Prompt Adaptation as a Dynamic Complement in Generative AI Systems*

Eaman Jahani
University of Maryland

Benjamin S. Manning
MIT

Joe Zhang
Stanford University

Hong-Yi TuYe
MIT

Mohammed Alsobay
MIT

Christos Nicolaides
University of Cyprus

Siddharth Suri†
Microsoft Research

David Holtz†
University of California, Berkeley

April 18, 2025

Abstract

As generative AI systems rapidly improve, a key question emerges: How do users keep up—and what happens if they fail to do so. Drawing on theories of dynamic capabilities and IT complements, we examine *prompt adaptation*—the adjustments users make to their inputs in response to evolving model behavior—as a mechanism that helps determine whether technical advances translate into realized economic value. In a preregistered online experiment with 1,893 participants, who submitted over 18,000 prompts and generated more than 300,000 images, users attempted to replicate a target image in 10 tries using one of three randomly assigned models: DALL-E 2, DALL-E 3, or DALL-E 3 with automated prompt rewriting. We find that users with access to DALL-E 3 achieved higher image similarity than those with DALL-E 2—but only about half of this gain (51%) came from the model itself. The other half (49%) resulted from users adapting their prompts in response to the model's capabilities. This adaptation emerged across the skill distribution, was driven by trial-and-error, and could not be replicated by automated prompt rewriting, which erased 58% of the performance improvement associated with DALL-E 3. Our findings position prompt adaptation as a dynamic complement to generative AI—and suggest that without it, a substantial share of the economic value created when models advance may go unrealized.

*We thank Vivian Liu for early contributions to this project. The authors are also grateful to Ethan Mollick, Nicholas Otis, Solene Delecourt, Rembrand Koning, Daniel Rock, Emma Wiles, Sonia Jaffe, Jake Hofman, and Benjamin Lira Luttges for their feedback. We have benefited from seminar and conference feedback at MIT CODE, UC Berkeley, Microsoft, and the World Bank. **Author contributions:** E.J., S.S., and D.H. led, directed, and oversaw the project; J.Z. designed and built the online experiment apparatus; B.S.M. led the design of the online experiment flow and Qualtrics survey; J.Z. led the prompt replay process; M.A. led the analysis of prompt text, with contributions from H.T. and E.J.; E.J. and H.T. led all other data analysis and engineering, with contributions from J.Z., D.H., and M.A.; B.S.M., S.S, and D.H. led the writing of the manuscript; and all authors contributed to designing the research and writing the manuscript and supplementary information. **Author declarations:** S.S. is currently an employee of Microsoft. M.A. is currently a paid intern at Microsoft. D.H. was formerly a paid intern at Microsoft, and is currently a visiting researcher at Microsoft. E.J. was supported by NSF grant #1745640. The authors gratefully acknowledge research funding from Microsoft. All of the "target images" for our study were collected from Unsplash, Reshot, Shopify, Pixabay, or Gratisography; all of these images have licenses for free use for commercial and noncommercial purposes. This study was reviewed by the UC Berkeley Committee for Protection of Human Subjects (CPHS) under Protocol 2023-06-16480.

†To whom correspondence may be addressed. Email: `dholtz@haas.berkeley.edu` or `suri@microsoft.com`.

1 Introduction

Generative AI is being integrated into work practices across the economy (Bright et al. 2024, Zhang and Kamel Boulos 2023), yielding notable productivity gains in tasks as diverse as software development, writing, and material discovery (Brynjolfsson et al. 2023, Dell'Acqua et al. 2023, Noy and Zhang 2023, Peng et al. 2023, Toner-Rodgers 2024, Yu 2024). Recent research points to even greater potential ahead, demonstrating advances in automating core scientific processes (Manning et al. 2024), including tasks as complex as chemical research and proving mathematical theorems (Boiko et al. 2023, Romera-Paredes et al. 2024). The adoption of generative AI is also occurring at an unprecedented pace, with recent research showing that approximately 28% of U.S. workers are already using generative AI in their jobs—a rate that significantly outpaces early adoption of personal computers and internet technology at comparable points in their diffusion (Bick et al. 2024, Bright et al. 2024).

As with many other general-purpose technologies, the effectiveness of generative AI depends not only on the technology itself, but on users' ability to craft inputs that produce high-quality results. To interact with generative AI systems, users provide written instructions—often referred to as *prompts*—that guide the model's behavior. These prompts can range from simple commands (e.g., "write a short story about a robot") to highly detailed specifications tailored to particular outputs (e.g., a series of paragraphs instructing an AI system to implement a complete piece of software). In this way, prompting serves as a complementary skill—one that, like spreadsheet modeling in the early PC era, can determine the productivity impact of a given tool (Brynjolfsson and Hitt 2000).

Prompting has quickly become an area of active research and practice. Scholars have developed taxonomies of prompt engineering techniques (Oppenlaender 2023), documented recurring patterns in prompt construction (Schulhoff et al. 2024), and examined how developers embed prompts into software systems (Liang et al. 2024). Other studies have explored prompting strategies for specific applications, including image generation (Don-Yehiya et al. 2023, Xie et al. 2023) and clinical documentation (Yao et al. 2024). In parallel, practitioners have built prompt libraries, shared tutorials, and developed tools to support prompt design. These developments signal a growing consensus that prompting plays a meaningful role in extracting value from generative AI systems.

Yet despite this consensus, prompting remains understudied as a dynamic practice. Many prompt libraries and tutorials present effective prompts as reusable artifacts. But prompts that work well with one model version may underperform or break entirely with the next (Liang et al. 2024, Meincke et al. 2025). While recent research increasingly views prompting as an adaptive process, empirical evidence remains limited on how these strategies evolve—both as users refine prompts for a single model and as they adjust to model updates—and on how these changes ultimately affect performance. This raises a broader question for individuals and organizations investing in prompting capabilities: Are prompt strategies transferable across model versions, or must they be continually revised to match changing model behavior?

To begin exploring this question, we identify *prompt adaptation*[1] as a measurable behavioral

[1]We use the phrase "prompt adaptation" to refer specifically to changes in user prompting behavior that arise

mechanism through which user-side inputs evolve alongside technical advances. We conceptualize prompt adaptation as a *dynamic complement*—that is, a user capability that adapts in response to changes in a technological system and is critical to realizing the full economic value of system improvements. In contrast to static complements (e.g., fixed training, prompt templates), dynamic complements emerge through situated use with rapid feedback, respond to model-level change, and may be enabled or suppressed by system design.

To assess the role of prompt adaptation in shaping realized performance—and to separate its contribution from the direct effects of model improvement—we draw on data from a pre-registered online experiment with $1,893$ participants. In the study, participants were asked to replicate a target image using prompts submitted to one of three randomly assigned text-to-image models of varying capability: DALL-E 2, DALL-E 3, or DALL-E 3 with an automated large language model (LLM)-based prompt revision. Each participant submitted at least 10 prompts in an effort to reproduce the image as closely as possible, with a large monetary bonus for top performers. By comparing outcomes across arms—and by conducting a posthoc analysis that re-evaluates prompts on alternative models—we estimate the extent to which users adapted their prompts in response to model improvements and how much this adaptation contributed to overall performance.

We find that participants assigned to DALL-E 3 produce significantly more faithful replications than those assigned to DALL-E 2. Importantly, about half of this improvement comes from participants *adapting* their prompts to exploit the new model's capabilities—replaying the old DALL-E 2 prompts on DALL-E 3 yields only about half the total improvement. Furthermore, we find that this prompt adaptation is not limited to advanced "prompt engineers"—participants across the full skill distribution benefit from refining their prompts after seeing how the advanced model responds. Finally, we show that an attempt to automate prompt revision via GPT-4 rewriting does not match the performance achieved by manual human adaptation; in fact, it substantially erodes the gains from DALL-E 3. Together, these findings position prompt adaptation as a key mechanism through which users realize the value of rapidly advancing generative AI systems—and as a concrete example of how digital complements must evolve to keep pace with technological change.

In terms of related literature and additional theory, this experiment builds on work in information systems, emphasizing the importance of dynamic, user-driven complements to digital technologies. Research on IT-enabled dynamic capabilities has shown that the value of new systems depends not only on technical infrastructure but on organizations' ability to reconfigure routines and user behaviors in response to ongoing change (Bharadwaj 2000, Joshi et al. 2010, Teece et al. 1997). Related work on post-adoptive IT use has demonstrated that users often engage only superficially with new systems and that meaningful performance gains tend to emerge only when users experiment with and refine their interaction strategies over time (Jasperson et al. 2005). Recent research on human-AI collaboration further underscores that interface design and task structure shape the degree to which users can learn from and adapt to model behavior (Fügener et al. 2022).

in response to evolving model capabilities—distinct from the broader practice of prompt engineering, which includes static best practices, libraries, and templates.

And the concept of co-evolution has been introduced to describe how humans and generative AI systems jointly adapt over time, forming interdependent capabilities that neither could realize alone (Böhm and Schedlberger 2023).

We also engage with work on general-purpose technologies, which has long emphasized that the productivity gains from technical advances depend on the development of new human and organizational complements (Brynjolfsson 1993, Brynjolfsson and Hitt 2000, Brynjolfsson et al. 2021, David 1990). We conceptualize prompt adaptation as one such complement—emerging through trial-and-error, accessible across the skill distribution, and potentially hindered when automation misaligns with user intent. In this way, prompt adaptation shapes how and when technical improvements translate into downstream economic value.

Finally, the contribution of the current paper is four-fold. First, we offer direct causal evidence that prompting is not a fixed input but a dynamic capability that co-evolves with model behavior (Böhm and Schedlberger 2023). Second, we show how even non-expert users adapt their prompting in response to model improvements, extending research on IT-enabled dynamic capabilities (Bharadwaj 2000, Joshi et al. 2010, Teece et al. 1997). Third, we document how performance gains emerge through user-initiated trial-and-error rather than static best practices, advancing work on post-adoptive IT use (Jasperson et al. 2005). Finally, we demonstrate that automation intended to reduce user effort can—if misaligned with user intent—undermine the very adaptations that generate value, contributing to ongoing research on human–AI interaction (Fügener et al. 2022, Yao et al. 2024).

The remainder of the paper is organized as follows. We begin by presenting a simple conceptual framework that characterizes how output quality evolves with improvements in model capacity and with users' corresponding adjustments in effort. We then introduce an experimental design that closely mirrors this framework and describes the data used in our analysis. Next, we present our empirical findings, including a decomposition of the overall effect into components attributable to model improvements versus prompt adaptation, the impact of automated prompt revision, and heterogeneity across user skill levels. Finally, we compare these empirical results to the model's predictions and conclude by discussing implications for organizations adopting generative AI.

2 Conceptual Framework

As generative AI systems evolve, a user's ability to adapt prompts for improved models can become an important source of realized performance gains. We develop a stylized conceptual framework to formalize how overall output quality depends on both the model's capacity and the user's skill and effort in prompt writing. Our goal is not to create a fully normative model. Rather, we wish to clarify the distinction between improvements directly attributable to the model itself versus those arising from user adaptation as motivation for our experimental design. Additionally, the framework naturally implies predictions about the returns to prompting skill and the heterogeneity of such skill.

Although our empirical setting focuses on image replication, this framework generalizes to other tasks where users interact with generative models—such as text generation, code assistance, or molecule design. In each case, we expect performance to reflect both model capacity and users' efforts to refine their inputs. We present the core findings here; a complete exposition of the framework with formal proofs is provided in Appendix E.1.

2.1 Notation and Problem Setting

Let $\theta \in (0, 1]$ represent the model's capacity to translate prompts into high-fidelity outputs (for instance, how accurately it captures requested details). Let $s \in (0, 1]$ denote a user's baseline skill in prompt engineering, and let $x \geq 0$ be the effort the user expends on writing and refining prompts. Each unit of x incurs a cost $k\,x$ for some $k > 0$, which may reflect the time or cognitive load.

$$Q(\theta, s, x) \;=\; 1 \;-\; e^{-\,\theta\, s\, x},$$

so that users cannot exceed perfect fidelity (quality of 1), and each additional unit of effort yields diminishing returns. The user's utility is then

$$U(\theta, s, x) \;=\; Q(\theta, s, x) \;-\; k\,x.$$

The user chooses their effort such that it maximizes the utility. We assume $\theta s > k$, which implies there is a unique interior optimum $x^*(\theta, s)$

$$x^*(\theta, s) \;=\; \frac{1}{\theta s}\,\ln\!\Big(\frac{\theta s}{k}\Big) > 0, \tag{1}$$

with an optimal quality

$$Q^* \;=\; Q\big(\theta, s, x^*(\theta, s)\big) = 1 - \frac{k}{\theta s} > 0. \tag{2}$$

Two implications immediately follow from equation 2. Optimal quality is increasing with both model capacity

$$\frac{\partial Q^*}{\partial \theta} = \frac{k}{\theta^2 s} > 0 \tag{3}$$

and user skill

$$\frac{\partial Q^*}{\partial s} = \frac{k}{\theta s^2} > 0. \tag{4}$$

Furthermore, the relationship between the changes in these parameters supplies us with a basic prediction, which we can evaluate empirically in our experiment.

Proposition 1. *As model capacity θ improves, the effect of user skill on optimal quality diminishes.*

$$\frac{\partial^2 Q^*}{\partial s \partial \theta} = -\frac{k}{\theta^2 s^2} < 0$$

In other words, improvements in model capacity reduce the performance gap between high and

low-skilled users.

2.2 Decomposition into Model and Prompt Effects

As model capacity θ increases, so does x^* in equation 1, implying that a more capable model not only delivers better output for a fixed prompt but also encourages the user to devote more effort to composing prompts. This distinction highlights both a direct capacity-based effect of an improved model, a *model effect*, and an indirect effect driven by users adapting their prompts to the new model's potential, a *prompting effect*.

To see this more formally, suppose the model is upgraded from capacity θ_1 to θ_2. Let the old prompting effort be $x^*(\theta_1, s)$ and the new (adapted) prompting effort be $x^*(\theta_2, s)$. Define the total improvement in quality ΔQ is

$$\Delta Q \;=\; Q\Big((\theta_2, s, x^*(\theta_2, s)\Big) \;-\; Q\Big(\theta_1, s, x^*(\theta_1, s)\Big).$$

We can decompose ΔQ into

$$\begin{aligned}\Delta Q = &\underbrace{\Big(Q(\theta_2, s, x^*(\theta_1, s)) \;-\; Q(\theta_1, s, x^*(\theta_1, s))\Big)}_{\text{Model Effect } (M)} \\ &+ \underbrace{\Big(Q(\theta_2, s, x^*(\theta_2, s)) \;-\; Q(\theta_2, s, x^*(\theta_1, s))\Big)}_{\text{Prompting Effect } (P)}.\end{aligned} \tag{5}$$

The first term isolates the gain from simply upgrading the model (holding the user's prompt strategy fixed at the old optimum), while the second term represents any additional gain from prompt adaptation. In other words, even if θ improves, failing to adjust how prompts are written could leave substantial performance gains on the table.

This decomposition naturally implies a prediction similar to Proposition 1 for each component of ΔQ that co-evolve with the changes in model capacity and user skill. In particular, when k is small, the following are proved in Appendix E.1.

Proposition 2. *As user skill s improves, the model effect M decreases and the prompting effect P increases.*

Intuitively, when costs are low and model capacity rises, higher-skilled users are already near the upper bound of performance, so the direct model effect they receive may be smaller. In contrast, users who can effectively refine prompts might benefit more from the new capacity, so the prompting effect they receive may be larger. Taken together, these theoretical predictions guide our experimental design, where we randomly vary θ and measure how users adapt.

3 Experiment Design and Methods

To empirically examine whether users do, in fact, adapt their prompts in response to model improvements—and how much this adaptation contributes to overall performance—we conducted a pre-registered online experiment with 2,059 participants on Prolific between December 12 and December 19, 2023.[2] Participants were asked to replicate a target image as closely as possible using a generative AI model, with the goal of assessing how both model capability and prompting behavior influence final outcomes.[3]

3.1 Experimental Setting

Each participant was randomly and blindly assigned to one of three model conditions: DALL-E 2, DALL-E 3 (hereafter "DALL-E 3 (Verbatim)"), or DALL-E 3 with automatic prompt revision ("DALL-E 3 (Revised)"). These models differ not only in technical capability but also in whether they apply hidden large language model (LLM)-based modifications to user prompts. In the DALL-E 2 condition, participants interacted with an earlier-generation model that interprets prompts directly without intermediate rewriting. In contrast, the DALL-E 3 API, by default, forwards user prompts to GPT-4 before image generation. This intermediate GPT-4 step rewrites the prompt—typically by adding detail or restructuring language—before passing it to the image model. This behavior is intended to improve image quality but occurs silently and without user visibility. In the DALL-E 3 (Revised) condition, we allowed this default behavior to proceed unaltered. In the DALL-E 3 (Verbatim) condition, we attempted to suppress GPT-4-based prompt rewriting by prepending a hidden system message instructing the model to leave the user's prompt unchanged. While some rewriting still occurred, the rate and extent of modifications were substantially reduced. In all conditions, participants were unaware that their prompt might be rewritten or modified before image generation.

In addition to model assignment, each participant was independently assigned one of 15 target images. These images were drawn from three broad categories—business and marketing, graphic design, and architectural photography—to represent common use cases for text-to-image generation. All images were sourced from platforms that permit free research use (e.g., Unsplash, Reshot, Shopify, Pixabay, Gratisography)[4]. We built a custom interface resembling ChatGPT, with the assigned target image on the right and a scrollable history of prompts and generated images on the left. Participants were explicitly informed that the model was memoryless: every new prompt was processed independently, carrying no information from previous attempts.

Participants had up to 25 minutes to submit prompts, with a requirement to submit at least 10. They were paid $4 for completing the task, plus a $8 bonus (a 200% increase) if their highest-scoring image was in the top 20% of participants. The median completion time was 22 minutes, implying

[2]Full pre-registration details, including hypotheses and planned analyses, are available in an online repository.

[3]All procedures were approved by an institutional review board (**link removed to preserve author anonymity**) and participants provided informed consent. We will release anonymized data and replication code upon publication.

[4]All of these images have licenses for free use for commercial and noncommercial purposes.

an average hourly wage of about $15. After replicating the target image, participants completed a demographic survey covering age, gender, education, occupation, and self-assessed proficiency in creative writing, programming, and generative AI. We removed from our analyses any participants who did not submit at least 10 prompts, repeated the same prompt five or more times in a row, or failed to complete the post-task survey, resulting in a final sample of 1,893 participants and 18,560 prompts.[5]

3.2 Outcome Definition and Stochastic Generation

The primary outcome in our experiment is the similarity between each participant-generated image and the assigned target image, measured using the cosine similarity of CLIP embeddings (Radford et al. 2021). CLIP (Contrastive Language–Image Pretraining) is a neural network trained to jointly embed images and text into a shared latent space, such that semantically or visually similar items lie close together. By embedding both the target image and each generated image into this space and computing the cosine similarity between them, we obtain a quantitative measure of how closely the generated image matches the target along both visual and conceptual dimensions.

Because the output of each generative model is stochastic, the same prompt can yield different images across attempts. To account for this variability, we generated 10 images for each prompt and computed their cosine similarity to the target image individually. We then averaged these 10 similarity scores to produce an expected quality score for each prompt—the primary outcome variable used in our analysis. As a robustness check, we replicated all analyses using DreamSim, a recently developed alternative to cosine similarity on CLIP embeddings that is based on perceptual similarity and aligns more closely with human judgments.[6] The two measures were highly correlated, and our findings were consistent across both.

3.3 Replay Analysis for Separating Model and Prompting Effects

A central goal of our experiment was to distinguish how much of the performance improvement in image replication stems from using a more capable model versus how much comes from users adapting their prompts. Recall that the language of our conceptual framework shows that the total improvement in output quality from upgrading a generative AI model with capacity θ_1 to a model with higher capacity θ_2 can be written as:

$$\Delta Q = Q\big(\theta_2, s, x^*(\theta_2, s)\big) - Q\big(\theta_1, s, x^*(\theta_1, s)\big).$$

We decompose this change into two parts. These are the model effect, the gain from applying

[5] Although participants were required to submit at least 10 prompts to be included in our analysis, the final prompt count is slightly below 10 × 1,893 because we excluded some prompts due to technical issues (e.g., safety filter triggers, duplicate attempt numbers) and limited our main analysis to each participant's first 10 prompts to mitigate potential selection bias. Full details are provided in Appendix C.4.

[6] DreamSim (Fu et al. 2023) is designed to better capture human perceptions of image similarity than traditional embedding-based methods. Our results are robust to using DreamSim in place of CLIP cosine similarity. Full DreamSim-based analyses can be found in Appendix F.

the same prompts to a better model,

$$M = Q\big(\theta_2, s, x^*(\theta_1, s)\big) - Q\big(\theta_1, s, x^*(\theta_1, s)\big),$$

and the prompting effect, the additional improvement from adapting prompts to take advantage of the more capable model,

$$P = Q\big(\theta_2, s, x^*(\theta_2, s)\big) - Q\big(\theta_2, s, x^*(\theta_1, s)\big).$$

To estimate these components empirically, we conducted an additional analysis using prompts from participants in the DALL-E 2 and DALL-E 3 (Verbatim) conditions. To do so, we took the exact prompts that participants submitted during the experiment and re-submitted (or "replayed") them to *both* their originally assigned model and the alternative model, generating new images in each case.[7] More specifically, prompts written by DALL-E 2 participants—corresponding to $x^*(\theta_1, s)$—were evaluated both on DALL-E 2 and on DALL-E 3, providing an empirical measure of $Q\big(\theta_1, s, x^*(\theta_1, s)\big)$ and $Q\big(\theta_2, s, x^*(\theta_1, s)\big)$, respectively. This comparison isolates the model effect: the improvement in output quality when keeping prompts fixed and simply upgrading the model. To estimate the prompting effect, we compared the quality of reused DALL-E 2 prompts on DALL-E 3—an estimate of $Q\big(\theta_2, s, x^*(\theta_1, s)\big)$—to the quality of prompts originally written by DALL-E 3 participants and evaluated on the same model—an estimate of $Q\big(\theta_2, s, x^*(\theta_2, s)\big)$. This captures the additional improvement from users adapting their prompts to better leverage the capabilities of the more advanced system.

4 Results

In this section, we present our empirical findings on how increased model capacity affects user performance in replicating target images. In particular, we focus on three main questions: (i) whether access to a more capable model (DALL-E 3) improves performance; (ii) how users modify their prompts in response to the improved model; (iii) and how much of the overall performance gain can be attributed to model improvements versus prompt adaptation. Finally, we compare these empirical results to the predictions of our conceptual framework. Our analysis draws on both the main experimental data and the replay data described above.

4.1 Overall Impact of Model Upgrades

We begin by examining whether participants using DALL-E 3 achieve higher performance than those using DALL-E 2 as implied by Equation 3. Figure 1 summarizes these findings. Panel A presents three representative target images and, for each, three generated images drawn from the

[7] Because the replay was conducted several weeks after the original experiment, all prompts were re-evaluated at the same time, using the same infrastructure, regardless of whether they were being run on the original or alternative model. This ensured that any observed differences in output quality could be cleanly attributed to model capacity and prompt adaptation rather than to changes in model behavior over time (i.e., model drift (Chen et al. 2023)).

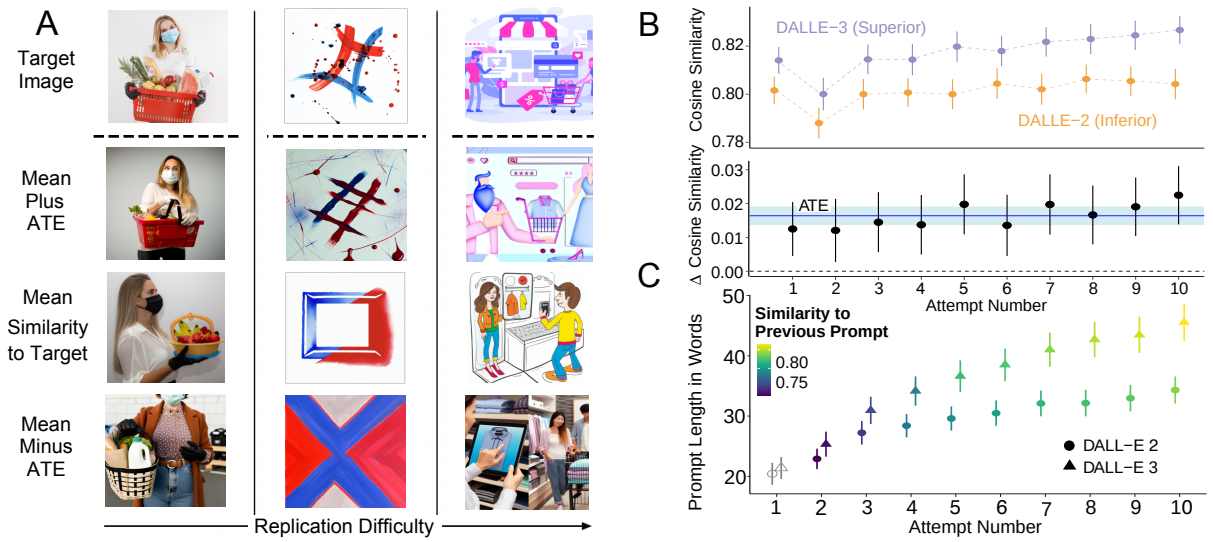


Figure 1: Overall Performance and Prompting Behavior. *(A)* For three example target images, the middle row shows participant-generated images closest to the mean similarity across all prompts. The rows above and below show images of approximately one average treatment effect (ATE) more or less similar to the target, illustrating the typical performance difference between model conditions. *(B)* Top: average CLIP cosine similarity to the target image by attempt, separately for DALL-E 2 and DALL-E 3 participants. Bottom: the difference between these averages, with the dark blue line indicating the overall ATE and the shaded region showing the 95% confidence interval. *(C)* Average prompt length by attempt (y-axis), with error bars representing 95% confidence intervals. Color shading indicates the average textual similarity between each prompt and the participant's previous prompt, capturing the extent of prompt reuse and refinement over time.

full sample of participants across both model conditions. The middle row for each target shows the image whose cosine similarity to the target is closest to the mean similarity across all participants. The rows above (below) show images that are approximately one average treatment effect (ATE) more (less) similar to the target than the mean. This visualization provides qualitative intuition for the magnitude of the effect we estimate: the typical difference in fidelity between participants using DALL-E 2 and those using DALL-E 3 (Verbatim). Panel B shows that, across the 10 required prompt attempts, participants assigned to DALL-E 3 (Verbatim) produce images that are, on average, 0.0164 higher in cosine similarity to the target (95% CI: [0.0104, 0.0224], $p < 10^{-5}$). This improvement corresponds to roughly 0.19 standard deviations in performance. The gap persists across all attempts; participants using DALL-E 3 start off producing closer matches and maintain that edge through their 10th prompt.

Participants' dynamic prompting behavior also differs substantially between the two models. As shown in panel C of Figure 1, those assigned to DALL-E 3 write prompts that are, on average, 24% longer than those assigned to DALL-E 2, and this gap widens over successive attempts. Moreover, we observe that DALL-E 3 participants are more likely to reuse or refine their previous prompts (indicated by the color scale), which suggests a more exploitative approach once they discover the model's capacity to handle detailed or complex instructions. Analyses of parts of speech confirm

that these extra words likely provide additional descriptive information rather than mere filler: the proportion of nouns and adjectives—the two most descriptively informative parts of speech—is nearly identical across model conditions (48% for DALL-E 3 vs. 49% for DALL-E 2; $p = 0.215$). This suggests that the increase in prompt length reflects the addition of semantically rich content rather than unnecessary verbosity.

4.2 Replay Analysis and Decomposition of Effects

The differences we observe in prompting behavior suggest that users are actively adapting to the capabilities of the model they are assigned. But how much of the overall performance improvement we observe for DALL-E 3 users is due to the model's enhanced technical capacity, and how much is due to users rewriting their prompts in response to that capacity? To answer this question, we turn to the replay analysis described earlier, which allows us to isolate these two effects identified in Equation 5 empirically.

Panel A of Figure 2 presents the results. To estimate the model effect, we compare the performance of prompts originally written by DALL-E 2 participants when evaluated on DALL-E 2 (the model they were written for) versus when evaluated on DALL-E 3 (Verbatim). Because these prompts were written without knowledge of DALL-E 3's capabilities, any improvement reflects the gain from using a more capable model while holding the prompt fixed. We find that performance improves by 0.0084 in cosine similarity when these prompts are evaluated on DALL-E 3 ($p < 10^{-8}$; bootstrapped standard errors clustered at the participant level), which accounts for approximately 51% of the total difference in performance between the DALL-E 2 and DALL-E 3 arms.

To estimate the prompting effect, we then compare the performance of these same DALL-E 2 prompts to the performance of prompts originally written by DALL-E 3 participants, both evaluated on DALL-E 3. Because both sets of prompts are evaluated on the same model, any difference reflects the effect of users adapting their prompts to the model's capabilities. We find that this prompting effect accounts for the remaining 48% of the total improvement, corresponding to an increase of 0.0079 in cosine similarity ($p = 0.024$). The total treatment effect—i.e., the difference between the original DALL-E 2 prompts on DALL-E 2 and the original DALL-E 3 prompts on DALL-E 3—is 0.0164.

Importantly, when we apply prompts written by DALL-E 3 users to DALL-E 2, we observe no performance benefit relative to the original DALL-E 2 prompts ($\Delta = 0.0020$; $p = 0.56$). This asymmetry reinforces the idea that the gains from prompt adaptation depend on the model's capacity to act on that additional information.

Panel B of Figure 2 illustrates these effects using a single target image. The two rows show different prompts submitted for that target, along with the images they generate when evaluated on each model. In the top row, a prompt originally written by a DALL-E 2 participant yields a higher-fidelity image when replayed on DALL-E 3, demonstrating the improvement in output quality that comes from upgrading the model while holding the prompt fixed. In the bottom row, a prompt written by a DALL-E 3 participant produces a noticeably lower-quality image when rendered by

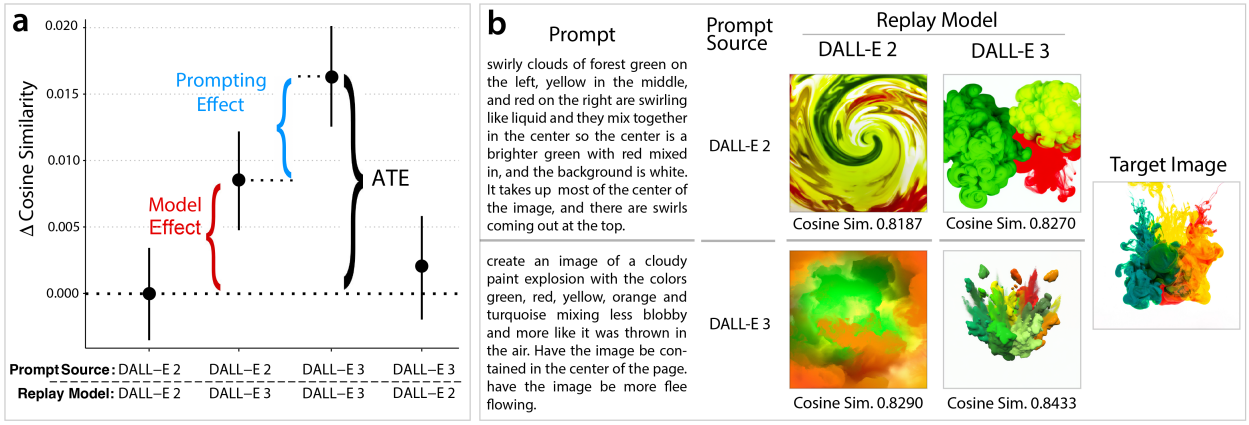


Figure 2: Replay Analysis and Effect Decomposition. *(a)* Average performance of prompts evaluated across four prompt-model combinations. Comparing DALL-E 2 prompts evaluated on DALL-E 2 versus DALL-E 3 isolates the model effect. Comparing reused DALL-E 2 prompts on DALL-E 3 to original DALL-E 3 prompts on DALL-E 3 isolates the prompting effect. Error bars represent 95% confidence intervals based on bootstrapped standard errors clustered at the participant level. *(b)* A single target image with two submitted prompts: one written by a DALL-E 2 participant (top row) and one by a DALL-E 3 participant (bottom row). Images show how each prompt performs on both models, illustrating the model and prompting effects qualitatively.

DALL-E 2, underscoring the limits of prompt adaptation when the model lacks sufficient capacity to execute the instructions effectively.

Taken together, these findings offer empirical support for our theoretical claim: prompt adaptation operates as a dynamic complement that users deploy in response to improved model capabilities—and accounts for a substantial share of realized performance gains.

4.3 Skill Heterogeneity

Table 1 presents the results of a regression analysis that tests whether the model, prompting and total effects vary systematically across participants of differing skill levels. We regress performance on indicators for model, prompting and total effects, as well as their interactions with participants' performance decile (ranging from lowest to highest).

We make three observations. First, the interaction between total effect and performance decile is negative and statistically significant (-0.000115, $p = 0.0152$). This suggests that model improvements reduce the overall gap between high and low-performing users, consistent with proposition 1 in our conceptual framework. Second, the interaction between the model effect and performance decile is also negative and statistically significant (-0.000059, $p = 0.0210$). This suggests that the model effect mainly benefits the lower-performing users—consistent with our theoretical predictions in proposition 2 and due to diminishing returns for those already near the performance ceiling. Third, we do not find any evidence that the benefits of prompt adaptation vary across the skill distribution in a meaningful way since the interaction between the prompting effect with performance decile is not statistically significant (-0.000056, $p = 0.2444$). While this contrasts

Effect	Estimate (SE)	p-value
Total	0.0224 (0.00312)	< 0.00001
Total × Performance Decile	-0.000115 (0.000047)	0.0152
Model	0.0113 (0.00186)	< 0.00001
Model × Performance Decile	-0.000059 (0.000025)	0.0210
Prompting	0.0109 (0.00316)	0.000564
Prompting × Performance Decile	-0.000056 (0.000048)	0.244

Table 1: Regression estimates for the total, model and prompting effects, and their interactions with participant performance decile. The total and model effects are significantly larger for lower-skilled participants, as indicated by the negative and statistically significant interaction term. The prompting effect does not vary significantly across deciles, suggesting that participants across the skill distribution adapt their prompts similarly.

with our theoretical prediction in Proposition 2—which anticipated that higher-skill participants would benefit more from prompt adaptation—it is possible that the effect exists but is modest in size and difficult to detect given the statistical power of our design.

4.4 Prompt Revision

We also evaluate whether an automated prompt adaptation system can substitute for human-driven adaptation. In the DALL-E 3 (Revised) condition, user prompts were silently rewritten by GPT-4 before being submitted to the image model. Although participants in this condition still outperformed those using DALL-E 2 ($\Delta = 0.0069$; $p = 0.042$), they achieved substantially lower performance than participants using DALL-E 3 without revision. On average, automated prompt revision reduced the benefit of DALL-E 3 by 58% (95% CI: [40%, 76%]). Manual inspection of the revised prompts in the DALL-E 3 (Revised) arm confirms that GPT-4 often added extraneous details or subtly altered the intended meaning of participant instructions, leading to degraded performance. In this case, the automatic rewriting process was not well-aligned with the user's goal of replicating a specific target image as faithfully as possible. More broadly, this result highlights a potential risk of using invisible system-level prompt rewriting: when the objectives of the rewriting mechanism are not tightly aligned with user intent, even well-intentioned modifications can interfere with the user's ability to effectively guide the model.

Together, these findings demonstrate that the performance gains observed when users are given access to a more capable generative model stem from both improvements in model architecture and changes in user behavior. Participants assigned to DALL-E 3 not only achieve higher performance but also write longer, more refined prompts over time. Our replay analysis shows that these adapted prompts account for nearly half of the overall performance improvement, with the remainder attributable to the model's enhanced rendering capabilities. This pattern holds across the skill distribution: while lower-skilled users benefit more from model upgrades, prompt adapta-

tion contributes consistently across all levels of baseline performance. Finally, automated prompt rewriting—when misaligned with user intent—fails to replicate the benefits of direct user adaptation. These results highlight the central role that prompting behavior plays in shaping how users interact with and benefit from improved generative AI systems.

5 Discussion and Conclusion

Our findings contribute to the literature on human–AI collaboration and the economics of technology by highlighting the role of prompt adaptation as a dynamic complement that co-evolves with model improvements (Böhm and Schedlberger 2023). As generative AI models advance—often substantially from month to month—organizations that fail to adapt their prompting strategies may forgo a meaningful share of the economic value these upgrades make possible.

This co-evolutionary pattern is not unique to AI. It echoes dynamics observed in earlier general-purpose technologies, where technical improvements often yielded modest returns until complementary skills and practices evolved to match (Brynjolfsson 1993, David 1990). However, the pace of generative AI advancement introduces a distinctive challenge: the window for adaptation is far shorter. Organizations that treat prompting as a one-time investment rather than an ongoing capability risk failing to capture the full value of model upgrades—echoing the productivity paradox observed when complementary assets lag behind technological potential (Brynjolfsson and Hitt 2000, Brynjolfsson et al. 2021). Our findings complement this stream of work by offering direct empirical evidence that a specific dynamic complement—user adaptation, specifically through prompt refinement—accounts for roughly half of the realized performance gains associated with a model upgrade.

This finding extends research on IT-enabled dynamic capabilities (Bharadwaj 2000, Joshi et al. 2010, Teece et al. 1997) and post-adoptive IT use, which emphasizes how value emerges through user experimentation and learning over time (Jasperson et al. 2005). Prompt adaptation, as we observe it, is not grounded in formal training or specialized skill. Participants in our study—none of whom were expert prompt engineers—improved performance through trial-and-error within a single session. This contrasts with earlier IT transitions, where complementary capabilities often required extensive training (Attewell 1992, Von Hippel 2006). The accessibility of prompt adaptation suggests a path toward more broadly distributed productivity gains—provided that users are supported by scaffolds and interfaces that enable iterative refinement (Rogers 2003).

At the same time, this accessibility introduces risk. Over-optimizing for a particular model version may reduce users' ability to adapt as systems evolve. This behavioral lock-in resembles challenges in architectural innovation, where tightly coupled routines inhibit flexibility when key system components change (Henderson and Clark 1990). Supporting long-term adaptation may require not just prompt training, but workflows and learning mechanisms that encourage ongoing experimentation.

Finally, our results caution against automated prompting systems that intervene without align-

ing with user intent. In our experiment, DALL-E 3's automatic prompt rewriting—designed to enhance usability—reduced realized performance gains by 58%. This does not imply that automated prompting is inherently counterproductive. In some settings, automation may simplify use or improve consistency. But when the goals of a rewriting system diverge from those of the user—as in our task, where general improvements (e.g., aesthetic quality or detail) were prioritized over faithful image replication—such interventions can interfere with the adaptive strategies users are developing. Similar challenges have been observed in other domains: for instance, in clinical documentation, automated prompt optimization improved consistency but still required manual revision to ensure quality and alignment (Yao et al. 2024). These findings reinforce the broader insight—rooted in both human–AI collaboration and dynamic capabilities research—that adaptive user behavior is not merely a response to system performance but a critical input. Interventions that disrupt this adaptive process risk eroding the very dynamic complements that make advanced systems productive (Fügener et al. 2022).

While this study offers new insight into prompt adaptation, it has several limitations. First, our focus was restricted to a single transition (from DALL-E 2 to DALL-E 3) and one type of generative AI (text-to-image generation). Although our conceptual framework suggests these mechanisms may generalize to other domains, further research is needed to assess how these dynamics play out in text generation, programming, scientific research, and other high-stakes settings. Second, we observe short-run adaptation behavior in a controlled setting, whereas longer-term learning dynamics, organizational feedback structures, or team-based workflows may shape prompting strategies differently in real-world environments. Third, while our replay analysis isolates the effect of prompt adaptation, we do not identify which specific types of prompt modifications (e.g., lengthening, lexical substitution, structural rephrasing) have a causal effect on performance. Future work should investigate these finer-grained mechanisms to better inform prompting practices and interface design.

Building on these limitations, several promising research directions emerge. One direction is to examine prompt adaptation longitudinally—tracking how users develop durable heuristics that generalize across domains and respond to shifting model behavior over time. Another line of inquiry involves organizational-level complements to prompting, such as shared prompt repositories, collaborative refinement practices, or analytics dashboards that surface effective patterns. A third area concerns interface design—specifically, how features like auto-complete, prompt scoring, or real-time feedback influence users' ability to experiment and adapt. Finally, researchers might explore how organizations balance standardization (which streamlines processes) with adaptability in prompting workflows, in order to avoid creating rigid routines that lag behind technical advances.

For researchers and practitioners interested in the economics of AI, our findings reinforce the idea that complements play a central role in shaping performance. The near-equal contributions of model improvements and prompt adaptation suggest that skill development and technological evolution must be treated as interdependent elements of innovation trajectories (Arthur 2009, Dosi 1982). As generative AI continues to advance, organizations that invest in *adaptive*, not static,

complementary skills will be better positioned to realize its full value.

References

Arthur WB (2009) *The nature of technology: What it is and how it evolves* (Simon and Schuster).

Attewell P (1992) Technology diffusion and organizational learning: The case of business computing. *Organization science* 3(1):1–19.

Bharadwaj A (2000) A resource-based perspective on information technology capability and firm performance: An empirical investigation. *MIS Quarterly* 24(1):169–196.

Bick A, Blandin A, Deming DJ (2024) The rapid adoption of generative ai. Technical report, National Bureau of Economic Research.

Böhm K, Schedlberger L (2023) The use of generative ai in the domain of human creations – a case for co-evolution? *Proceedings of the 9th International Conference on Socio-Technical Perspectives in IS (STPIS'23)* (Portsmouth, UK: CEUR Workshop Proceedings), URL `http://ceur-ws.org/Vol-3535/`, available under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Boiko D, MacKnight R, Kline B, et al. (2023) Autonomous chemical research with large language models. *Nature* 624:570–578, URL `http://dx.doi.org/10.1038/s41586-023-06792-0`.

Bright J, Enock FE, Esnaashari S, Francis J, Hashem Y, Morgan D (2024) Generative ai is already widespread in the public sector. *arXiv preprint arXiv:2401.01291* .

Brynjolfsson E (1993) The productivity paradox of information technology. *Communications of the ACM* 36(12):66–77.

Brynjolfsson E, Hitt LM (2000) Beyond computation: Information technology, organizational transformation and business performance. *Journal of Economic perspectives* 14(4):23–48.

Brynjolfsson E, Li D, Raymond LR (2023) Generative ai at work. Technical report, National Bureau of Economic Research.

Brynjolfsson E, Rock D, Syverson C (2021) The productivity j-curve: How intangibles complement general purpose technologies. *American Economic Journal: Macroeconomics* 13(1):333–372.

Chen L, Zaharia M, Zou J (2023) How is chatgpt's behavior changing over time? URL `https://arxiv.org/abs/2307.09009`.

David PA (1990) The dynamo and the computer: an historical perspective on the modern productivity paradox. *The American economic review* 80(2):355–361.

Dell'Acqua F, McFowland E, Mollick ER, Lifshitz-Assaf H, Kellogg K, Rajendran S, Krayer L, Candelon F, Lakhani KR (2023) Navigating the jagged technological frontier: Field experimental evidence of the effects of ai on knowledge worker productivity and quality. Technical Report 24-013, Harvard Business School Technology & Operations Management Unit, working Paper.

Don-Yehiya S, Choshen L, Abend O (2023) Human learning by model feedback: The dynamics of iterative prompting with midjourney. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 4146–4161, EMNLP '23, URL `http://dx.doi.org/10.18653/v1/2023.emnlp-main.253`.

Dosi G (1982) Technological paradigms and technological trajectories: a suggested interpretation of the determinants and directions of technical change. *Research policy* 11(3):147–162.

Fu S, Tamir N, Sundaram S, Chai L, Zhang R, Dekel T, Isola P (2023) Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344* .

Fügener A, Grahl J, Gupta A, Ketter W (2022) Cognitive challenges in human–artificial intelligence collaboration: Investigating the path toward productive delegation. *Information Systems Research* 33(2):678–696.

Henderson RM, Clark KB (1990) Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms. *Administrative science quarterly* 9–30.

Jasperson J, Carter PE, Zmud RW (2005) A comprehensive conceptualization of post-adoptive behaviors associated with it-enabled work systems. *MIS Quarterly* 29(3):525–557.

Joshi KD, Chi L, Datta A, Han S (2010) Changing the competitive landscape: Continuous innovation through it-enabled knowledge capabilities. *Information Systems Research* 21(3):472–495.

Liang JT, Lin M, Rao N, Myers BA (2024) Prompts are programs too! understanding how developers build software containing prompts. *arXiv preprint arXiv:2409.12447* .

Manning BS, Zhu K, Horton JJ (2024) Automated social science: Language models as scientist and subjects. Technical report, NBER, accessed: 2024-03-12.

Meincke L, Mollick E, Mollick L, Shapiro D (2025) Prompting science report 1: Prompt engineering is complicated and contingent. *arXiv preprint arXiv:2503.04818* .

Neelakantan A, Xu T, Puri R, Radford A, Han JM, Tworek J, Yuan Q, Tezak N, Kim JW, Hallacy C, Heidecke J, Shyam P, Power B, Nekoul TE, Sastry G, Krueger G, Schnurr D, Such FP, Hsu K, Thompson M, Khan T, Sherbakov T, Jang J, Welinder P, Weng L (2022) Text and code embeddings by contrastive pre-training.

Noy S, Zhang W (2023) Experimental evidence on the productivity effects of generative artificial intelligence. *Science* 381(6654):187–192, URL `http://dx.doi.org/10.1126/science.adh2586`.

OpenAI (2024) CLIP: Contrastive Language–Image Pretraining. `https://huggingface.co/docs/transformers/en/model_doc/clip`, accessed: 2024-01-30.

Oppenlaender J (2023) A taxonomy of prompt modifiers for text-to-image generation. *Behaviour & Information Technology* 1–14, URL `http://dx.doi.org/10.1080/0144929X.2023.2286532`.

Peng S, Kalliamvakou E, Cihon P, Demirer M (2023) The impact of ai on developer productivity: Evidence from github copilot. URL `https://arxiv.org/abs/2302.06590`.

Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, et al. (2021) Learning transferable visual models from natural language supervision. *International conference on machine learning*, 8748–8763 (PMLR).

Rogers EM (2003) *Diffusion of Innovations* (New York: Free Press), 5th edition.

Romera-Paredes B, Barekatain M, Novikov A, et al. (2024) Mathematical discoveries from program search with large language models. *Nature* 625:468–475, URL `http://dx.doi.org/10.1038/s41586-023-06924-6`.

Sävje F, Higgins MJ, Sekhon JS (2021) Generalized full matching. *Political Analysis* 29(4):423–447.

Schulhoff S, Ilie M, Balepur N, Kahadze K, Liu A, Si C, Li Y, Gupta A, Han H, Schulhoff S, et al. (2024) The prompt report: A systematic survey of prompting techniques. *arXiv preprint arXiv:2406.06608* .

Singla N, Garg D (2012) String matching algorithms and their applicability in various applications. *International journal of soft computing and engineering* 1(6):218–222.

Teece DJ, Pisano G, Shuen A (1997) Dynamic capabilities and strategic management. *Strategic Management Journal* 18(7):509–533.

Toner-Rodgers A (2024) Artificial intelligence, scientific discovery, and product innovation. URL `https://arxiv.org/abs/2412.17866`.

Torricelli M, Martino M, Baronchelli A, Aiello LM (2023) The role of interface design on prompt-mediated creativity in generative ai. *arXiv preprint arXiv:2312.00233* .

Universal Dependencies Project (2024) Universal POS tags. `https://universaldependencies.org/u/pos/`, accessed: 2024-07-05.

Von Hippel E (2006) *Democratizing innovation* (the MIT Press).

Xie Y, Pan Z, Ma J, Jie L, Mei Q (2023) A prompt log analysis of text-to-image generation systems. *Proceedings of the ACM Web Conference 2023*, 3892–3902, WWW '23, ISBN 9781450394161, URL `http://dx.doi.org/10.1145/3543507.3587430`.

Yao Z, Jaafar A, Wang B, Yang Z, Yu H (2024) Do clinicians know how to prompt? the need for automatic prompt optimization help in clinical note generation. *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, 182–201.

Yu Z (2024) The impacts of ai on scientific labor: Evidence from protein structure prediction. *SSRN Electronic Journal* URL `http://dx.doi.org/10.2139/ssrn.4711334`, available at SSRN: `https://ssrn.com/abstract=4711334` or `http://dx.doi.org/10.2139/ssrn.4711334`.

Zhang P, Kamel Boulos MN (2023) Generative ai in medicine and healthcare: promises, opportunities and challenges. *Future Internet* 15(9):286.

A Experiment Design

A.1 Task Design

Participants were asked to reproduce a single target image as closely as possible using a text-to-image generative AI model (i.e., DALL-E 2, DALL-E 3 without prompt revision, or DALL-E 3 with prompt revision, all developed by OpenAI). They did so by successively submitting prompts. In response to each submitted prompt, the model would generate an image, which was then displayed to the participant next to their assigned target image. Participants were instructed to make at least 10 attempts at trying to recreate the target image within a 25-minute window, with no upper limit on their number of attempts.

All interactions between participants and the generative AI models occurred on a custom-built online interface designed to resemble OpenAI's ChatGPT interface but with some adjustments related to our task (e.g., displaying the target image and the total number of attempts so far to the user). On the right-hand side of the interface, participants were shown the target image they were randomly assigned to recreate. On the left-hand side, participants were shown their previously submitted prompts as well as the resulting generated images. We placed the text box where participants were able to write and submit their prompts at the bottom of the interface. Prompts were limited to a maximum of 1,000 characters. Participants were informed that their interactions with their assigned model would be memory-less, i.e., the model retained no memory of previous prompts and only used the current prompt to generate each image. Before the task, participants were provided with written and video instructions on how to interact with our experiment interface. Our task did not assume nor require prior experience with *any* generative AI tools.

After the task, we surveyed participants' opinions and preferences regarding generative AI tools. We also inquired about their self-assessed occupational skills and how often they 1) engaged in creative writing, 2) wrote specific instructions, and 3) engaged in any sort of computer programming. Finally, we collected socio-demographic data, such as age, gender, and occupation.

A.2 Randomization

We randomized participants across two dimensions: the target image and the text-to-image generative AI model that participants had access to. We randomized participants across both dimensions simultaneously using complete randomization, generating 45 possible target image-model cells. We conducted a balance check after the conclusion of the experiment with a χ^2 test across all cells. With $\chi^2 = 7.056$, $df = 44$, the resulting p-value equals to 1 and thus we cannot reject the null hypothesis that the proportions are equal across all 45 groups:

$$H_0 : p_1 = p_2 = \cdots = p_{45}$$

Participants were unaware of this randomization.

A.2.1 Generative Models

We randomly assigned participants to 1 of 3 generative models:

1. DALL-E 2, which is referred to at points in the main text as the inferior model.
2. DALL-E 3 (Verbatim), which is referred to at points in the main text as the superior model.
3. DALL-E 3 (Revised), which is referred to at points in the main text as DALL-E 3 with Revision

Both the "verbatim" and "revised" versions of the DALL-E 3 treatment utilize the same underlying image-generating model; the distinction lies in the pre-processing applied before submitting user prompts to OpenAI's image-generating API. OpenAI's DALL-E 3 system, by design, employs a GPT-4 model to rewrite user prompts, adding more detail before processing the modified prompt using the DALL-E image-generating model. During our experiment, it was not possible to explicitly disable this prompt rewriting feature of the DALL-E 3 system. To manage this behavior, we defined two treatments utilizing the DALL-E 3 model.

In the DALL-E 3 (Revised) treatment arm, we submit the participant's prompt directly to OpenAI's API and do not interfere with the default prompt rewriting process. In the DALL-E 3 (Verbatim) treatment arm, we prepend a string instructing the GPT-4 model to not modify the participant's prompt before passing it forward to DALL-E 3. This string is never visible to participants and was modeled after a prefix specifically suggested in OpenAI's online documentation for the DALL-E 3 endpoint.[8] We modified the recommended prefix slightly to account for the fact that we did not expect our participants to always submit "extremely simple prompts." The string we prepended to prompts is found below:

> "I NEED to test how the tool works with my prompt as it is written. DO NOT add any detail; just use it AS IS:"

Prepending this string to participants' prompt did reduce the rate at which OpenAI's endpoint modified prompts, but compliance was not perfect. Thus, we view the "verbatim" treatment arm as more of an intent-to-treat intervention. The GPT model still modified 59% of participant prompts. The average token sort ratio (TSR) between the original prompt and the modified prompt was 77 for the DALL-E 3 (Verbatim) arm, compared to an average token sort ratio of 44 across the entire DALL-E 3 (Revised) treatment arm (a TSR of 100 denotes an exact string match). Conditional on any modification (any observations with TSR < 100), the average TSR between the original prompt and the modified prompt was 61 for the DALL-E 3 (Verbatim) arm, compared to an average TSR of 44 across the entire DALL-E 3 (Revised) treatment arm.

[8] See here and here for online documentation.

A.2.2 Target Images

We randomly assigned participants to 1 of 15 target images. The set of target images consisted of 5 images each from 3 different broad categories: business and marketing, graphic design, and architectural photography. We chose these to represent the use cases suggested by the prompt categories on https://promptbase.com/, a leading marketplace for image generation prompts. The images vary in color, style, content, and complexity within and across categories. These images can be found online linked to the pre-registration document: **URL removed to maintain anonymity**. Performance, and variability of performance varied substantially across images. In other words, some images were much easier than others to replicate with the generative models, which we view as additional evidence that the set of 15 images was reasonably diverse.

A.3 Subjects

Our Prolific-recruited US sample (N = 2,059) was limited to fluent English speakers, and we prevented participants from completing the task more than once. We also prevented users from completing the task on mobile devices or tablets. Data was collected between December 12, 2023 and December 19, 2023. Participants were guaranteed a payment of $4 USD for completing the task and could earn an additional $8 USD (a 200% bonus) if they ranked in the top 20% of participants in DreamSim of their image most similar to the target (construction of DreamSim is described in section C.5.1). The median time to complete our entire task, including a demographic survey, was 22 minutes. Given that 20% of subjects received a bonus, the average compensation for participants in our study was $5.60 USD per person, or about $15 USD per hour. We explained the payment and incentive scheme to participants in full multiple times during the onboarding phase of the experiment, and asked participants to confirm their understanding before they were allowed to complete the task. The onboarding process also included multiple attention checks; participants who failed the first check were immediately disqualified. For subsequent checks, participants were required to retry until they demonstrated understanding.

A.4 Model Endpoints

We used the following model endpoints and parameters to generate images from prompts:

1. **OpenAI API:** We used the image generation endpoint of the official OpenAI Node.js library to generate images for user prompts during the experiment. For all treatment arms, we set the image size parameter to be 1024 x 1024 pixels. For the DALL-E 3 (Revised) and DALL-E 3 (Verbatim) treatment arms, we set the quality parameter to standard and the style parameter to natural.

2. **Azure OpenAI Service:** We used the image generation endpoints in the Python implementation of Azure OpenAI Service to generate all replay images based off user prompts collected

during the experiment. For prompts replayed through the DALL-E 2 treatment arm, we deployed a set of DALL-E 2 models on Azure OpenAI Service and set the API version for each to the `2023-06-01-preview` version. For DALL-E 2, we created replay images in batches of 5. For prompts replayed through the DALL-E 3 (Revised) and DALL-E 3 (Verbatim) treatment arms, we deployed a set of DALL-E 3 models on Azure OpenAI Service and set the API version for each to the `2023-12-01-preview` version. The parameter values for image size, and quality and style for the DALL-E 3 treatment arms, were set to the same values as in the experiment.

A.5 Image Similarity Metrics

The primary outcome in our experiment is the similarity between each participant-generated image and the assigned target image, measured using the cosine similarity of CLIP embeddings (Radford et al. 2021). CLIP (Contrastive Language–Image Pretraining) is a neural network trained to jointly embed images and text into a shared latent space, such that semantically or visually similar items lie close together. By embedding both the target image and each generated image into this space and computing the cosine similarity between them, we obtain a quantitative measure of how closely the generated image matches the target along both visual and conceptual dimensions.

Because the output of each generative model is stochastic, the same prompt can yield different images across attempts. To account for this variability, we generated 10 images for each prompt and computed their cosine similarity to the target image individually. We then averaged these 10 similarity scores to produce an expected quality score for each prompt—the primary outcome variable used in our analysis. As a robustness check, we replicated all analyses using DreamSim, a recently developed perceptual similarity metric that aligns more closely with human judgments.[9] The two measures were highly correlated, and our findings were consistent across both.

A.6 Pre-registration

This study was pre-registered. The pre-registration document included our hypotheses, planned analyses, and sample size justification. The pre-registration document can be found at **URL removed to maintain anonymity**.

B Example Prompts

Table 2 presents the complete set of prompts and their corresponding generated images that were used in our analysis shown in Figure 1. The table displays three example target images (woman with shopping basket, abstract painting with cross, and online shopping interface), with three different participant-generated versions of each. For each target, we show examples of varying

[9] DreamSim (Fu et al. 2023) is designed to better capture human perceptions of image similarity than traditional embedding-based methods. Our results are robust to using DreamSim in place of CLIP cosine similarity. Full DreamSim-based analyses can be found in Appendix F.

quality, arranged from most similar to least similar. The prompts are presented verbatim from participant submissions, preserving all original formatting and punctuation (or lack thereof).

C Measurement and Variables

This appendix provides detailed information about the data collection, measurement, and variables used in our analyses, expanding on the overview provided in the main text.

C.1 Survey Data

We collected additional participant information via a Qualtrics survey that included:

- **Demographics:** Ethnicity, Gender, Age, Highest level of education attained (some high school, high school, some college, associate's degree, bachelor's degree, master's degree, doctoral degree, professional degree, other), Years of work experience, Annual Income (0-$25k, $25.001k-$50k, $50.001k-75k, $75.001k-$100k, $100.001k-$150k, $150k+), and elicitation of sets of O*NET job skills that participants used in their occupation (reading comprehension, active listening, writing, speaking, critical thinking, social perceptiveness, coordination, instructing, programming, judgment and decision making, systems evaluations, science, active learning, learning strategies, monitoring, complex problem analysis, technology design, troubleshooting, quality control analysis, systems analysis).
- **Opinions and Skills:** Computer programming proficiency and usage frequency (self-reported), Structured and creative writing proficiency and usage frequency (self-reported), Generative AI tool proficiency and usage frequency (self-reported), Attitudes towards net social impact of Generative AI (self-reported), Advice for (hypothetical) future participants on how to perform well on the task.

C.2 Prompt Data

For each prompt, we recorded the text of the participant's prompt, the order in which it was submitted, the timestamp of submission, and for the DALL-E 3 treatment arms, the revised prompt returned by the model.

C.3 Image Data

We collected three different sets of images:

1. **The participant-facing images (OpenAI API endpoint):** The image shown to the participant during the experiment, generated by the model they were assigned to using the prompt they submitted. These images were generated from December 12-19, 2023.

Table 2: Participant prompts for the images provided in Figure 1

Image	Prompt
	"create an image of a woman with blond hair, wearing a surgical mask, black gloves, a white t-shirt, and holding a red shopping basket full of groceries"
	"A lady has long blonde hair, wears BLACK COLORED GLOVES and a BLUE COLORED mask, with a white blouse on. She is holding a RED PLASTIC BASKET filled with different kinds of fruit."
	"Generate a stock photo of a woman holding a basket of groceries during the pandemic. Make sure she has on black gloves."
	"Thank you! Please create a mostly white painting with red and blue lines in the center, shaped a bit like the # symbol. Add red and blue splatter as well, but not too much"
	"Water color style. Japanese vibes. White background. Square outline in the middle of the page is standing on its side at a 12 degree angle. The inside of the square is white. Two legs are blue and two are red. T"
	"painting of two x's intersecting to make a slanted square. The square is at a 35 degree angle with the left upper side being two red lines making the x and the lower right side two blue lines. The painting only has red and blue and the square with 2 x's sits in the middle of the painting with light speckles of red and blue drops of paint scattered throughout the painting"
	"pastels, abstract, bearded man on left in a pink shirt with blue sleeves pointing to a shirt box image, woman with pink hair in a blue dress and high heels holding the handle of a shopping cart full of purses on right side, pink and white awning in center top, 4 out of 5 stars on left side in pink, animation style. single pink dollar sign on the left. white search bar in the center with a magnifying glass icon on the right side of the search bar."
	"cartoon-y virtual self checkout. man shopping for clothes. woman pushing cart. phone that looks like a storefront in background."
	"at the store a man using a touch screen monitor to shop for a shirt and a woman walking by with a store cart full of gift"

2. **Post-hoc resampled images (Azure OpenAI endpoint):** For any given prompt, the output of the text-to-image model is stochastic. To better approximate the expected image from a given prompt, we generated 20 additional images for each prompt after the experiment concluded. We provide full details on this procedure in Section D. These images were generated from December 26, 2023 - January 27, 2024. These images are not used for analyses presented in the main text, but were used for other pre-registered analyses. These additional analyses are discussed in Section G.

3. **Post-hoc replayed images (Azure OpenAI endpoint):** To decompose our overall effects into model and prompting effects, we generated "counterfactual images" for each prompt written under the DALL-E 2 and DALL-E 3 (Verbatim) treatments. In other words, we submitted all prompts written under both the DALL-E 2 and DALL-E 3 (Verbatim) treatments to both the DALL-E 2 and DALL-E 3 (Verbatim) endpoints. Similarly to the resampling procedure outlined above, we generated 10 images per prompt per model: we generated a single replay for each prompt-model pair from March 16-18, 2024, and then, to increase power, generated the replications for these replay images from June 14-27, 2024. This replay process produced a total of 20 images per prompt—10 under the original model, 10 under the counterfactual model. We re-submitted prompts to their original model to account for potential model drift, as this exploratory analysis was conducted multiple months after our initial data collection. For consistency, this replay data is used throughout the main text of our paper.

C.4 Sample Construction

The sample analyzed in the main text was constructed as follows:

- The initial "raw" dataset collected during the experiment is comprised of 24,672 rows of raw prompt data (one prompt per row) generated by 2,059 participants.

- We first removed rows with blank prompt entries, invalid prolific IDs, and unsuccessful attempts (logging errors). These exclusion criteria were pre-registered. This left us with 2,029 participants and 24,123 prompts.

- We next removed participants from our sample if they failed to submit at least 10 prompts or if a participant submitted the same prompt at least five times in a row at any point during the task. Both of these exclusion criteria were pre-registered. These exclusion criteria were also explained to participants, who were told that payment was contingent on submitting at least 10 successful prompts and a "good-faith effort." To avoid reward hacking, we did not specify the "no more than 5 repeated prompts" criterion for "good-faith effort." This left us with 1,899 participants.

- Although participants were allowed to submit as many prompts as they desired in the 25-minute time span, we limited all analyses to each participant's first 10 prompts—the minimum

required to receive payment for the task. This exclusion criteria was not pre-registered, and is noted in the list of deviations from pre-registration in Section G.G.2. We restrict our analysis dataset in this way because participants who chose to submit more than 10 prompts may have been systematically different than those who did not. Excluding any prompt beyond the 10^{th} attempt allows us to alleviate selection bias concerns. This left us with 18,990 prompt observations from 1,899 participants.

- We next removed participants who failed to complete the Qualtrics survey. This exclusion criteria was pre-registered. This left us with 1,893 participants and 18,930 prompts.
- We also removed prompts from our dataset according to a number of post-hoc, non-pre-registered exclusion criteria to ensure data quality and avoid selection bias. If a prompt had any of the following flags, it was removed from the sample:
  - Prompts sometimes trigger errors in OpenAI's safety system because they contain language that might be deemed unsafe under OpenAI's policies. The specific language that triggers these errors is constantly changing and not available publicly. If a prompt triggered a safety error during the replication or replay process, we re-submitted the prompt up to 50 times or until the 10 original arm replications/replay samples had been collected. We removed prompts if they failed to generate 10 replications on the original model or 10 replay samples under the counterfactual model during the replication/replay process. This affected 305 prompts between the DALL-E 2 and DALL-E 3 (Verbatim) treatment arms. It did not affect any DALL-E 3 (Revised) prompts, as we did not conduct replay analysis with the prompts from this treatment arm.
  - Due to rare latency issues, some prompts were assigned duplicate attempt numbers by the MongoDB database that we used to collect our data. This data collection error led to issues in the data analysis process. Thus, we excluded prompts with duplicate attempt numbers. This affected 34 prompts across all three treatment arms, and 20 prompts between the DALL-E 2 and DALL-E 3 (Verbatim) treatment arms, approximately 0.1% of the original data.
- Our final sample included 1,893 participants and 18,560 prompts.

C.4.1 "Off-Topic" Robustness Check

While analyzing our data, we found that our sample contained a number of "off-topic" prompts that did not seem related to the task. As a robustness check on our main results, we used the following process to systematically identify and remove "off-topic" prompts. First, we generated embeddings for each prompt using OpenAI's `text-embedding-3-small` model. We then calculated the mean embedding for each target image. Next, we calculated the Euclidean distance between each prompt's embedding vector and the mean embedding vector for prompts corresponding to the focal prompt's assigned target image. Finally, we removed the 2.5% of prompts that were

most distant from the mean image-level prompt embedding vector. This led to the removal of 481 prompts across all three treatment arms, and 338 prompts between the DALL-E 2 and DALL-E 3 (Verbatim) treatment arms. All of our main text results are robust to the exclusion of these "off-topic" prompts.

C.5 Dependent Variables

C.5.1 Image Similarity

We pre-registered two quantitative measures of image similarity: the cosine similarity of CLIP embedding vectors and a recently developed measure called 'DreamSim' (Fu et al. 2023). In the main text, we present analyses using CLIP embedding cosine similarity, since it is likely more familiar to readers. Our results are qualitatively and quantitatively similar using DreamSim instead.

- **CLIP Embedding Cosine Similarity:** To calculate CLIP embedding cosine similarity, we first generated CLIP embedding vectors (Radford et al. 2021) from Hugging Face (OpenAI 2024) for each participant-generated image and for each target image. Unlike traditional image embeddings that only encode visual features, CLIP embeddings also capture semantic relationships between images and descriptive text. We then calculated the cosine similarity between each participant-generated image's CLIP embedding and the relevant target image's CLIP embedding.

- **DreamSim:** DreamSim is an image similarity measure proposed recently by (Fu et al. 2023). The authors claim that relative to a measure such as CLIP embedding cosine similarity, DreamSim measures image similarity in a way that more effectively captures human visual perceptions of similarity. Because the original DreamSim metric outputs a distance measure, we invert this score $\tilde{D} = 1 - (\text{original DreamSim})$ to recast it as a similarity score. After doing so, both the inverted DreamSim and CLIP embedding cosine similarity are closer to 1 when two images are more similar and closer to 0 when two images are more dissimilar.

We find that these two measures of image similarity are highly correlated in our sample ($\rho_{pearson} = 0.763$, 95% CI: [0.755 0.770]), and our main results are robust to the use of either measure. We present the results obtained when conducting our main text analyses using DreamSim in Section F.F.1.

C.5.2 Prompt Length

We measure the lengths of prompts written by participants in our sample, both in terms of the number of *words* in a given prompt and in terms of the number of *characters* in a given prompt. In our main text analysis, we present results only in terms of the number of words, since the two outcomes are highly correlated ($\rho_{pearson} = 0.9954$, 95% CI: [0.99528, 0.99560]).

C.5.3 Embedding-based Prompt Similarity

We calculate two measures of embedding-based prompt similarity: successive similarity and aggregate similarity. Both measures use the vector embedding representation of each prompt in our sample, which we obtained using OpenAI's `text-embedding-3-small` model (Neelakantan et al. 2022). The two similarity measures are defined as follows:

- **Successive similarity:** The successive similarity (ss) is a measure of the similarity of a participant's prompt to their immediately preceding prompt. We define the successive similarity of a prompt $p_{i,n}$ written by user i to the their immediately preceding prompt $p_{i,n-1}$ as:

$$ss_{i,n,n-1} = \frac{\mathbf{E(p_{i,n})} \cdot \mathbf{E(p_{i,n-1})}}{||\mathbf{E(p_{i,n})}||\,||\mathbf{E(p_{i,n-1})}||}, \tag{6}$$

  where $\mathbf{E(p_{i,n})}$ is the vector embedding representation of participant i's n^{th} prompt, $p_{i,n}$. This measure starts with participant i's 2^{nd} attempt, as the calculation requires a previous attempt.

- **Aggregate similarity:** The aggregate similarity (as) is a measure of how dispersed each user's prompts are around their "average prompt" (calculated by taking the element-wise average of all prompt embeddings produced by the user). We define the aggregate similarity for the 10 prompts written by a given user as:

$$as_i = \frac{1}{10}\sum_{n=1}^{10} \|\mathbf{E(p_{i,n})} - \overline{\mathbf{E(p_{i,n})}}\|_2^2, \tag{7}$$

  where $\mathbf{E(p_{i,n})}$ is again the vector embedding representation of participant i's n^{th} prompt, $p_{i,n}$, and $\overline{\mathbf{E(p_{i,n})}}$ is the element-wise mean of all 10 of participant i's prompts.

C.5.4 Successive Prompt Token Sort Ratio

Starting with each participant's second prompt, we also calculated the token sort ratio (TSR) of each prompt $p_{i,n}$ to the immediately preceding prompt $p_{i,n-1}$. TSR is a fuzzy string-matching technique (Singla and Garg 2012) that provides a continuous measure of how similar two strings are.

C.5.5 Successive Prompt 'Contains Previous Prompt' Dummy

Starting with each participant's second prompt, we record whether each prompt $p_{i,n}$ contains the immediately preceding prompt $p_{i,n-1}$ as an exact substring.

C.5.6 Prompt Composition

We use the `spaCy v3.7.4` Python package's `en_core_web_sm` model to tag the parts of speech (POS) in each prompt. SpaCy's models utilize the "universal POS tags" from the Universal Dependencies framework for grammar annotation Universal Dependencies Project (2024). These tags encompass parts of speech such as adjectives, adverbs, nouns, and verbs. The model tags each word in a prompt according to this framework, after which we count the total number of words corresponding to each part of speech for each prompt.

C.5.7 Strategic Shifts

In addition to calculating the successive and aggregate similarity of prompts written by particular users, we also attempt to identify particular moments when participants shift their approach to prompting. In order to do so, we adapt a method proposed in (Torricelli et al. 2023) (because they are conducting research in a different context, (Torricelli et al. 2023) refer to these shifts as "topical transitions" as opposed to "strategic shifts"). To identify these strategic shifts, we first calculate the mean cosine similarity (MCS) for the embedding vectors of every possible pair of prompts submitted in response to a given target image, t:

$$MCS_t = \frac{2}{P_t(P_t-1)} \sum_{a=1}^{P_t} \sum_{b=a+1}^{P_t-1} \text{CosineSim}(\mathbf{E}(\mathbf{p_{a,t}}), \mathbf{E}(\mathbf{p_{b,t}})). \tag{8}$$

where P_t is the total number of prompts submitted in response to a given target image, and a and b are indices representing individual prompts for that target.

We then label any given prompt as a strategic shift (SS) if the cosine similarity of its embedding vector with that of the previous prompt is lower than this target-image-level mean:

$$SS(\mathbf{p}_{i,n,t}) = \begin{cases} 1 & \text{if CosineSim}\big(\mathbf{E}(\mathbf{p}_{i,n,t}), \mathbf{E}(\mathbf{p}_{i-1,n,t})\big) \\ & \quad < MCS_t \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

It is worth noting that Torricelli et al. (2023) uses the participant-level mean, as opposed to the task-level mean, as the cutoff for a topical shift. We instead use the task-level mean because in our setting, as it did not seem appropriate that half of each participant's submitted prompts would be strategic shifts.

D Methods

This appendix provides additional methodological details to supplement the analyses presented in the main text.

D.1 Stratification

The results shown in the main text and in the supplementary information are typically stratified by reference image and iteration. In some analyses, we have stratified only on the reference image (e.g., for analyses presented at the iteration level). The exact stratification for each finding is indicated in section E. To stratify our results, we calculate a weighted average across $j = 1, ..., J$ cells defined by our stratification variables:

$$\overline{Y}_{strat} = \sum_{j=1}^{J} \frac{N_j}{N} \bar{Y}_j$$

To calculate the variance (and standard error) of this sample mean, we apply the following:

$$\widehat{\text{Var}}(\overline{Y}_{strat}) = \widehat{\text{Var}} \left(\sum_{j=1}^{J} \frac{N_j}{N} \bar{Y}_j \right) = \sum_{j=1}^{J} \left(\frac{N_j}{N} \right)^2 \frac{s_j^2}{N_j}$$

where:

- N_j is the population size of stratum j.
- N is the total population size across all strata.
- $\bar{Y}_j$ is the sample mean for stratum j.
- J is the total number of strata.
- s_j is the sample standard deviation of stratum j. Therefore, s_j^2 is sample variance stratum j.

D.2 Z-Scoring

Our analysis found statistically significant differences in performance variability across the 15 target images used in our experiments, as discussed in Section G. The main text also demonstrated that performance increases across successive attempts. To ensure our results are not driven by this image-level or attempt-level variation, we replicated all analyses using the within-image-attempt Z-score of CLIP-cosine similarity for each image produced by participants. Formally, this is:

$$Z(\text{Sim}_{i,n,t}) = \frac{\text{Sim}_{i,n,t} - \text{Mean}_{n,t}\left(\text{Sim}_{i,n,t}\right)}{\text{SD}_{n,t}\left(\text{Sim}_{i,n,t}\right)} \tag{10}$$

where $\text{Sim}_{i,n,t}$ is the cosine similarity of user i's image in attempt n to target image t. The mean and standard deviation are computed for each image-attempt pair, but across both the DALL-E 2 and DALL-E 3 treatment arms. We also applied this rescaling to test the robustness of our DreamSim-based analyses.

Nearly all robustness analyses reported here use the Z-score scaled measure of performance within each image-attempt set. The only exception is analyses that examine improvements and prompts across attempts (e.g., Figure 1B), where Z-scores are computed only within each target image and across all attempts for that image.

D.3 Accounting for Model Stochasticity

Generative AI models produce stochastic outputs in response to a given prompt. This stochasticity is controlled by a parameter called temperature, which could not be modified using the DALL-E API at the time of our experiment. To account for this model stochasticity, we generated 10 images for each prompt submitted by participants across all treatment arms. We then calculated the similarity between each replication and its corresponding target image, and computed an "expected" CLIP cosine similarity and DreamSim score for each prompt by averaging across these samples.

Using these replicated images, we also calculated the standard deviation of cosine similarity induced by this stochasticity. With the expected cosine similarity and its standard deviation per prompt, we computed a normalized Z-score for the observed image relative to its replication distribution. This Z-score measures the extent to which the observed image is better or worse than what's expected for that prompt and is used for further analysis in section G.

We generated these additional samples both for the original prompts on their assigned treatment arms and by replaying prompts on the counterfactual arms, as introduced in Figure 2 in the main text. Importantly, OpenAI updated its content filters between our initial experiment and image re-sampling. As a result, some prompts that originally produced images either generated no images or fewer images than requested during our regeneration attempts. This affected 1.8% (371 out of 18,990 prompts) of the data in our sample under the "replaying" procedure (Section C.C.3.3).

E Main Text Analyses

This section provides detailed methodological information about the analyses presented in the main text.

E.1 Conceptual Framework

This appendix provides detailed derivations and proofs for the conceptual framework presented in the main text. While the main text introduces the conceptual framework and key insights, here we describe the model in more detail and develop its mathematical foundations and results more thoroughly.

We model a task where users attempt to replicate an object (such as an image) using a generative model. Perfect replication yields a maximum quality value of 1. The bounded *quality function* is defined as:

$$Q(\theta, s, x) \;=\; 1 \;-\; \exp\big[-\,\theta\, s\, x\big].$$

where $\theta \in (0, 1]$ denotes the *model capacity*, $s \in (0, 1]$ denotes the *user's skill* at writing prompts, and $x \geq 0$ is the level of *user's effort* in generating the appropriate prompts.

This function satisfies two intuitive properties:

1. *Boundedness:* Even as x becomes large, Q remains below 1, reflecting that the user cannot exceed perfect replication.
2. *Diminishing returns:* As user effort (x), increases, its marginal impact on quality becomes smaller.

We assume a linear *cost function* of prompting:

$$c(x) \;=\; k\,x,$$

where $k > 0$ represents the marginal cost of user effort, for example in terms of time spent. The user's utility given the quality and cost functions is

$$U(\theta, s, x) = 1 - e^{-\theta s x} - k\,x.$$

E.1.1 User Optimization

The user chooses the effort $x \geq 0$ such that it maximizes $U(\theta, s, x)$. We assume the user's solution is always interior with positive effort which requires $\frac{\theta s}{k} > 1$. This can be justified since the time cost associated with the task is minimal ($k \approx 0$). As such, optimal user effort is $x^*(\theta, s)$:

$$x^*(\theta, s) \;=\; \frac{1}{\theta s}\,\ln\!\Big(\frac{\theta s}{k}\Big) > 0$$

Thus, the optimal quality the user achieves at their optimal effort level becomes:

$$Q^* = Q\big(\theta, s, x^*(\theta, s)\big) = 1 - \frac{k}{\theta s} > 0$$

E.1.2 Comparative Statics

We now analyze how quality, Q^*, varies with θ, s, and k. Comparative statics are easily shown by taking the derivative of Q^* with respect to the model capacity, user's skill, and prompting cost.
Model Capacity: As model capacity θ increases, optimal quality Q^* increases.

$$\left.\frac{\partial Q^*}{\partial \Theta}\right|_{\Theta=\theta} = \frac{k}{\theta^2 s} > 0 \tag{11}$$

User's Skill: As the skill of the user s increases, the optimal quality Q^* also increases.

$$\left.\frac{\partial Q^*}{\partial S}\right|_{S=s} = \frac{k}{\theta s^2} > 0 \tag{12}$$

Prompting Cost: As the prompting cost k increases, the optimal quality Q^* decreases.

$$\left.\frac{\partial Q^*}{\partial K}\right|_{K=k} = -\frac{1}{\theta s} < 0 \tag{13}$$

In the three derivatives above, the capital letters Θ, S and K correspond to variables with respect to which derivatives are taken, and small letters θ, s and k are the values at which derivatives are evaluated.

E.1.3 Model Improvement and Skill Heterogeneity

We next show that improvements in model capacity reduce the performance gap between high and low prompting skills. This result is mainly due to the diminishing returns in output quality.

Result. *As model capacity θ improves, the effect of user skill s on optimal quality Q^* diminishes.*

Proof: This can be easily shown by taking the derivative of equation 12, which denotes the total effect of user skill, with respect to a model improvement (θ).

$$\left.\frac{\partial^2 Q^*}{\partial S \partial \Theta}\right|_{S=s,\Theta=\theta} = -\frac{k}{\theta^2 s^2} < 0 \tag{14}$$

As before capital letters Θ, S correspond to variables that are evaluated at θ, s. □

E.1.4 Decomposition into Model and Prompt Effects

As discussed in the main text, we decompose the total quality improvement into two parts: a *model effect* (the improvement from upgrading the model but holding the prompting effort fixed) and a *prompting effect* (the improvement from re-optimizing the effort in response to the upgraded model). To perform this decomposition, we consider a "counterfactual" quality function that uses one model's capacity but another model's optimal prompt:

$$Q\big(\theta_2, s, x^*(\theta_1, s)\big) = 1 - \exp\big(-\theta_2 s\, x^*(\theta_1, s)\big) \tag{15}$$

which is the quality obtained by using the model θ_2 while choosing the prompting effort $x^*(\theta_1, s)$ that would have been optimal if the user actually had model θ_1. Note that if $\theta_1 = \theta_2$, then $x^*(\theta_1, s)$ is indeed the true optimum for θ_2, yielding Q^*.

Given the formulation above, we can now express the *model effect* (M) and the *prompting effect* (P) by taking the derivative of the counterfactual quality in equation 15 with respect to the model capacity directly or indirectly through its effect on the prompt. The model effect is the derivative

of 15 with respect to the direct effect of model capacity, while keeping the user's effort fixed.

$$\begin{aligned} M &= \left.\frac{\partial Q\big(\Theta_2, s, x^*(\Theta_1, s)\big)}{\partial \Theta_2}\right|_{\Theta_1=\Theta_2=\theta} \\ &= s \cdot x^*(\Theta_1, s) \cdot \exp\left(-\Theta_2 s \cdot x^*(\Theta_1, s)\right)\Big|_{\Theta_1=\Theta_2=\theta} \\ &= \frac{k}{\theta^2 s} \cdot \ln\left(\frac{\theta s}{k}\right) \end{aligned} \tag{16}$$

We can obtain the prompt effect by taking the derivative of 15 with respect to its indirect impact on the user's effort, while keeping the direct impact of the model capacity fixed.

$$\begin{aligned} P &= \left.\frac{\partial Q\big(\Theta_2, s, x^*(\Theta_1, s)\big)}{\partial \Theta_1}\right|_{\Theta_1=\Theta_2=\theta} \\ &= \Theta_2 s \cdot \exp\left(-\Theta_2 s \cdot x^*(\Theta_1, s)\right) \cdot \left.\frac{\partial x^*(\Theta_1, s)}{\partial \Theta_1}\right|_{\Theta_1=\Theta_2=\theta} \\ &= \frac{k}{\theta^2 s}\left(1 - \ln\left(\frac{\theta s}{k}\right)\right) \end{aligned} \tag{17}$$

We now show that the model effect mainly benefits low-skilled users in contrast to the prompting effect which mainly benefits the high-skilled users.

Proposition 3. *As user skill s increases, the model effect becomes smaller, for sufficiently small prompting cost k.*

Proof. To examine how the model effect varies by user skill, we can take the derivative of the model effect M in equation 16 with respect to s.

$$\begin{aligned} \left.\frac{\partial M}{\partial S}\right|_{S=s} &= \left.\frac{\partial}{\partial S}\frac{k}{\theta^2 S} \cdot \ln\left(\frac{\theta S}{k}\right)\right|_{S=s} \\ &= \frac{k}{\theta^2 s^2}\left(1 - \ln\left(\frac{\theta s}{k}\right)\right) < 0 \end{aligned}$$

for sufficiently small k. □

Intuitively when prompting costs are low, users can afford to put substantial effort into writing prompts (so x^* is high). Once the effort is large enough, the marginal effect of the model on quality (i.e. the model effect) will be decreasing by user skill. This happens due to the diminishing returns in the quality function. Once a high-skill user is already achieving near-peak quality by putting in large effort, small increases in θ deliver small additional benefit. However, the same model improvement yields a much larger marginal impact for a low-skilled user who is putting the same level of effort.

Proposition 4. *As user skill s improves, the prompting effect becomes larger, for sufficiently small cost k.*

Proof. Proof To examine how the prompt effect varies by user skill, we can again take the derivative of the prompting effect P in equation 17 with respect to s.

$$\begin{aligned} \left.\frac{\partial P}{\partial S}\right|_{S=s} &= \left.\frac{\partial}{\partial S}\frac{k}{\theta^2 S}\cdot\left(1-\ln(\frac{\theta S}{k})\right)\right|_{S=s} \\ &= \frac{k}{\theta^2 s^2}\left(\ln\left(\frac{\theta s}{k}\right)-2\right) > 0 \end{aligned} \tag{18}$$

for sufficiently small k. □

Comparing equations E.1.4 and 18, we conclude that the prompting effect is less sensitive to user skill than the model effect, as its marginal with respect to skill has a smaller magnitude. This makes the overall marginal effect of user skill negative, matching the earlier finding in equation 14.

E.2 Task Performance and ATEs

The top pane of Figure 1B compares the average performance across models and attempt numbers (also referred to as iterations). It displays the average cosine similarity score stratified by the reference image. A notable feature in this figure is the performance dip during the second recreation attempt across both treatment arms. This is likely due to participants' initial misunderstanding of the model's "memoryless" nature. Participants failed to recognize that context from previous prompts was not carried over to new iterations. We observed numerous prompts in the second iteration across users that explicitly referenced the first prompt, a behavior that rarely occurred in subsequent attempts. However, from the third prompt onward, participants appeared to grasp the independence of each attempt, as evidenced by a marked decrease in cross-prompt references and a corresponding rebound in performance.

The bottom pane of Figure 1B shows the average treatment effect (ATE) per iteration, which is the difference between the stratified averages of DALL-E 3 and DALL-E 2 in the top pane.[10] To test the widening impact of using DALL-E 3 on performance relative to DALL-E 2, we run the following fixed effects linear model with participant-level (i) clustered standard errors where iteration is treated as a numeric variable:

$$\begin{aligned} Y_{i,n,t} = \beta_0 &+ \beta_1 \,\text{iteration} + \beta_2 \,\mathbb{I}[\text{dalleVersion} = 3]_i \\ &+ \beta_3 \,\text{iteration} \times \mathbb{I}[\text{dalleVersion} = 3]_i + \gamma_t + \epsilon_{i,n,t} \end{aligned} \tag{19}$$

The coefficient estimates generated by this model are:

- $\hat{\beta}_1 = 0.0011$, $\hat{SE}(\beta_1) = 0.0003$, $p = 0.0004$
- $\hat{\beta}_2 = 0.0120$, $\hat{SE}(\beta_2) = 0.0037$, $p = 0.0013$

[10] In Section E, when we refer to "DALL-E 3", we mean "DALL-E 3 (Verbatim)" unless otherwise specified.

- $\hat{\beta}_3 = 0.0010$, $\hat{SE}(\beta_3) = 0.0004$, $p = 0.0227$

The overall ATEs that we report between different pairs of treatment arms (DALL-E 2, DALL-E 3, and DALL-E 3 with revisions) in the main text are estimated from a two-way fixed effect (iteration and target image) model per each pair. Standard errors are cluster robust at the participant level.

E.3 Prompt Characteristics

Figure 1C compares the prompt length and prompt similarity of the two models. To generate these results, we first remove any prompt that does not constitute a good-faith attempt according to the sample construction procedure detailed in Section C.C.4. The prompt length is the average number of words per model and iteration stratified by the reference image. The prompt similarity is the average cosine similarity between all consecutive pairs of user prompts, which are both determined to be valid attempts, stratified by the reference image (see Section C.C.5.3 for details on similarity calculations).

The color scale in Figure 1C shows the stratified average similarity to the previous prompt across all users per each model. We find that superior model users write prompts that, on average, have $\beta = 0.0184$ higher in cosine similarity to their previous prompts using cluster robust standard errors at the participant level ($p = 0.0236$).

Comparing the aggregate similarity of all attempts made by a given participant, we also find the prompts from DALL-E 3 participants were, on average, more similar than the prompts of the inferior model participants. For this analysis, we use the dispersion around the centroid in the prompt embedding space, explained in Section C.C.5.3, as the dependent variable. When we average across all participants by model, we find that the average distance of prompts written by superior model participants to their centroid is $\beta = 0.0191$ smaller than inferior users ($p = 0.0083$). Standard errors are cluster-robust at the participant level.

E.4 ATE Decomposition

Figure 2 in the main text decomposes the ATE into the model and prompting effects. This decomposition is conceptually similar to a simple mediation analysis, with an important difference being that we can observe counterfactual outcomes (e.g., prompting the superior model as if it is the inferior model). This is not typically the case in mediation analysis, and makes causal identification rely on fewer assumptions.

To obtain counterfactual outcomes, we fed or "replayed" the participant prompts when interacting with one model (e.g., inferior) on another model (e.g., superior). The notation *(prompt, model)* specifies which treatment arm the prompts were written under and which model was used in the replay. For example, (2,3) indicates replaying prompts written under DALL-E 2 on DALL-E 3. To be clear, (2,2) and (3,3) correspond to the original observed treatment arms, while (2,3) and (3,2) are the counterfactual outcomes of interest.[11]

[11] To avoid problems with model drift, we regenerated images for all four possible combinations at the same time

The left-most point in Figure 2 corresponds to the average CLIP cosine similarity to the target image of (2,2). To make the interpretation of the results clearer, we have subtracted this quantity from all average quality scores and added a dashed line throughout. The second point from the left corresponds to average similarity to the target of (2,3), the third point from the left to (3,3), and the rightmost points to (3,2). All average similarity scores are stratified by iteration and reference image, and the standard errors are bootstrapped and cluster-robust at the participant level.

The model effect, as shown by the red braces in Figure 2, corresponds to the average increase in quality of (2,3) relative to (2,2). In the terminology of mediation analysis, the model effect would be referred to as the direct effect. The prompting effect, as shown by the blue braces in Figure 2, corresponds to the average increase in quality of (3,3) relative to (2,3). In the terminology of mediation analysis, the prompting effect would be referred to as the indirect effect. We can also test the difference in average quality between (3,3) and (3,2), as well as the difference between (3,2) and (2,2). Both of these differences are visible in Figure 2; the second is small and not statistically significant.

The standard errors in Figure 2 correspond to the uncertainty around the estimated average score for each of the four replay conditions. These uncertainty estimates are insufficient for exact inference on the direct, indirect, and treatment effects. The statistics and significance values reported in the main text, which correspond to such effects (i.e., the difference between average estimates in two conditions) are obtained using a two-way (iteration and target image) fixed effect model with the effect type as the main independent variable:

$$Y_{i,n,t} = \beta_0 + \beta_1 \text{effect} + \alpha_n + \gamma_t + \epsilon_{i,n,t} \tag{20}$$

where β_1 is the coefficient on the effect type in question (i.e., model or prompting). To estimate the different effects, we simply use the above model and filter the data as appropriate. For example, to estimate the ATE, the data contains all (2,2) and (3,3) scores, and in this case effect=1 for observations in (3,3) group. Similarly, to estimate the direct or model effect, the data contains all (2,2) and (2,3) scores and effect=1 for observations in (2,3) group. Finally, to estimate the indirect or prompting effect, the data contains all (2,3) and (3,3) scores and effect=1 for observations in (3,3) group. The standard errors for each estimated model are cluster robust at the participant level, and p-values are adjusted accordingly.

E.5 Heterogeneity by Skill

Table 1 in the main text demonstrates how total, model, and prompting effects vary across different user skill levels. The total effect compares the Cosine similarity of outputs from DALL-E 3 users replayed on DALL-E 3 to those from DALL-E 2 users replayed on DALL-E 2. The model effect compares outputs from DALL-E 2 users replayed on DALL-E 3 against the same users replayed on DALL-E 2. The prompting effect compares outputs from DALL-E 3 users replayed on DALL-E 3

and used these images for all analyses in the main text.

to those from DALL-E 2 users replayed on DALL-E 3.

To assess user skill, users within each target image, iteration and *replay scenario* are divided into 50 equally sized brackets based on their performance, assigning each bracket a rank corresponding to performance deciles. For instance, when evaluating DALL-E 2 users replayed on DALL-E 2, each user's skill level is determined by their percentile rank among all DALL-E 2 users replayed on DALL-E 2. Conversely, when the same user is replayed on DALL-E 3 (to estimate the model effect or to serve as a baseline in estimating the prompting effect), their performance decile is determined by their rank relative to all DALL-E 2 users replayed on DALL-E 3, per each iteration and target image.

After assigning performance deciles, we estimate the following linear model with two-way fixed effects (iteration and target image) for each effect type (model, prompting, and total):

$$\begin{aligned} Y_{i,n,t} =& \beta_0 + \beta_1 \text{effect} + \beta_2 \text{effect} \times \text{Performance Decile}_{i,n,t} + \\ & \alpha_n + \gamma_t + \epsilon_{i,n,t} \end{aligned} \tag{21}$$

where β_1 represents the coefficient associated with the effect type being analyzed (model, prompting, or total), and β_2 captures the interaction between the effect type and user skill level. User skill is measured as the user's rank (binned) within the same iteration, target image, and replay scenario. Standard errors are calculated using cluster-robust methods at the participant level.

This methodology closely follows the procedure described earlier in section E.4, with the additional step of incorporating an interaction term to evaluate skill-level heterogeneity.

F Robustness Checks

This section provides additional analyses that confirm the robustness of our main findings using alternative metrics and statistical approaches.

F.1 DreamSim-based Analysis

As discussed in Section C, we repeated all main-text analyses using DreamSim as an alternative image similarity metric. The results, presented below, demonstrate that our findings are not dependent on the specific similarity measure used.

F.1.1 Overall ATEs

In terms of DreamSim, participants using DALL-E 3 (the superior model) produced images that were, on average, $z = 0.238$ standard deviations (95% CI = [0.152, 0.324]) closer to the target image ($\Delta DreamSim = 0.0306$, $p < 10^{-7}$) than those produced by participants using DALL-E 2 (the inferior model).

F.1.2 Figure 1

We reran the regression in Section E with $Y_{i,n,t}$ representing the DreamSim outcome instead of cosine similarity:

$$\begin{aligned} Y_{i,n,t} = \beta_0 + \beta_1 \,\text{iteration} + \beta_2 \,\mathbb{I}[\text{dalleVersion} = 3]_i \\ + \beta_3 \,\text{iteration} \times \mathbb{I}[\text{dalleVersion} = 3]_i + \gamma_t + \epsilon_{i,n,t} \end{aligned} \tag{22}$$

The coefficient estimates generated by this analysis are:

- $\hat{\beta_1} = 0.0034$, $\hat{SE}(\beta_1) = 0.0005$, $p = 1.3 \times 10^{-12}$
- $\hat{\beta_2} = 0.0200$, $\hat{SE}(\beta_2) = 0.0061$, $p = 0.0011$
- $\hat{\beta_3} = 0.0024$, $\hat{SE}(\beta_3) = 0.0007$, $p = 0.0009$

F.1.3 Figure 2

Decomposing the ATE as measured in terms of DreamSim, we find similar results to those in the main text. The model effect accounts for 54.4% of the ATE ($\Delta DreamSim = 0.0166$, $p < 10^{-7}$), whereas the prompting effect accounts for 45.4% of the ATE ($\Delta DreamSim = 0.01390$, $p = 0.014$).

F.2 Z-score-based Analysis

As we discuss in Section D.2, we repeated all main-text analyses using the within-image-attempt Z-score of CLIP-cosine similarity to account for variation between images and attempts. When comparing across attempts, the Z-score was computed within each image, as mentioned in Section D.2. The results remain consistent throughout, and in some cases, the differences between the superior and inferior model are even more pronounced than when using raw cosine similarity.

F.2.1 Overall ATEs

As mentioned in the main text, participants using DALL-E 3 (the superior model) produced images that were, on average, $z = 0.19$ standard deviations (obtained from ATE in terms of Z-Scored Cosine Sim = 0.19, 95% CI = [0.100, 0.271]) closer to the target image ($\Delta CoSim = 0.0164$, $p < 10^{-5}$) than those produced by participants using DALL-E 2 (the inferior model). Standard errors are clustered at the participant level.

F.2.2 Figure 1

On average, participants using the superior model produced images that were $z = 0.19$ standard deviations closer (the ATE) to the target image than those using the inferior model. Like in the main text with CLIP cosine similarity, this treatment effect increased as participants made successive attempts to replicate the target image.

$$\begin{aligned} \text{ZScore}_{i,n} = \beta_0 + \beta_1 \,\text{iteration} + \beta_2 \,\mathbb{I}[\text{dalleVersion} = 3]_i \\ + \beta_3 \,\text{iteration} \times \mathbb{I}[\text{dalleVersion} = 3]_i + \epsilon_{i,n} \end{aligned} \tag{23}$$

- $\hat{\beta}_1 = 0.0129$, $\hat{SE}(\beta_1) = 0.0038$, $p = 0.0007$
- $\hat{\beta}_2 = 0.1250$, $\hat{SE}(\beta_2) = 0.0457$, $p = 0.0064$
- $\hat{\beta}_3 = 0.0128$, $\hat{SE}(\beta_3) = 0.0053$, $p = 0.015$

F.2.3 Figure 2

When we decompose the ATE into the model effect ($z = 0.0791$; $p = 8.35 \times 10^{-6}$) and prompting effect ($z = 0.1046$; $p = 0.016$), they account for 43% and 56% of the treatment effect, respectively. And when we replay the inferior model prompts on the superior model, the difference in similarity to the target from these prompts played on the inferior is not statistically significant and close together ($z = -0.033$; $p = 0.45$). In short, Figure 2 in the main text is quantitatively and qualitatively unchanged when using the within-image Z-score of the cosine similarity.

G Pre-registration

Prior to conducting our experiment, we developed and pre-registered a comprehensive set of hypotheses and analyses. This pre-registration is deposited at OSF at the following URL: **URL removed to maintain anonymity**. While the main text focuses on a subset of our pre-registered analyses that constitute the most important and timely findings, this appendix provides a comprehensive overview of all analyses specified in our pre-registration.

In our pre-registration, we outlined plans to conduct each analysis using six[12] possible outcome variables, all representing different transformations of the same underlying data. These include CLIP embedding cosine similarity and DreamSim, each with three rescaling methods:

- **No rescaling**: The outcome variable used in its original form.
- **Z-score rescaling**: The outcome variable transformed into a Z-score following the procedure detailed in Section D.2.
- **Percentile rank rescaling**: The outcome variable converted to a percentile rank relative to all other prompts submitted for the same target image.

While we are still in the process of completing all analyses with each of these outcome variables, our preliminary results suggest that our findings are robust across these different transformations. As additional analyses are completed, we will update this appendix. Full results, data, and replication code available online upon publication.

[12]See section G.2 on deviations from the pre-registration for the remaining two pre-registered outcome variables.

G.1 Hypotheses and Results

Below, we present each hypothesis exactly as stated in our pre-registration document, followed by a summary of our findings. This comprehensive review demonstrates the robustness of the key results highlighted in the main text, while also providing additional insights into the relationship between model capabilities, user behavior, and task performance.

H1

There are differences in prompt engineering ability (as measured through metrics such as average expected prompt quality, initial expected prompt quality, and max expected prompt quality) across demographic attributes and other observables, such as educational background and occupational skills.

Analysis approach: For this hypothesis, we conducted multiple ANOVA tests, one for each demographic variable, with the relevant performance measure as the dependent variable and the model (DALL-E version) as a covariate. As a robustness check, we repeated these analyses using the non-parametric Kruskal-Wallis U test. To account for multiple comparisons, we applied the Benjamini-Hochberg adjustment with a false discovery rate of 0.05.

Results: Our analyses revealed several consistent demographic patterns in prompt engineering ability across different outcome measures:

- **CLIP embedding cosine similarity**
  - **No rescaling:** Using ANOVA, we found significant associations between performance and: computer programming frequency, self-reported programming ability, outlook towards generative AI, age, gender, generative AI use, education, imagery writing skill, and self-reported occupational skills (particularly critical thinking, active listening, and quality control).

    In linear models examining directionality, we found that participants who reported critical thinking as a job skill and little usage of generative AI or imagery writing skill performed better, on average. Conversely, older participants, men, those with more positive outlooks regarding generative AI, those reporting quality control as an occupational skill, and frequent programmers performed worse on our task.

    The more conservative Kruskal-Wallis tests identified fewer significant variables: self-reported programming frequency, outlook towards generative AI, self-reported programming skill, gender, and age.
  - **Z-score rescaling:** ANOVA tests revealed significant associations with similar demographic factors as the unscaled measure, plus additional significant associations with self-reported occupational skills in technology design, social perceptiveness, and troubleshooting.

Linear models showed that participants reporting critical thinking and troubleshooting as job skills, little generative AI use, little imagery writing skill, some programming skill, and some instruction writing skill performed better. Worse performance was associated with older participants, men, positive generative AI outlook, graduate degrees, self-reported quality control and technology design skills, and frequent programming experience.

Kruskal-Wallis tests again identified a smaller set of significant predictors: self-reported programming frequency, generative AI outlook, programming skill, gender, and age.

- **Percentile rank rescaling:** ANOVA identified significant associations between performance and computer programming frequency, self-reported programming ability and frequency, generative AI outlook, age, gender, generative AI use, and critical thinking as an occupational skill.

  Linear models showed better performance among those reporting critical thinking as a job skill, little generative AI use, and some programming skill. Worse performance was associated with older participants, men, positive generative AI outlook, and frequent programming experience.

  Kruskal-Wallis tests revealed the same set of significant predictors as with other scalings: programming frequency, generative AI outlook, programming skill, gender, and age.

- **DreamSim**

  - **No rescaling:** ANOVA tests identified significant associations between performance and: programming frequency, self-reported programming ability, generative AI outlook, age, gender, generative AI use, imagery and instructional writing skills, and several occupational skills (critical thinking, learning strategies, technology design, and quality control).

    Linear models showed better performance among those reporting critical thinking and social perceptiveness as job skills, little generative AI use, and little imagery writing or programming skills. Worse performance was associated with older participants, men, positive generative AI outlook, and frequent programming experience.

    Kruskal-Wallis tests identified significant relationships with: programming frequency, generative AI outlook, imagery writing skill, gender, age, and learning strategies as an occupational skill.

  - **Z-score rescaling:** Using ANOVA, significant associations included all factors identified in the unscaled analysis, plus education and additional occupational skills (social perceptiveness).

    Linear model directional findings were consistent with the unscaled measure, with additional negative associations for graduate degrees and self-reported technology design, quality control, and learning strategies skills.

Kruskal-Wallis tests showed significant relationships with: programming frequency, generative AI outlook, imagery writing skill, gender, age, and occupational skills in learning strategies, social perceptiveness, and technology design.

- **Percentile rank rescaling:** ANOVA identified significant associations with: programming frequency, programming ability, generative AI outlook, age, gender, generative AI use, imagery writing skill, education, and occupational skills in social perceptiveness, learning strategies, and technology design.

  Linear models showed better performance among participants reporting social perceptiveness as a job skill, little generative AI use, and little imagery writing or programming skills. Worse performance was associated with older participants, men, positive generative AI outlook, frequent programming experience, and self-reported technology design and learning strategies skills.

  Kruskal-Wallis tests revealed significant relationships with: programming frequency, generative AI use, generative AI outlook, imagery writing skill, gender, age, and occupational skills in learning strategies, social perceptiveness, and technology design.

Across these analyses, several consistent patterns emerge. Interestingly, we found that frequent programming experience was associated with worse task performance, contrary to what might be expected. Men and older participants also performed worse on average. Those reporting critical thinking or social perceptiveness as job skills tended to perform better, while participants with positive outlooks toward generative AI and higher generative AI usage performed worse. These patterns were generally consistent across different outcome measures and scaling approaches, suggesting robust demographic differences in prompt engineering ability.

H2

There are observable differences in the prompting techniques of successful prompt engineers and unsuccessful prompt engineers. Such prompting techniques might include the use of longer prompts, the use of structured prompting techniques, and/or specific patterns in the way that the participant iterates on their prompts over time.

Analysis approach: We investigated this hypothesis by examining the relationship between exploration/exploitation strategies in the prompting space and performance outcomes. We characterized participants as more "exploitative" if they wrote prompts that were similar to their previous prompts, and more "exploratory" if their prompts exhibited greater deviation from previous attempts. We operationalized these concepts using several metrics:

1. **Measures positively associated with exploitation**:

   - Average token sort ratio compared to the previous prompt
   - Average cosine similarity between the embeddings of consecutive prompts

- Fraction of times a prompt contains the previous prompt as an exact substring

2. **Measures negatively associated with exploitation**:

   - Variance of the prompt embedding
   - Number of topical transitions

We also measured each participant's average prompt length. These variables were calculated for each user across their first 10 attempts, and their association with performance was estimated using linear models with DALL-E version fixed effects.

To examine how exploration/exploitation strategies influence performance across successive attempts, we conducted an additional analysis at the iteration level. We divided user-iteration observations into 6 equal-sized brackets based on performance in the previous iteration, allowing us to explore how prompting behavior might vary depending on the quality of previous attempts. We then estimated the effect of textual similarity to the previous prompt on the quality of the next attempt within each bracket. Our estimates adjusted for other covariates by matching user-iteration observations on target image, DALL-E model, iteration number, and exact quality of the previous attempt (Sävje et al. 2021).

Results: Our analyses revealed consistent patterns across different outcome measures:

- **CLIP embedding cosine similarity**
  - **No rescaling:** At the user level, we found strong and statistically significant associations between performance and prompting strategies, even after Benjamini-Hochberg adjustment for multiple testing. Specifically, token sort ratio, cosine similarity with the previous prompt, and frequency of including the previous prompt were all positively associated with performance. Conversely, embedding variance and number of topical transitions showed negative correlations with performance. Together, these findings indicate that more successful users engaged in greater exploitation, writing prompts that exhibited higher similarity to one another. As noted in the main text, we also found that longer prompts were associated with higher performance.

    At the user-prompt level, we observed an interesting pattern: when previous performance was poor, moderate exploration (lower cosine similarity with the previous prompt) was associated with improved subsequent performance, though extremely high levels of exploration did not yield additional benefits. In contrast, when previous performance was high, increased exploitation consistently improved subsequent performance. Similar results emerged when using token sort ratio as the measure of exploration.

    With binary measures of exploration (topical transitions or containing the previous prompt), exploitation was associated with higher performance in the next iteration regardless of previous performance bracket. Since these are binary measures, they couldn't capture the non-linear relationship observed with continuous measures in low-performing

groups. Nevertheless, we found that topical transitions (more exploration) led to overall performance decreases, with larger decreases for higher previous performance. Similarly, prompts that included the previous prompt showed higher performance, with larger improvements as previous performance increased.

  - **Z-score rescaling:** Results were consistent with those for unscaled cosine similarity. The primary difference was that the non-linear relationship between performance and continuous measures of exploitation (TSR ratio and cosine similarity with previous prompt) became more pronounced for the bottom two brackets of previous performance. For low-performing prompts in the previous attempt, moderate exploitation levels yielded optimal performance, with significant performance deterioration at both high and low exploitation levels.
  - **Percentile rank rescaling:** We found the same general patterns described above, with the non-linearity in exploration/exploitation versus performance in the bottom two brackets of previous performance even stronger than observed with Z-score rescaled cosine similarity.

- **DreamSim**
  - **No rescaling:** Results were consistent with those described for unscaled cosine similarity.
  - **Z-score rescaling:** Results matched those described for Z-score cosine similarity, with particularly stark non-linearity between performance and continuous measures of exploitation at the user-prompt level for the bottom two brackets of previous performance.
  - **Percentile rank rescaling:** Results were consistent with those described for percentile rank rescaled cosine similarity.

These findings suggest a nuanced relationship between exploration/exploitation strategies and performance. Users who were generally more successful tended to employ more exploitative strategies, refining and building upon previous prompts rather than making dramatic changes. However, at the iteration level, the optimal strategy depended on previous performance: when performance was already high, continued exploitation yielded the best results; when performance was poor, moderate exploration was beneficial. This pattern was consistent across different outcome measures and scaling approaches.

H3

There are differences in prompt engineering techniques (as measured through metrics such as prompt length and iteration-to-iteration token sort ratio) across demographic attributes and other observables, such as educational background and occupational skills.

Analysis approach: To test this hypothesis, we estimated linear models with each demographic trait as the independent variable, treatment arm fixed effects as controls, and various prompting behaviors (described in Section C.C.5) as the dependent variables. To account for multiple testing, we adjusted p-values across all models using the Benjamini-Hochberg procedure.

Results: Our analysis revealed several significant demographic differences in prompting strategies, with consistent patterns emerging across different measures of prompt similarity and evolution. The detailed findings for each dependent variable are summarized below:

- **Prompt embedding variance:** We did not find any significant differences across demographic traits in the overall variance of prompt embeddings, suggesting that the breadth of prompt space exploration was relatively consistent across different demographic groups.
- **Strategic Shifts:** We observed statistically significant differences based on age, programming frequency, and instructional writing frequency. Specifically, younger participants, those with low programming frequency, and those with some instructional writing experience demonstrated fewer topical transitions across their prompts, indicating a more consistent approach to prompt development.
- **Successive Prompt Token Sort Ratio:** Significant differences emerged by age, education, programming frequency, and imagery/instructional writing frequencies. Older users, those with post-graduate degrees, those with high programming frequency, and those with high writing frequency (both for precise instructions and imagery) wrote prompts that were less similar to their previous attempts as measured by token sort ratio. This suggests these groups took a more exploratory approach to prompt iteration.
- **Successive similarity:** We found significant differences across numerous demographic factors: age, gender, education, generative AI outlook, programming skill/frequency, instructional/imagery writing frequency, and certain occupational skills. On average, older users, males, those with post-graduate degrees, frequent programmers, self-reported skilled programmers, those with positive outlooks toward generative AI, frequent generative AI users, those who frequently write instructions or imagery, and those reporting critical thinking and social perceptiveness as occupational skills wrote prompts with lower cosine similarity to their previous attempts. This consistent pattern across multiple demographic factors suggests robust differences in exploration-exploitation tendencies.
- **Successive Prompt 'Contains Previous Prompt' Dummy:** We found significant differences by age, gender, outlook toward generative AI, and imagery writing frequency. Older users and those with neutral outlooks toward generative AI were less likely to write prompts that contained their previous prompt as an exact substring. In contrast, males and those with some imagery writing skills were more likely to build directly upon their previous prompts by including them in subsequent attempts.

These findings reveal a nuanced picture of demographic differences in prompt engineering approaches. Overall, younger users, those with less programming experience, and those with less generative AI experience tended to employ more exploitative strategies, building more consistently upon their previous prompts. In contrast, older users, those with more technical backgrounds, and those with more exposure to generative AI systems exhibited more exploratory behaviors, making larger changes between successive prompts.

Interestingly, while H1 showed that a more exploitative approach was generally associated with better performance, here we find that demographics associated with technical expertise (programming experience, education, etc.) tend toward more exploratory approaches. This apparent contradiction suggests that the relationship between background, prompting strategy, and performance is complex and potentially mediated by other factors not captured in these analyses.

H4

Insofar as the output returned by a generative AI model in response to a prompt is stochastic, the subsequent prompting strategies and prompting outcomes of participants that get lower-than-expected, higher-than-expected, or approximately expected outputs in response to their first prompt are different.

Analysis approach: To test this hypothesis, we examined how the random variation in image quality resulting from model stochasticity affects subsequent user behavior. We first calculated a Z-score for each participant's generated image relative to the expected distribution for that prompt, following the procedure outlined in Section C.C.3 (with computational details in Section D). This Z-score quantifies how much better or worse the actually generated image was compared to what would be expected for that prompt on average. Higher Z-scores represent instances where the realized image quality exceeded expectations, while lower Z-scores indicate instances where the quality was randomly lower than expected.

To analyze the relationship between this stochasticity and subsequent prompting behavior and performance, we transformed the Z-score into a trichotomous variable with three categories: "lower-than-expected" (Z-score ¡ -0.45), "expected" (-0.45 $\leq$ Z-score $\leq$ 0.45), and "higher-than-expected" (Z-score ¿ 0.45). We then performed two-sample t-tests comparing performance and prompting behavior across these three groups. As a robustness check, we also estimated linear models regressing performance and prompting measures on both the trichotomous Z-score variable and the continuous Z-score, with treatment arm fixed effects. Additionally, we conducted analyses at the user level to determine whether the Z-score of the first prompt affects average performance across all subsequent attempts.

Results: Our analyses revealed several consistent patterns across different outcome measures:

- **CLIP embedding cosine similarity**
  - **No rescaling:** We found statistically significant evidence that higher Z-scores for images in previous prompts were associated with increased cosine similarity in subsequent at-

tempts. When comparing performance across the trichotomous Z-score variable, the difference between the top bracket ("higher-than-expected") and bottom bracket ("lower-than-expected") was statistically significant, with the top bracket showing better performance in the next attempt. However, we did not find significant differences between the middle bracket and either the top or bottom brackets.

The linear model did not yield a statistically significant relationship between Z-score and performance in the next iteration, likely due to non-linearity in this relationship across the negative and positive ranges of the Z-score distribution.

At the user level, we found differences between the top and middle brackets of the first prompt's Z-score, though these differences were marginally significant (p=0.048) and did not account for multiple testing. Linear models at the user level did not reveal statistically significant relationships between the first prompt's Z-score and average performance in subsequent attempts.

- **Z-score rescaling:** Results were similar to those for unscaled cosine similarity. We found a statistically significant difference between the top and bottom Z-score brackets, with higher performance in the next attempt for the top bracket. No statistically significant effects emerged when comparing the middle bracket with either the top or bottom brackets, or when using linear models. Unlike with the unscaled measure, we found no statistically significant effects at the user level, either when comparing Z-score brackets or when using linear models.
- **Percentile rank rescaling:** Results were consistent with those for Z-score rescaled cosine similarity.

- **DreamSim**
  - **No rescaling:** Results showed similar patterns to CLIP embedding cosine similarity. Higher Z-scores in previous prompts were associated with better performance in subsequent attempts. The difference between the top Z-score bracket and both the bottom and middle brackets was statistically significant, with the top bracket showing higher performance in the next attempt. However, the difference between the bottom and middle brackets was not significant.

    As with cosine similarity, linear models did not yield statistically significant relationships between Z-scores and subsequent performance, likely due to non-linearity. At the user level, we found no statistically significant relationships between the first prompt's Z-score and performance in subsequent attempts.
  - **Z-score rescaling:** Results closely matched those for unscaled DreamSim scores, with statistically significant differences between the top bracket and both the bottom and middle brackets, but no significant difference between the bottom and middle brackets. No significant effects emerged from linear models or from user-level analyses.

- **Percentile rank rescaling:** Results aligned with those for Z-score rescaled DreamSim, with one exception: at the user-attempt level, we observed a statistically significant difference only between the top and bottom brackets, not between the top and middle brackets.

- **Prompting Behaviors**
  - **Prompt Length:** We found statistically significant evidence that higher Z-scores led to longer subsequent prompts. When using the trichotomous variable, the difference between the top and bottom Z-score brackets was significant, with longer prompts on average in the top bracket. However, we did not observe significant differences between the middle bracket and either the top or bottom brackets. Linear models at the user-attempt level showed a positive and statistically significant relationship, with a one-unit increase in Z-score associated with prompts approximately 0.3 words longer on average. At the user level, the Z-score of the first prompt did not significantly affect the length of subsequent prompts.
  - **Successive Similarity:** Higher Z-scores were associated with increased similarity between consecutive prompts. The top Z-score bracket showed significantly higher prompt similarity compared to both the bottom and middle brackets, though no significant difference emerged between the bottom and middle brackets. Linear models confirmed a positive and statistically significant relationship between Z-scores and successive prompt similarity. No significant effects were found at the user level.
  - **Successive Prompt Token Sort Ratio:** Results mirrored those for successive cosine similarity, with higher Z-scores associated with increased token sort ratios between successive prompts. This pattern was consistent across both trichotomous variable comparisons and linear models. No significant effects emerged at the user level.
  - **Successive Prompt 'Contains Previous Prompt' Dummy:** Higher Z-scores significantly increased the likelihood that subsequent prompts would contain the previous prompt as an exact substring. The difference between top and bottom Z-score brackets was significant, though differences involving the middle bracket were not. Linear models at the user-attempt level confirmed a positive and statistically significant relationship. No significant effects were found at the user level.

These findings reveal a consistent pattern: when the stochastic nature of generative AI produces unexpectedly high-quality outputs for a given prompt, users tend to build more directly upon that prompt in their subsequent attempts—writing longer prompts that are more similar to and often directly incorporate the previous prompt. This adaptive behavior leads to better performance in subsequent attempts. However, this effect appears to be local rather than global; the quality of the first prompt affects the immediate next attempt but does not significantly influence overall performance or prompting behavior across all subsequent attempts.

This pattern suggests that users are sensitive to and learn from random variation in model outputs, even when they have no way of knowing whether a particularly good or bad result stems from the quality of their prompt or from model stochasticity. The tendency to build upon prompts that produce better-than-expected results represents an intuitive but potentially suboptimal learning strategy, as it may attribute too much importance to random variations rather than focusing on the underlying quality of the prompt itself.

H5

Average prompt engineering ability (as measured through metrics such as average expected prompt quality, initial expected prompt quality, and max expected prompt quality) and prompting strategies will depend on the capacity of the model that participants are interacting with.

Analysis approach: To test this hypothesis, we conducted pairwise comparisons across the three treatment arms using two-sample t-tests to evaluate differences in both performance outcomes and prompting behaviors. For robustness, we also performed ANOVA tests to examine whether any significant differences existed across all three treatment arms simultaneously. P-values were adjusted for multiple testing using the Benjamini-Hochberg procedure.

Results: Our analyses revealed several key differences in both performance and prompting behavior across the different model conditions:

- **CLIP embedding cosine similarity**
  - **Z-score rescaling:** We found statistically significant evidence for performance differences across treatment arms when examining participants' first attempts, average performance across all attempts, and best attempts (all measured using taskwide Z-scores rather than task-iteration Z-scores).

    For first attempt performance, DALL-E 3 (Verbatim) participants significantly outperformed both DALL-E 3 (Revised) and DALL-E 2 participants, while no statistically significant difference emerged between DALL-E 3 (Revised) and DALL-E 2 participants.

    Average performance across all attempts showed the same pattern: DALL-E 3 (Verbatim) outperformed both other conditions, with no significant difference between DALL-E 3 (Revised) and DALL-E 2.

    When comparing participants' best attempts, we found statistically significant differences between all three treatment arms in a clear hierarchy: DALL-E 3 (Verbatim) produced better results than DALL-E 3 (Revised), which in turn produced better results than DALL-E 2. ANOVA tests for all three performance measures (first, average, and best attempts) confirmed significant effects of model assignment.
  - **Percentile rank rescaling:** Results using percentile rank rescaling mirrored those found with Z-score rescaling. DALL-E 3 (Verbatim) participants outperformed both

other conditions on first attempt performance and average performance, with no significant differences between DALL-E 3 (Revised) and DALL-E 2 for these metrics. For best attempt performance, all pairwise comparisons revealed significant differences, with DALL-E 3 (Verbatim) outperforming DALL-E 3 (Revised), which outperformed DALL-E 2. ANOVA tests confirmed significant effects of model assignment for all three performance measures.

- **DreamSim**
  - **Z-score rescaling:** Results using DreamSim with Z-score rescaling showed the same pattern as CLIP embedding cosine similarity. DALL-E 3 (Verbatim) participants significantly outperformed both other conditions on first attempt and average performance, with no significant differences between DALL-E 3 (Revised) and DALL-E 2. For best attempts, we again found the same hierarchical pattern of performance differences across all three conditions. ANOVA tests confirmed significant effects across all performance measures.
  - **Percentile rank rescaling:** Results with percentile rank rescaling of DreamSim scores were consistent with the Z-score findings. DALL-E 3 (Verbatim) participants outperformed both other conditions on first and average performance, with no differences between DALL-E 3 (Revised) and DALL-E 2. For best attempts, all pairwise comparisons were significant, showing DALL-E 3 (Verbatim) ¿ DALL-E 3 (Revised) ¿ DALL-E 2. ANOVA tests confirmed significant effects across all performance measures.
- **Prompting Behaviors**
  - **Mean prompt Length:** We found statistically significant differences in prompt length across treatment arms. Participants using DALL-E 2 wrote significantly shorter prompts compared to both DALL-E 3 (Revised) and DALL-E 3 (Verbatim) groups. Interestingly, there was no significant difference in prompt length between the two DALL-E 3 variants. ANOVA results confirmed a significant effect of model version on prompt length.
  - **Aggregate Similarity:** We found no statistically significant differences in the overall variability of prompts across the three treatment arms. ANOVA results confirmed no significant effect of model version on this measure, suggesting that participants explored similar breadths of prompt space regardless of the model they were using.
  - **Successive Similarity:** Analysis of the average cosine similarity between successive prompts revealed no statistically significant differences across treatment arms. ANOVA results confirmed no significant effect of model version on successive prompt similarity, indicating that the tendency to build upon or deviate from previous prompts was similar across models.
  - **Successive Prompt Token Sort Ratio:** No statistically significant differences emerged across treatment arms. ANOVA results confirmed no significant effect of model version

on this measure of textual similarity between successive prompts.

- **Successive Prompt "Contains Previous Prompt" Dummy:** Examining the probability of a current prompt being a superset of the previous prompt showed no statistically significant differences across treatment arms. ANOVA results confirmed no significant effect of model version on this measure.

These findings paint an interesting picture of how model capacity influences both performance and prompting behavior. In terms of performance, DALL-E 3 (Verbatim) consistently outperformed the other conditions across all metrics and scaling approaches, demonstrating the clear advantage of the more advanced model when used without automated prompt revision. The DALL-E 3 (Revised) condition generally performed worse than DALL-E 3 (Verbatim) but better than DALL-E 2 on best attempt measures, suggesting that automated prompt revision partially degraded the benefits of the more capable model.

For prompting behavior, the primary difference observed was in prompt length: participants using either version of DALL-E 3 wrote significantly longer prompts than those using DALL-E 2. This suggests that users recognized and adapted to the increased capacity of the more advanced model by providing more detailed instructions. However, we found no significant differences across models in measures of prompt similarity or evolution, suggesting that while participants provided more detail when using more capable models, their overall strategies for iterating on prompts remained relatively consistent regardless of model assignment.

H6

Variability in participants' ability to prompt engineer effectively and prompting strategies will depend on the capacity of the model that participants are interacting with.

Analysis approach: To test this hypothesis, we conducted two types of analyses. First, we performed F-tests comparing the variance of participant performance and prompting behaviors between all three pairs of model conditions (DALL-E 2 vs. DALL-E 3 Revised, DALL-E 3 Revised vs. DALL-E 3 Verbatim, and DALL-E 2 vs. DALL-E 3 Verbatim). Second, we estimated quantile treatment effects (QTEs) between all three pairs of models on participant performance and prompting behaviors. We also visually inspected the QTE patterns to determine whether dispersion/"inequality" was being reduced or increased when participants used different models. For example, positive effects for low quantiles and negative/null effects for high quantiles would indicate inequality reduction.

F-test Results (with BH Adjusted p-values):

- **DALL-E 3 (revised) vs. DALL-E 2**
  - **Mean prompt length (words):** DALL-E 3 Revised showed significantly less variance than DALL-E 2 (ratio 0.5217, $p \leq 10^{-4}$).

- **Prompt embedding variance:** DALL-E 3 Revised also showed significantly less variance than DALL-E 2 (ratio 0.7123, $p = 2 \times 10^{-4}$).
- **Cosine similarity with target image, Z-Score:** DALL-E 3 Revised showed significantly more variance than DALL-E 2 (ratio 1.255, $p = 0.0159$).
- **Failed to reject null of no differences in variance:** Mean raw DreamSim score vs. target image (p = 0.1333), number of topical transitions (by token sort ratio) (p = 0.3294), mean token sort ratio with previous prompt (p = 0.6244), mean percentile rank of DreamSim score (p = 0.9035), mean cosine similarity to previous prompt (p = 0.8748), mean raw CosineSim score vs. target image (p = 0.8301), mean proportion of prompts containing previous prompt (p = 0.5461), Z-score of DreamSim score (p = 0.4016), number of topical transitions (cosine similarity) (p = 0.3505), and mean percentile rank of CosineSim (p = 0.0795).

- **DALL-E 3 (verbatim) vs. DALL-E 3 (revised)**
  - **Mean percentile rank of CosineSim:** DALL-E 3 (verbatim) showed significantly less variance than DALL-E 3 (revised) (ratio 0.7347, $p = 0.0009$).
  - **Mean proportion of prompts containing previous prompt:** DALL-E 3 (verbatim) also showed significantly less variance (ratio 0.7352, $p = 0.0009$).
  - **Mean percentile rank of DreamSim:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.7888, $p = 0.0141$).
  - **Cosine similarity with target image, Z-Score:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.7961, $p = 0.0159$).
  - **DreamSim with target image, Z-Score:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.8195, $p = 0.0377$).
  - **Number of topical transitions (token sort ratio):** DALL-E 3 (verbatim) showed significantly more variance (ratio 1.2303, $p = 0.0301$).
  - **Failed to reject null of no differences in variance:** Mean raw DreamSim score vs. target image (p = 0.1842), mean raw CosineSim score vs. target image (p = 0.5916), prompt embedding variance (p = 0.9771), mean cosine similarity to previous prompt (p = 0.9035), mean prompt length (words) (p = 0.6244), number of topical transitions (cosine similarity) (p = 0.5916), and mean token sort ratio with previous prompt (p = 0.5206).

- **DALL-E 2 vs. DALL-E 3 (verbatim)**
  - **Mean prompt length (words):** DALL-E 3 (verbatim) showed significantly less variance than DALL-E 2 (ratio 0.553, $p \leq 10^{-4}$).
  - **Prompt embedding variance:** DALL-E 3 (verbatim) also showed significantly less variance (ratio 0.7157, $p = 2 \times 10^{-4}$).

- **Mean raw DreamSim score vs. target image:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.7501, $p = 0.0015$).
- **Mean proportion of prompts containing previous prompt:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.7904, $p = 0.0141$).
- **Mean percentile rank of DreamSim:** DALL-E 3 (verbatim) showed significantly less variance (ratio 0.8005, $p = 0.0159$).
- **Failed to reject null of no differences in variance:** Mean percentile rank of CosineSim (p = 0.1842), Z-score of DreamSim score (p = 0.3294), Mean Raw CosineSim (p = 0.8151), Z-score of CosineSim score with target image (p = 0.9906), mean token sort ratio with previous prompt (p = 0.8748), mean cosine similarity to previous prompt (p = 0.807), number of topical transitions (token sort ratio) (p = 0.3505), and number of topical transitions (cosine similarity) (p = 0.085).

QTE Highlighted Results: Through our quantile treatment effects analysis and visual inspection of QTE plots, we found:

- **Evidence of dispersion reduction:** Z-score of cosine similarity with respect to the target image (calculated within task-iteration) for the DALL-E 3 Revised vs. DALL-E 2 comparison showed patterns consistent with inequality reduction, with larger positive effects at lower quantiles.
- **Evidence of dispersion increase:** Mean prompt length for all three pairwise model comparisons (DALL-E 3 Verbatim vs. DALL-E 2, DALL-E 3 Revised vs. DALL-E 2, and DALL-E 3 Verbatim vs. DALL-E 3 Revised) showed patterns consistent with increased inequality, with larger positive effects at higher quantiles. Similar patterns emerged for raw DreamSim performance in both DALL-E 3 Verbatim vs. DALL-E 2 and DALL-E 3 Revised vs. DALL-E 2 comparisons.

These findings present a nuanced picture of how model capacity affects variability in performance and prompting behaviors. For prompt length and embedding variance, the more advanced models (both DALL-E 3 variants) showed significantly less variance than DALL-E 2, suggesting more consistent prompting behaviors. For performance measures, the results were mixed: DALL-E 3 (Verbatim) generally showed less variance than DALL-E 3 (Revised), indicating more consistent performance across participants, while DALL-E 3 (Revised) showed higher variance in performance than DALL-E 2.

The QTE analysis suggests that the relationship between model capacity and performance inequality is complex. For some measures (e.g., Z-score of cosine similarity), more advanced models appeared to reduce inequality by disproportionately benefiting lower-performing users. For other measures (e.g., prompt length and raw DreamSim scores), more advanced models appeared to increase inequality by disproportionately benefiting higher-performing users.

H7

As participants repeatedly try to complete a task with a given model, the quality of their attempts will increase, and the extent to which the quality increases varies as a function of model capacity.

Analysis approach: To evaluate how performance improves across successive attempts and whether this improvement varies by model, we conducted stratified two-sample tests comparing participants' initial scores with their best scores. These comparisons were performed both within each treatment arm (DALL-E 2, DALL-E 3 Verbatim, and DALL-E 3 Revised) and overall across all participants. This approach allowed us to determine both whether participants generally improved with practice and whether the rate of improvement differed across models with varying capabilities.

Results: Our analyses revealed consistent performance improvements across all treatment arms and outcome measures:

- **CLIP embedding cosine similarity**
  - **No rescaling:** We found statistically significant improvements from initial to best performance both within each treatment arm and in the overall sample. This indicates that participants were able to improve their performance through iterative attempts regardless of which model they were using.
  - **Z-score rescaling:** Results mirrored those of the unscaled analysis, with statistically significant improvements from initial to best performance observed both within each treatment arm and overall.
  - **Percentile rank rescaling:** Consistent with other scaling approaches, we observed statistically significant improvements from initial to best performance both within each treatment arm and overall.
- **DreamSim**
  - **No rescaling:** DreamSim results aligned with CLIP similarity findings, showing statistically significant improvements from initial to best performance both within each treatment arm and overall.
  - **Z-score rescaling:** We again found statistically significant improvements from initial to best performance both within each treatment arm and in the overall sample.
  - **Percentile rank rescaling:** Results were consistent with other analyses, showing statistically significant improvements from initial to best performance across all conditions.

These findings demonstrate that participants consistently improved their performance through iterative attempts across all model conditions and performance measures. Users were able to learn from feedback and refine their prompts to achieve better results over time, regardless of which model they were using. This learning effect was robust across different outcome measures and

scaling approaches, highlighting the importance of iteration and feedback in developing effective prompting strategies.

Although our original hypothesis also posited that the rate of improvement might vary as a function of model capacity, we did not find consistent evidence for differences in improvement rates across models. This suggests that while more capable models yield better absolute performance (as demonstrated in H5), the relative improvement from first to best attempt was similar across model conditions. This finding highlights that the benefits of iterative refinement apply broadly across different model capabilities.

H8

The extent to which participants can recreate images using models such as DALL-E 2/3 will vary across images.

Analysis approach: To evaluate this hypothesis, we employed two complementary approaches. First, we used GPT-4V (a multimodal generative AI model capable of processing both text and images) to generate optimal prompts for each target image. For each of our 15 target images, we instructed GPT-4V to "Write a DALL-E 2, 3 prompt to recreate this image verbatim as closely and as detailed as possible." We generated two such "AI prompts" per target image—one optimized for DALL-E 2 and one for DALL-E 3. We then submitted these AI-generated prompts to the respective models 20 times each, yielding 60 replicated images per target (as the DALL-E 3 prompts were sent to both DALL-E 3 variants). We measured the cosine similarity between CLIP embeddings of these generated images and their target images, then averaged these similarity scores to quantify GPT-4V's ability to generate effective replication prompts for each target.

Second, we analyzed human performance by measuring the similarity between all participant-generated images and their corresponding target images, averaging these similarities to determine which images were easier or harder for participants to replicate. This two-pronged approach allowed us to assess image difficulty from both AI and human perspectives.

Results:

- **CLIP embedding cosine similarity:** When ranking target images by GPT-4V's ability to generate effective prompts, we found substantial variation across images:

  1. Business Image #3
  2. Business Image #5
  3. Business Image #1
  4. Photography Image #1
  5. Design Image #2
  6. Business Image #2
  7. Photography Image #5
  8. Design Image #4
  9. Business Image #4
  10. Design Image #1
  11. Photography Image #2
  12. Design Image #5
  13. Design Image #3
  14. Photography Image #3
  15. Photography Image #4

  The highest average cosine similarity (easiest image for GPT-4V to replicate) was $CoSim = 0.944$ for Business Image #3, while the lowest score (hardest image to replicate) was $CoSim = 0.734$ for Photography Image #4.

When ranking target images by participants' ability to generate effective prompts, we observed:

1. Business Image #3
2. Business Image #5
3. Business Image #1
4. Photography Image #5
5. Design Image #2
6. Design Image #4
7. Business Image #2
8. Design Image #5
9. Design Image #1
10. Photography Image #1
11. Design Image #3
12. Photography Image #2
13. Photography Image #3
14. Photography Image #4
15. Business Image #4

The highest average cosine similarity (easiest image for participants to replicate) was $CoSim = 0.892$ for Business Image #3, while the lowest score (hardest image to replicate) was $CoSim = 0.669$ for Business Image #4.

- **DreamSim:** When using DreamSim as our similarity metric (inverted as 1-DreamSim to create a similarity rather than distance measure), GPT-4V's prompt generation performance ranked as follows:

1. Design Image #3
2. Business Image #2
3. Business Image #4
4. Design Image #2
5. Business Image #1
6. Business Image #5
7. Design Image #4
8. Business Image #3
9. Photography Image #1
10. Photography Image #5
11. Photography Image #2
12. Design Image #1
13. Design Image #5
14. Photography Image #3
15. Photography Image #4

The highest 1-DreamSim score (easiest image for GPT-4V to replicate) was $\tilde{D} = 0.75$ for Design Image #3, while the lowest score (hardest image to replicate) was $\tilde{D} = 0.40$ for Photography Image #4.

When ranking target images by participants' replication ability using DreamSim:

1. Business Image #3
2. Business Image #2
3. Business Image #5
4. Business Image #1
5. Design Image #2
6. Design Image #3
7. Photography Image #2
8. Photography Image #5
9. Design Image #5
10. Design Image #4
11. Business Image #4
12. Photography Image #1
13. Photography Image #4
14. Design Image #1
15. Photography Image #3

The highest 1-DreamSim score (easiest image for participants to replicate) was $\tilde{D} = 0.575$ for Design Image #3, while the lowest score (hardest image to replicate) was $\tilde{D} = 0.356$ for Photography Image #4.

These results clearly demonstrate substantial variation in image replicability across our target set. Business-related images were generally easier to replicate, while photography images tended to be more challenging. While the exact rankings varied somewhat between GPT-4V and human

participants, and between similarity metrics, the overall pattern of relative difficulty remained fairly consistent. This variation in difficulty confirms the importance of our stratified analysis approach and underscores the need to consider image-specific characteristics when evaluating generative AI performance.

Exploratory Analyses

Our pre-registration outlined several exploratory analyses that we planned to conduct beyond our primary hypotheses. Several of these exploratory analyses appear in our main text, particularly the replay analysis that decomposed performance improvements into model and prompting effects. The pre-registered exploratory analyses were described as follows:

> "We plan to investigate whether differences in prompt engineering ability across demographic and other observed variables will vary depending on the complexity of the task, e.g., the difficulty of the image participants are being asked to replicate. We anticipate power for this analysis will be very low, so we chose to label it as an exploratory analysis rather than a pre-registered hypothesis.
>
> We anticipate that we may conduct additional analysis of the prompts submitted by participants (and how these prompts evolve over the course of a session). Furthermore, we might explore the tips that participants provide after completing the task on how to prompt engineer effectively.
>
> We also may take original and revised prompts submitted to DALL-E 3 treatment arms and submit them to DALL-E 2 (and vice versa) to see how participants would have counterfactually performed under different treatment assignments than the one to which they were assigned."

As noted in the main text, the third exploratory analysis—submitting prompts from one model condition to another model—became central to our investigation of model versus prompting effects. This approach allowed us to isolate how much improvement came from the model's enhanced capabilities versus users adapting their prompting strategies to take advantage of those capabilities.

G.2 Deviations From Pre-registration

We report below all deviations from our pre-registered analysis plan. These deviations primarily resulted from statistical or methodological considerations that became apparent during data analysis, rather than from substantive changes to our research questions or hypotheses.

- **Statistical tests:** Our pre-registration specified t-tests and Mann-Whitney U tests for many hypotheses. However, this approach proved inappropriate for variables with multiple categories (which characterized most demographic traits). We therefore employed ANOVA and Kruskal-Wallis tests instead, which provide equivalent information for multi-category variables.

- **Z-score computation:** Our pre-registration was insufficiently precise about computing Z-score performance measures. As detailed in Section D.2, we calculated Z-scores within image-attempt pairs to adjust for variations across target images and attempts, except when comparing performance across attempts, where Z-scores were computed within target images only. The pre-registration had anticipated computing Z-scores only within target images for all analyses.

- **Additional demographic variables:** We inadvertently omitted Education and Generative AI outlook from our pre-registered list of demographics for measuring task performance heterogeneity. Given their theoretical importance, we included these variables in our analyses despite this accidental omission.

- **Additional prompt exclusion criteria:** Beyond our pre-registered exclusion criteria, we removed additional prompts that did not appear to constitute "good-faith efforts" based on their text content (see Section C.4 for details). Robustness checks confirmed that our results remain consistent when including these prompts.

- **T-tests vs. Z-tests:** For testing H5, we used t-tests rather than z-tests because they were easier to implement. As these tests are asymptotically equivalent, this change does not meaningfully affect our results.

- **Z-score outlier handling:** When testing H4, we observed that Z-scores did not follow a normal distribution and included extreme values (ranging from -21.9 to 7.1). To prevent these outliers from disproportionately influencing our linear models, we excluded observations with absolute Z-scores greater than 3 (2.65

- **GPT-4V rescaling omission:** Our pre-registration included a rescaled version of performance metrics based on GPT-4V prompt quality. However, this rescaling would have amounted to subtracting a constant from each unscaled score, and since all our analyses adjusted for target image (either through post-stratification or as a covariate), results would have been identical to those using unscaled measures. We therefore omitted these redundant robustness checks.

G.3 OSF Pre-registration

A complete copy of the pre-registration deposited on OSF on December 11, 2023 will be included beginning on the next page in the final manuscript, but has been removed here to maintain anonymity.