is inherent and intentional: our framework relies on aggregating a diverse set of hypotheses, whose combined coverage spans multiple explanatory dimensions.

The error curve exhibits a U-shaped pattern. At very small b, hypotheses remain close to the zero-shot prior and are often vague or generic, yielding high error. As b increases, the reasoner generates more task-relevant hypotheses and empirical error declines. However, beyond a certain point, further increasing b leads to degradation in generalization. Hypotheses become too tightly coupled to batch-specific artifacts, reflected in a rise in hold-out error.

Lexical Diversity Grows with b. To better understand how mini-batch size shapes hypothesis quality, we also measure lexical diversity using a standard metric: distinct-2 (bigram diversity) (Li et al. 2015). Distinct-2 computes the ratio of unique bigrams to total bigrams in the generated text, capturing how varied and non-repetitive the language is. As shown in Figure 9, diversity increases monotonically with b, indicating that larger batches lead to greater surface-level variation in the generated hypotheses. This reflects the increasing influence of batch-specific signals, allowing the reasoner LLM to move away from generic zero-shot responses toward more customized outputs.

However, this increase in diversity does not necessarily signal better generalization. As shown in Figure 9, generalization error decreases initially but then rises again at large b, even as diversity continues to grow. This divergence suggests that excessive diversity may result from overfitting to batch idiosyncrasies rather than encoding transferable insights. In other words, diversity alone is not a sufficient proxy for quality—what matters is whether that diversity is anchored in broadly applicable principles.

PAC-Bayesian Interpretation. To formalize this tradeoff, we draw on PAC-Bayesian theory, which provides high-probability generalization bounds for data-dependent distributions over hypotheses. These bounds consist of two terms: the empirical risk of the learned distribution and a complexity penalty proportional to $\mathrm{KL}(Q_B|P)$, the Kullback–Leibler divergence between the pos-

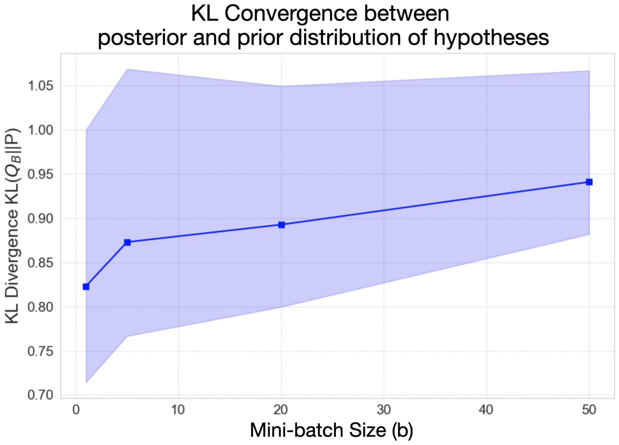


Figure 10: Empirical $\mathrm{KL}(Q_B|P)$ for hypotheses generated the reasoner LLM ($\circ 3$) across mini-batch sizes b, averaged over the total number of clusters set to 10 to 30.

terior Q_B (hypotheses generated after conditioning on batch B) and the data-independent prior P (zero-shot hypotheses). The PAC-Bayes bound thus captures a core tradeoff: increasing b reduces empirical error in-sample but inflates $\mathrm{KL}(Q_B|P)$, weakening generalization guarantees.

Since we cannot access the probability outputs of $\circ 3$, we approximate this KL divergence empirically by embedding generated hypotheses and clustering them to construct discrete distributions over hypothesis space. We vary the total number of clusters from 10 to 30 for robustness of the analyses. For each b, we compute $\mathrm{KL}(\hat{Q}_B|\hat{P})$, where $\hat{Q}_B$ and $\hat{P}$ are empirical posterior and prior distributions derived from cluster frequencies (see Figure 10). As expected, KL divergence grows monotonically with b: small batches yield posteriors close to the prior, while large batches induce stronger shifts in the hypothesis distribution.

Taken together, the empirical error and KL curves reveal a consistent pattern. Small b produces low KL but high error (underfitting), while large b achieves lower in-sample error but at the cost of higher divergence from the prior (overfitting). Moderate batch sizes strike the best balance, producing hypotheses that are both specific enough to improve accuracy and stable enough to generalize. These empirical patterns are consistent with the formal PAC-Bayes bound, which guarantees that the expected out-of-sample loss under Q_B is bounded by its empirical loss plus a penalty term increasing in $KL(Q_B||P)$. We state this result formally in Appendix B.

Composite Objective for Multi-Dimensional Alignment

While catchy headlines can boost click-through rates (CTR), *relevance*—the semantic fidelity between a headline and its corresponding article—is essential for sustaining long-term engagement, fostering user trust, and maintaining platform credibility. Optimizing solely for CTR in preference-based tuning risks prioritizing superficial curiosity over content integrity, since the reward signal focuses exclusively on short-term engagement, and an LLM can drift toward producing attention-grabbing but misleading headlines.

To address this, we replace the single-objective, CTR, with a *composite score*:

$$s(h^{(i)}) \;=\; \text{CTR}(h^{(i)}) + \alpha \cdot r(a^{(i)}, h^{(i)}), \tag{4}$$

where each $r(a^{(i)}, h^{(i)})$ measures the relevance between article content $a^{(i)}$ and headline $h^{(i)}$. The hyperparameter α governs the trade-off: larger values place greater emphasis on relevance, smaller values on catchiness. For each headline pair in DPO, the winner and loser are determined by this composite score rather than CTR alone.

By aligning on a composite multi-dimensional objective, the model learns to produce headlines that are both engaging and faithful to the source content—mitigating reward hacking and promoting sustainable audience satisfaction.

Training a Relevance Scoring Model from Human Labels Because the Upworthy corpus lacks headline–article relevance labels, we annotate them ourselves. We sample 50 articles and generated eight candidate headlines per article—four from *vanilla* DPO and four from *knowledge-guided* DPO at different β—yielding 400 article–headline pairs. Each headline is averagely rated by 10 human annotators on average. The percentage of annotators rating “relevant” is used as the target labels for training a regressor for evaluating the relevance of a headline for a given article. During subsequent training, $r(a^{(i)}, h^{(i)})$ supplies the relevance term in the composite score of Eq. (4), allowing our pipeline to optimize both click-through potential and semantic fidelity *without* additional human annotation at generation time.

Adapting the Framework to Composite Alignment Objective We adapt our knowledge-guided DPO framework with two modifications. First, preference labels are computed from the composite score in Eq. (4), balancing catchiness and relevance rather than optimizing catchiness alone. Second, during hypothesis generation, we augment the abductive prompts to also consider *relevance*: given a batch of articles with their associated headlines and relevance judgments (relevant vs. irrelevant), the LLM is asked to analyze patterns or characteristics that distinguish relevant headlines from irrelevant ones. Thus, the knowledge synthesis step forms a knowledge base that consists of both catchiness-oriented hypotheses and relevance-oriented ones. The optimization via simulated annealing searches from two hypothesis pools to find the optimal two sets and then includes both sets in fine-tuning via DPO.

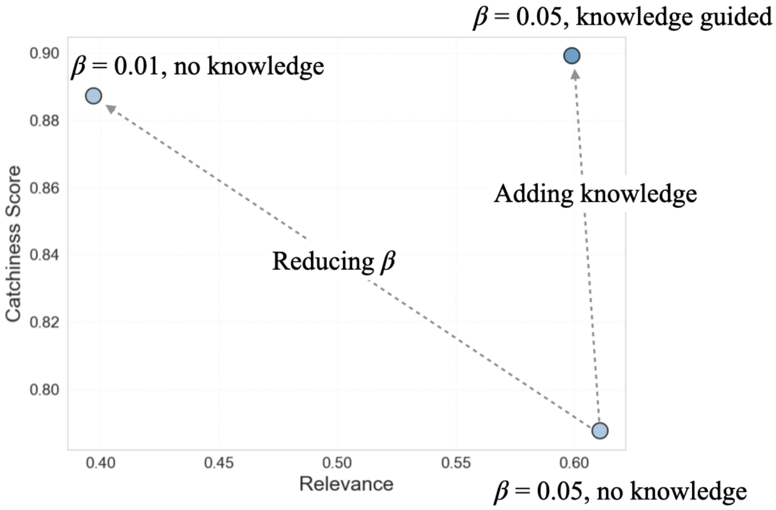


Figure 11: The regularization path for using β versus using knowledge guidance.

Regularization by knowledge vs. by β In vanilla DPO, the hyperparameter β controls how far the fine-tuned policy may deviate from the base policy, implicitly regulating the trade-off between catchiness and relevance. Since pretrained LLMs typically generate highly relevant headlines, a large β leads the model to stay close to the base policy—preserving relevance but limiting gains in catchiness. Reducing β loosens this constraint, allowing greater deviation and higher catchiness, but often at the expense of relevance. In this sense, β acts as a *built-in regularizer*.

We compare this baseline form of regularization to the knowledge regularization in our framework. Starting from the same baseline ($\beta = 0.05$), we increase catchiness in two ways: (1) by reducing β in vanilla DPO, and (2) by keeping β fixed but adding theory-driven knowledge guidance. We then adjust β in vanilla DPO until its catchiness matches that of the knowledge-guided model and compare relevance scores.

As shown in Figure 11, vanilla DPO suffers a steep relevance drop at matched catchiness, while knowledge-guided DPO maintains substantially higher relevance. This reflects the difference in how each method navigates the search space. Vanilla DPO achieves gains by loosening constraints, often leading to superficial shortcuts (e.g., vague teasers, hyperbolic phrases) that decouple headlines from article content. In contrast, knowledge-guided DPO biases generation toward semantically grounded strategies (e.g., highlighting specific benefits, audience-aligned framing) learned through abductive–inductive reasoning.

In effect, knowledge serves as a structured form of regularization; that improves the catchiness of headlines while still preserve relevance, enabling meaningful performance improvements without sacrificing alignment quality.

CONCLUSION

We have introduced a framework for *knowledge-guided alignment* that augments preference-based fine-tuning with *LLM-synthesized knowledge*. Unlike standard alignment methods such as RLHF or DPO, which risk reward hacking and reliance on superficial correlations, our approach uses LLMs to generate and validate hypotheses that function like theory-based constraints. These hypotheses provide interpretable structure, anchoring fine-tuning in principles that generalize across contexts rather than in transient data patterns.

This approach extends the tradition of theory-guided modeling in marketing and economics. Just as structural models constrain estimation with behavioral theory, our method uses LLMs to surface and formalize tacit knowledge that can serve as regularization for machine learning. The key departure from prior work is scalability: instead of relying on labor-intensive manual curation and codification of theory, we leverage pretrained LLMs to synthesize knowledge by inductively validating the LLM's abductive hypotheses based on limited data on the full set of observed data patterns. Our approach is particularly useful in marketing settings such as content marketing when theory may be incomplete and tacit, and therefore difficult to formalize.

The empirical results on the Upworthy headline dataset confirm the value of our framework. Knowledge-guided fine-tuning improves performance on catchiness and relevance, while also mitigating reward hacking behaviors such as excessive clickbait. These gains are particularly strong in low-data (limited number of A/B tests) settings, where validated hypotheses provide guidance that compensates for a limited number of A/B tests for training. This property is practically important: firms often face limited access to training data, since user preference experiments are costly and time-consuming. The ability to inject strong priors via natural-language hypotheses enables managers to accelerate time-to-market while reducing reliance on extensive experimentation. By guiding models toward deeper behavioral drivers, knowledge guidance balances immediate engagement goals with long-term brand trust.

More broadly, our findings highlight that pretrained LLMs encode useful generalizations that can be leveraged not just for content generation, but across applications as a tool for hypothesis generation and validation. By combining abductive generation with inductive validation, we provide a method for aligning generative AI with underlying mechanisms of consumer response. The paradigm is domain-agnostic: it can be applied wherever outcome data are available ("what works") but the drivers behind those outcomes ("why it works") remain latent. For marketing scholars, this opens a path toward more interpretable, transparent, and generalizable applications of AI—linking empirical regularities to the mechanisms that explain them. Ultimately, our work offers both a practical blueprint for firms deploying generative AI and a conceptual advancement that integrates interpretable, theory-driven insights directly into scalable, data-driven methodologies.

REFERENCES

Alquier, Pierre (2021), "User-friendly introduction to PAC-Bayes bounds," *arXiv preprint arXiv:2110.11216*.

Angelopoulos, Panagiotis, Kevin Lee, and Sanjog Misra (2024), "Causal Alignment: Augmenting Language Models with A/B Tests," *Available at SSRN*.

Banker, Sachin, Promothesh Chatterjee, Himanshu Mishra, and Arul Mishra (2024), "Machine-assisted social psychology hypothesis generation.," *American Psychologist*, 79 (6), 789.

Bazgir, Adib, Yuwen Zhang et al. (2025), "Agentichypothesis: A survey on hypothesis generation using llm systems," *Towards Agentic AI for Science: Hypothesis Generation, Comprehension, Quantification, and Validation*.

Bertsimas, Dimitris and John Tsitsiklis (1993), "Simulated annealing," *Statistical science*, 8 (1), 10–15.

Bradley, Ralph Allan and Milton E Terry (1952), "Rank analysis of incomplete block designs: I. The method of paired comparisons," *Biometrika*, 39 (3/4), 324–345.

Breiman, Leo (2001), "Statistical modeling: The two cultures (with comments and a rejoinder by the author)," *Statistical science*, 16 (3), 199–231.

Burks, Arthur W (1946), "Peirce's theory of abduction," *Philosophy of science*, 13 (4), 301–306.

Catoni, Olivier (2007), "PAC-Bayesian supervised classification: the thermodynamics of statistical learning," *arXiv preprint arXiv:0712.0248*.

Chakraborty, Abhijnan, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly "Stop clickbait: Detecting and preventing clickbaits in online news media," "2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)," pages 9–16, IEEE (2016).

Chakraborty, Ishita, Khai Chiong, Howard Dover, and K Sudhir (2024), "Can AI and AI-Hybrids detect persuasion skills? Salesforce hiring with conversational video interviews," *Marketing Science*.

Cheng, Mengjie, Elie Ofek, Hema Yoganarasimhan et al. (2025), "Balancing Engagement and Polarization: Multi-Objective Alignment of News Content Using LLMs," *arXiv preprint arXiv:2504.13444*.

Cuomo, Salvatore, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli (2022), "Scientific machine learning through physics–informed neural networks: Where we are and what's next," *Journal of Scientific Computing*, 92 (3), 88.

D. Molina, Maria, S Shyam Sundar, Md Main Uddin Rony, Naeemul Hassan, Thai Le, and Dongwon Lee "Does clickbait actually attract more clicks? Three clickbait studies you must read," "Proceedings of the 2021 CHI conference on human factors in computing systems," pages 1–19 (2021).

Fong, Hortense, Vineet Kumar, and K Sudhir (2024), "A Theory-Based Explainable Deep Learning Architecture for Music Emotion," *Marketing Science*.

Gabaix, Xavier (1999), "Zipf's law for cities: an explanation," *The Quarterly journal of economics*, 114 (3), 739–767.

Ghafarollahi, Alireza and Markus J Buehler (2025), "SciAgents: automating scientific discovery through bioinspired multi-agent intelligent graph reasoning," *Advanced Materials*, 37 (22), 2413523.

Gruver, Nate, Anuroop Sriram, Andrea Madotto, Andrew Gordon Wilson, C Lawrence Zitnick, and Zachary Ulissi (2024), "Fine-tuned language models generate stable inorganic materials as text," *arXiv preprint arXiv:2402.04379*.

Hoffer, Johannes G, Andreas B Ofner, Franz M Rohrhofer, Mario Lovrić, Roman Kern, Stefanie Lindstaedt, and Bernhard C Geiger (2022), "Theory-inspired machine learning—towards a synergy between knowledge and data," *Welding in the World*, 66 (7), 1291–1304.

Hsu, Hsiang and Flavio Calmon (2022), “Rashomon capacity: A metric for predictive multiplicity in classification,” *Advances in Neural Information Processing Systems*, 35, 28988–29000.

Huang, Kaixuan, Yuanhao Qu, Henry Cousins, William A Johnson, Di Yin, Mihir Shah, Denny Zhou, Russ Altman, Mengdi Wang, and Le Cong (2024), “Crispr-gpt: An llm agent for automated design of gene-editing experiments,” *arXiv preprint arXiv:2404.18021*.

Karniadakis, George Em, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang (2021), “Physics-informed machine learning,” *Nature Reviews Physics*, 3 (6), 422–440.

Karpatne, Anuj, Gowtham Atluri, James H Faghmous, Michael Steinbach, Arindam Banerjee, Auroop Ganguly, Shashi Shekhar, Nagiza Samatova, and Vipin Kumar (2017), “Theory-guided data science: A new paradigm for scientific discovery from data,” *IEEE Transactions on knowledge and data engineering*, 29 (10), 2318–2331.

Karpatne, Anuj, Xiaowei Jia, and Vipin Kumar (2024), “Knowledge-guided machine learning: Current trends and future prospects,” *arXiv preprint arXiv:2403.15989*.

Kulkarni, Adithya, Fatimah Alotaibi, Xinyue Zeng, Longfeng Wu, Tong Zeng, Barry Menglong Yao, Minqian Liu, Shuaicheng Zhang, Lifu Huang, and Dawei Zhou (2025), “Scientific hypothesis generation and validation: Methods, datasets, and future directions,” *arXiv preprint arXiv:2505.04651*.

Kumbhar, Shrinidhi, Venkatesh Mishra, Kevin Coutinho, Divij Handa, Ashif Iquebal, and Chitta Baral “Hypothesis Generation for Materials Discovery and Design Using Goal-Driven and Constraint-Guided LLM Agents,” “Findings of the Association for Computational Linguistics: NAACL 2025,” pages 7524–7555 (2025).

Laurent, Jon M, Joseph D Janizek, Michael Ruzo, Michaela M Hinks, Michael J Hammerling, Siddharth Narayanan, Manvitha Ponnapati, Andrew D White, and Samuel G Rodriques (2024), “Lab-bench: Measuring capabilities of language models for biology research,” *arXiv preprint arXiv:2407.10362*.

Leng, Yan, Hao Wang, and Yuan Yuan (2024), “Llm-Assisted Hypothesis Generation and Graph-Based Evaluation,” *Available at SSRN 4948029*.

Li, Jiwei, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan (2015), “A diversity-promoting objective function for neural conversation models,” *arXiv preprint arXiv:1510.03055*.

Lu, Chris, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha (2024), “The ai scientist: Towards fully automated open-ended scientific discovery,” *arXiv preprint arXiv:2408.06292*.

McAllester, David A “PAC-Bayesian model averaging,” “Proceedings of the twelfth annual conference on Computational learning theory,” pages 164–170 (1999).

Muddiman, Ashley and Joshua Scacco (2019), “Clickbait content may not be click-worthy,” *Center for Media Engagement*.

Nathan, Matias J, Kevin Munger, Marianne Aubin Le Quere, and Charles Ebersole (????), “The Upworthy Research Archive, a time series of experiments in US media,” *Nature: Scientific Datasets. https://doi.org/10.1038/s41597-021-00934-7 PMID*, 34341340.

Pazzani, Michael (1993), “Learning causal patterns: Making a transition from data-driven to theory-driven learning,” *Multistrategy Learning: A Special Issue of MACHINE LEARNING*, pages 65–86.

Piantadosi, Steven T (2014), “Zipf’s word frequency law in natural language: A critical review and future directions,” *Psychonomic bulletin & review*, 21, 1112–1130.

Rafailov, Rafael, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn (2024), “Direct preference optimization: Your language model is secretly a reward model,” *Advances in Neural Information Processing Systems*, 36.

Ruan, Yixiang, Chenyin Lu, Ning Xu, Yuchen He, Yixin Chen, Jian Zhang, Jun Xuan, Jianzhang Pan, Qun Fang, Hanyu Gao et al. (2024), "An automatic end-to-end chemical synthesis development platform powered by large language models," *Nature communications*, 15 (1), 10160.

Schmidgall, Samuel, Rojin Ziaei, Carl Harris, Eduardo Reis, Jeffrey Jopling, and Michael Moor (2024), "AgentClinic: a multimodal agent benchmark to evaluate AI in simulated clinical environments," *arXiv preprint arXiv:2405.07960*.

Schumann, Raphael, Wanrong Zhu, Weixi Feng, Tsu-Jui Fu, Stefan Riezler, and William Yang Wang "Velma: Verbalization embodiment of llm agents for vision and language navigation in street view," "Proceedings of the AAAI Conference on Artificial Intelligence," Vol. 38., pages 18924–18933 (2024).

Semenova, Lesia, Cynthia Rudin, and Ronald Parr "On the existence of simpler machine learning models," "Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency," pages 1827–1858 (2022).

Skalse, Joar, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger (2022), "Defining and characterizing reward gaming," *Advances in Neural Information Processing Systems*, 35, 9460–9471.

Sybrandt, Justin, Michael Shtutman, and Ilya Safro "Moliere: Automatic biomedical hypothesis generation system," "Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining," pages 1633–1642 (2017).

Sybrandt, Justin, Micheal Shtutman, and Ilya Safro "Large-scale validation of hypothesis generation systems via candidate ranking," "2018 IEEE International Conference on Big Data (Big Data)," pages 1494–1503, IEEE (2018).

Tang, Xiangru, Qiao Jin, Kunlun Zhu, Tongxin Yuan, Yichi Zhang, Wangchunshu Zhou, Meng Qu, Yilun Zhao, Jian Tang, Zhuosheng Zhang et al. (2024), "Prioritizing safeguarding over autonomy: Risks of llm agents for science," *arXiv preprint arXiv:2402.04247*.

Timoshenko, Artem, Chengfeng Mao, and John R Hauser (2025), "Can Large Language Models Extract Customer Needs as well as Professional Analysts?," *arXiv preprint arXiv:2503.01870*.

Tong, Song, Kai Mao, Zhen Huang, Yukun Zhao, and Kaiping Peng (2024), "Automating psychological hypothesis generation with AI: when large language models meet causal graph," *Humanities and Social Sciences Communications*, 11 (1), 1–14.

Van Laarhoven, Peter JM, Emile HL Aarts, Peter JM van Laarhoven, and Emile HL Aarts (1987), *Simulated annealing* Springer.

Xiong, Guangzhi, Eric Xie, Amir Hassan Shariatmadari, Sikun Guo, Stefan Bekiranov, and Aidong Zhang (2024), "Improving scientific hypothesis generation with knowledge grounded large language models," *arXiv preprint arXiv:2411.02382*.

Ye, Zikun, Hema Yoganarasimhan, and Yufeng Zheng (2024), "LOLA: LLM-Assisted Online Learning Algorithm for Content Experiments," *arXiv preprint arXiv:2406.02611*.

Zhang, Shunyuan, Dokyun Lee, Param Vir Singh, and Kannan Srinivasan (2022), "What makes a good image? Airbnb demand analytics leveraging interpretable image features," *Management Science*, 68 (8), 5644–5666.

Zhang, Xiaohui, Qianzhou Du, and Zhongju Zhang (2022), "A theory-driven machine learning system for financial disinformation detection," *Production and Operations Management*, 31 (8), 3160–3179.

APPENDIX

A. EXAMPLE QUESTIONS IN THE HUMAN EVALUATION

Source	Headline
Original	I Never Knew American Healthcare Was A Lottery Till I Saw What This Guy Had To Say
Vanilla DPO	The Shocking, Unbelievable Truth About the Broken American Healthcare System Exposed in One Jaw-Dropping Video
Theory-guided	A Single Doctor's Unflinching Look at America's $3.8 Trillion Health Disaster
Original	The New High-Tech Medical Device That's Changing Lives For Little Money
Vanilla DPO	A Revolutionary Device Is Implanting a New Standard of Immortality, One Person at Assistant
Theory-guided	A Tiny Implant Lets People with Paralysis Type a Message with Their Brain

Table 4: Examples question choices in the human evaluation.

B. PAC-BAYESIAN PERSPECTIVE ON HYPOTHESIS GENERALIZATION

We apply standard PAC-Bayesian theory to analyze the generalization properties of natural language hypotheses generated by large language models (LLMs). In our framework, each hypothesis is abductively proposed from a small batch of preference-labeled headline pairs. A central question is whether such hypotheses generalize beyond the mini-batches that produced them. The PAC-Bayesian framework (McAllester 1999) offers a principled lens for answering this question, by bounding out-of-sample loss in terms of in-sample error and the divergence from a prior.

Classical PAC-Bayes Setting. Let $\mathcal{X}$ be an input space and $\mathcal{Y}$ an output space. A predictor $f : \mathcal{X} \to \mathcal{Y}$ incurs a loss $\ell(f(x), y)$ on data (x, y). The generalization risk is

$$\mathbb{E}_{(X,Y)\sim D}\big[\mathcal{L}(f(X), Y)\big],$$

and its empirical counterpart is the sample average over S.

PAC-Bayes considers a distribution Q over a hypothesis space $\mathcal{H}$ (here, a space of predictors), together with a data-independent prior P. After observing data, one may form a data-dependent posterior Q. We assume absolute continuity $Q \ll P$ so that $\mathrm{KL}(Q|P)$ is finite. This divergence quantifies the "cost" of moving from prior to posterior, and appears as a complexity term in PAC-Bayes bounds.

Adaptation to Hypothesis Generation. In our setting, each input $x = (z_1, z_2)$ is a pair of headlines and the label $y \in <, >$ indicates which has higher CTR, yielding a binary classification problem. Let $\mathcal{H}$ be the space of natural-language hypotheses that attempt to predict the winner from a pair. For $h \in \mathcal{H}$ and x, define the loss $\mathcal{L}(h, x) \in [0, 1]$, where 0 indicates correct prediction and 1 indicates error.

The prior P is induced by the pretrained LLM in zero-shot mode, while conditioning on a mini-batch B of b examples yields a posterior Q_B, the distribution over hypotheses proposed after observing B. Thus, $\mathrm{KL}(Q_B|P)$ measures the extent to which conditioning on B shifts the hypothesis distribution away from the pretrained prior.

Generalization Bound (from Standard PAC-Bayes Theorem)

As a starting point, we recall the classical PAC-Bayes bound (e.g., McAllester 1999; Catoni 2007; Alquier 2021) —expressed in our notation—to help in interpreting our empirical patterns.

Theorem 1 (PAC-Bayes Bound)**.** *Let $\mathcal{H}$, $\mathcal{L}$, S, D, and P be as defined above. For any fixed mini-batch $B \subset S$ of size b, and any posterior distribution Q_B over hypotheses generated from B, the following holds with probability at least $1-\delta$ over the draw of sample S of size n:*

$$\mathbb{E}_{h\sim Q_B}\,\mathbb{E}_{x\sim D}\big[\mathcal{L}(h,x)\big] \leq \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{h\sim Q_B}\big[\mathcal{L}(h,x_i)\big] + \sqrt{\frac{\mathrm{KL}(Q_B\|P)+\log\frac{2\sqrt{n}}{\delta}}{2n}}$$

This states that, with high probability, the generalization loss under Q_B does not exceed its empirical loss on S by more than a complexity term that grows with $\mathrm{KL}(Q_B\|P)$ and shrinks with n. This bound states that the expected generalization loss of the hypotheses sampled from Q_B is controlled by (i) their empirical loss on the observed data, and (ii) their divergence from the prior P. The latter term acts as a regularizer, penalizing posteriors that deviate too far from the prior and thus potentially overfit.

Interpretation. This classical PAC-Bayes bound provides theoretical justification for our empirical finding that small mini-batches produce generic, underfit hypotheses (high error, low KL), while large batches produce more specific but potentially overfit hypotheses (lower error, high KL). The tradeoff between these regimes, observed in Figure 8 and Figure 10, aligns with the structure of the bound above: increasing empirical fit (lower in-sample error) must be balanced against increasing divergence from the prior. The "sweet spot" in mini-batch size corresponds to a region where this tradeoff is optimized.