On the Effects of Complementary and Substitutive Use of Generative Artificial Intelligence in Innovation Processes

Inauguraldissertation

zur

Erlangung des Doktorgrades

der

Wirtschafts- und Sozialwissenschaftlichen Fakultät

der

Universität zu Köln

2025

vorgelegt

von

Matthias Lehmann, M.Sc.

aus

Würzburg

Referent: Prof. Dr. Fabian J. Sting

Koreferent: Prof. Dr. Andreas Fügener

Tag der Promotion:

Contents

1. Introduction

Artificial Intelligence (AI) is on the rise. The last decade has witnessed a profound paradigm shift driven by the rapid evolution of AI. The seminal success of AlexNet (Krizhevsky et al. 2012) heralded the modern era of deep learning, showcasing unprecedented generalization capabilities, while AlphaZero (Silver et al. 2017, 2018) demonstrated the emergence of autonomous, super-human agents that can transcend human understanding as best exemplified by the renowned "move 37"[1] against Lee Sedol. This trajectory culminated in the widespread adoption of *generative AI*, such as Large Language Models (LLMs) and image generation models, epitomized by systems like ChatGPT (OpenAI 2022). Modern frontier LLMs now act as conversational agents exhibiting broad human-level intelligence across domains such as coding, writing, and mathematics (e.g., Brown et al. 2020, Gemini Team et al. 2023, Dubey et al. 2024). This new class of AI represents a fundamental transformation in how we work, create, and innovate, capable of producing non-trivial outputs that can yield the perception of creativity (Boussioux et al. 2024, Chen and Chan 2024).

These advances are not only of technical interest but are already reshaping business processes at large (Dell'Acqua et al. 2023, Brynjolfsson et al. 2025). In particular, generative AI is transforming knowledge-intensive domains by augmenting or automating cognitive tasks that were previously considered uniquely human. Among these, innovation management, a domain concerned with the generation, development, and implementation of novel ideas, stands out as especially susceptible to disruption. Early research has identified AI's potential to overcome long-standing inefficiencies in innovation processes (Haefner et al. 2021), and the advent of generative AI amplifies this potential. The interactive nature of LLMs and their grounding in vast knowledge enable applications across the entire innovation lifecycle, from ideation to prototyping. Empirical evidence is beginning to substantiate this view, showing that AI boosts innovation both on an organizational and on a process-level. On an organizational level, AI adoption increases the innovation performance of firms at

[1] https://www.wired.com/2016/03/two-moves-alphago-lee-sedol-redefined-future/

early stages of development (Wang and Wu 2025) and as they scale (Wu et al. 2025). On a process-level, LLMs have been shown to outperform traditional human ideation methods (Meincke et al. 2024). Beyond idea generation, AI tools increasingly support the evaluation of ideas, surpassing human experts in selecting the most promising ideas (Bell et al. 2024). Finally, Gottweis et al. (2025) also demonstrate that AI systems can fully automate the entire ideation process of creating, refining, recombining, critically evaluating, and selecting ideas as well as learning from previous ideas in a closed loop, resulting in ideation so broad and deep that it can even result in novel scientific discoveries.

While the potential of AI in innovation management is readily apparent, the critical question for scholars and managers is: *How to design innovation processes to harness the potential of AI most effectively?*. Two primary avenues have emerged for integrating AI into workflows: *substitutive* (automation) and *complementary* (augmentation) use (Lichtenthaler 2018). Substitutive use involves replacing human labor with AI to fully automate process steps, while complementary use focuses on creating a symbiotic relationship where humans and AI collaborate, leveraging the distinct strengths of both. Research suggests that substitutive use boosts productivity broadly, yet that it simultaneously leads to a shift in required skills and roles (Acemoglu and Restrepo 2018, Law and Shen 2025). Nonetheless, prior work increasingly points to the superiority of complementary over substitutive AI use to benefit from both human and AI capabilities and create synergies (Raisch and Krakowski 2021, Fügener et al. 2022).

Yet, we still lack a fine-grained understanding of how substitution and complementarity unfold at the micro-level of innovation processes, and what consequences each mode of use entails. Which subprocess steps should be automated or augmented with AI – or left to humans to focus their engagement on? This dissertation addresses this gap by investigating how the substitutive versus complementary use of generative AI shapes human behaviors and outcomes in two fundamental innovation processes: *ideation* and *learning*. These processes form the building blocks of innovation. Ideation constitutes the entry point of innovation as

the ability to generate new ideas is a prerequisite for innovation. Learning, in turn, sustains innovation over time by enabling individuals and organizations to continuously absorb, adapt, and recombine knowledge from experiences in order to improve. By studying both, this work provides generalizable insights applicable across the entire innovation lifecycle. Across two experimental studies, we systematically study the mode of AI use to uncover its effects on process engagement, performance outcomes, and implications for innovation.

Chapter 2 presents the first study, which is based on the paper *"AI Meets the Classroom: When Do Large Language Models Harm Learning?"*. In a controlled laboratory setting, we manipulate access to an LLM for students learning to code and observe the learning process, their modes of AI use, and learning outcomes in order to answer the question whether AI harms or helps learning. On aggregate, we find neither, as we uncover two counteracting effects. On the one hand, students that use LLMs as personal tutors (complementary use) by asking for explanations improve their understanding of the course contents, albeit at the cost of progressing slower. On the other hand, students who use LLMs to replace effort by generating solutions to practice exercises (substitutive use) can cover more topics superficially but exhibit significantly less understanding of each topic. Interestingly, despite the readily apparent adverse effects of substitutive use, the majority of students opts for using LLMs as shortcuts this way. Finally, we also find that AI use – regardless of the use mode – results in an overestimation of the own abilities; students think they learned more than they actually did. Overall, this study demonstrates how substitutive use can inhibit the very learning processes upon which sustained innovation depends, whereas strategic complementary use can amplify it.

Chapter 3 presents the second study, which is based on the paper *"Don't Stop Sketching! Design Ideation Processes in the Age of Generative AI"*. This study examines complementary and substitutive use of generative AI in design ideation. We conduct an experiment where we manipulate whether subjects retain a traditional planning step – sketching – prior to AI-empowered ideation in a logo design competition. Thus, we explicitly control whether

the AI merely substitutes sketching, or complements it, in order to determine how to optimally structure AI-augmented ideation processes. We find that sketching prior to AI use significantly reduces design idea variance, thereby undermining a key benefit of AI use as it promises to enable rapid exploration. Contrary to classical innovation literature however, which equates variance with higher-quality ideas (Girotra et al. 2010), sketching improves the average and peak design idea quality despite reducing variance for both design novices and experts. Exploratory analyses provide potential explanations for the observed effects, hinting that AI may exacerbate the human tendency to break off local search too early and instead entices off-task exploration, such that preparatory human engagement, e.g., via sketching, can anchor attention and therefore enhance focus. This study thus illuminates the paradoxical role of AI in ideation: while it vastly expands the solution space, its benefits depend critically on how humans are prepared and constrained in interacting with it.

Across both studies, we demonstrate that not merely AI adoption but the *mode* of AI integration including the specific interleaving of human, AI-augmented and automated process steps critically determines innovation outcomes. We reveal pervasive adverse effects of substitutive AI use in innovation processes. When users substitute the wrong process steps with AI, they reduce their own mental efforts and are less engaged in the overall process. This lack of engagement has long-term consequences, as it can leads to lower understanding and diminishing long-term learning. In ideation processes, this also induces immediately worse results by weakening cognitive focus on the present task. Nonetheless, we consistently observe a human tendency towards substitutive use, even when the adverse ramifications appear obvious. Conversely, when users do not take such shortcuts and instead complement or even reinforce their engagement with AI, they derive significant benefits from AI use. In ideation processes, using a combination of traditional planning steps and AI-empowered idea generation ensures a stronger task focus which in turn yields superior results. In learning processes, using AI as a tutor to engage even more deeply in the learning materials, rather than shortcutting through them, boosts students' understanding. Lastly, we

also observe paradoxical tensions between objectively measured and perceived outcomes of AI use. In the learning process, AI use boosts the students' perceived learning significantly beyond their actual progress. In the ideation process, retaining the manual sketching step does not increase self-perceived idea quality despite measurable benefits and users do not realize their reduced idea variance. Thus, we reveal across both studies that many effects of AI use (complementary and substitutive) appear to be subconscious. This also lends further importance to the problem of innovation managers having to carefully design AI-augmented processes as, when left alone, users may choose suboptimal use modes.

Practically, our work yields actionable design principles for innovation managers regarding the design of AI-supported innovation processes. Our studies show that user engagement is critical in AI-supported innovation processes and complementary use helps preserving or even reinforcing this engagement, while misplaced substitutive use can systematically erode it. Effective AI integration therefore requires safeguarding human engagement in critical process steps, whether through retaining traditional practices (such as sketching or manual exercises), designing AI systems to promote engagement rather than shortcuts (e.g., via system prompts), or implementing nudges and constraints that reduce the temptation of substitution. In sum, we provide both a cautionary perspective on the risks of over-automation and a constructive pathway for designing innovation processes that realize the full potential of AI. Ultimately, a new philosophy of human-AI collaboration is needed, one that recognizes the nuanced effects of different AI interaction modes on short- and long-term behavior at the micro-level as we elucidated.

Author Statement

I am the first author of both presented papers. Paper 1 (Chapter 2) was written with co-authors being Fabian Sting and Philipp Cornelius. A first version of the paper was accepted and presented at the Workshop for Empirical Research in Operations Management at the Wharton School, University of Pennsylvania, in October 2024. The paper as presented here was submitted to *Management Science* in March 2025 where it received a *reject-and-resubmit*

decision. I took on the lead role in this project by conceptualizing, designing, conducting the laboratory experiments and performing the data analysis, as well as writing the majority of the manuscript. Philipp Cornelius provided the initial research direction of investigating relation between AI and student learning, and conducted the field study (study 1 in the paper). I thank my co-authors for providing guidance, inputs and feedback, for both the experiment design and the writing.

Paper 2 (Chapter 3) is joint work with co-authors Fabian Sting and Henrik Franke. The paper is in the final preparatory steps of submission to *Management Science* with the intent to submit in October 2025. I originated the research idea, designed, conducted and analyzed the experiments and took the lead role in writing the manuscript. I thank my co-authors for the valuable discussions and inputs towards the research and experiment design and their feedback and edits on the manuscript.

AI Meets the Classroom: When Do Large Language Models Harm Learning?

Matthias Lehmann
University of Cologne, matthias.lehmann@wiso.uni-koeln.de

Philipp B. Cornelius
Rotterdam School of Management, Erasmus University, cornelius@rsm.nl

Fabian J. Sting
University of Cologne, Rotterdam School of Management, Erasmus University, sting@wiso.uni-koeln.de

Abstract. The effect of large language models (LLMs) in education is debated: Previous research shows that LLMs can help as well as hurt learning. In two pre-registered and incentivized laboratory experiments, we find no effect of LLMs on overall learning outcomes. In exploratory analyses and a field study, we provide evidence that the effect of LLMs on learning outcomes depends on usage behavior. Students who substitute some of their learning activities with LLMs (e.g., by generating solutions to exercises) increase the volume of topics they can learn about but decrease their understanding of each topic. Students who complement their learning activities with LLMs (e.g., by asking for explanations) do not increase topic volume but do increase their understanding. We also observe that LLMs widen the gap between students with low and high prior knowledge. While LLMs show great potential to improve learning, their use must be tailored to the educational context and students' needs.

Key words: generative artificial intelligence, large language models, education, learning, technology management

2.1. Introduction

Large language models (LLMs) have become pervasive in education. They can reproduce knowledge on a vast range of subjects and perform well above passing grades in university-level exams (e.g., Drori et al. 2022, Terwiesch 2023). As a result, the majority of students

now use LLMs regularly (Freeman 2025). As business adoption of LLMs spreads throughout the economy, emerging research points to significant productivity gains (Brynjolfsson et al. 2023, Dell'Acqua et al. 2023, Noy and Zhang 2023). Yet, despite similar if not greater adoption of LLMs in education, their impact on learning outcomes remains contentious. Whereas some studies show that LLMs help learning (Nie et al. 2024, Kestin et al. 2024), others find the opposite (Bastani et al. 2024).

To make a step toward reconciling these conflicting findings on whether LLMs help or harm learning, we report on a field study and two pre-registered and incentivized laboratory experiments. In the two experiments, we find no significant effect of LLM access on overall learning outcomes. In subsequent exploratory analyses, we provide evidence that the effect depends on how students use LLMs and how learning outcomes are measured. Students who use LLMs to substitute some of their learning activities (e.g., practice exercises) move faster through the instructional material and thus increase the volume of topics they can cover. However, this increase in speed comes at the cost of a shallower understanding of each topic. In contrast, students who use LLMs to complement their learning activities (e.g., using LLMs as tutors) improve their topic understanding. In the field study, in which we only observe substitutive use, we leverage exogenous variation in LLM availability to find a significant long-term decline in overall learning outcomes. In addition, we show that how LLMs affect learning is contingent on students' prior knowledge: LLMs support the learning of students with more prior knowledge but harm the learning of students with less prior knowledge. Unlike in workplaces (Brynjolfsson et al. 2023, Dell'Acqua et al. 2023), LLMs appear to increase inequality in education.

We make several empirical and theoretical contributions to the literature on education operations (e.g., Smilowitz and Keppler 2020, Yoo and Zhan 2023, Keppler et al. 2022, Keppler 2024) and education technology (e.g., Zhang et al. 2017, Corsten and Skousen 2023, Bray 2024). First, we study how unrestricted LLMs affect learning outcomes. Previous studies restricted LLMs to certain use cases, in particular by limiting substitutive use (Nie

et al. 2024, Kestin et al. 2024). While such customized LLMs offer great opportunities in supervised educational settings, a large part of learning occurs unsupervised, and during such unsupervised learning most students have access to free and unrestricted LLMs, such as ChatGPT. Second, we are among the first to investigate the effect of LLMs on learning in a laboratory without potentially confounding effects from undocumented LLM use (e.g., at home), classrooms, teachers, and peers. We hope that our simple experimental setup provides a basis for more laboratory studies on the microfoundations of learning with LLMs. Third, we add to the emerging literature on how technology use influences learning (e.g., Teodoridis 2018, Wuttke et al. 2022). Our results suggest that students prefer to use LLMs to substitute rather than complement their own learning activities, which can harm learning outcomes. Despite an increasing focus on student behavior (Lavecchia et al. 2016, Damgaard and Nielsen 2020), issues such as user self-control have been largely absent from the evaluation of technology in education (Bulman and Fairlie 2016, Chatterji 2018). Our study also offers insights to broader innovation scholarship about the impact that technology use has on learning and knowledge acquisition (e.g., Shane and Ulrich 2004, Loch 2017).

The rest of the paper is organized as follows. In Section 2.2, we review the literature and in Section 2.3 we develop hypotheses. We provide an overview of all studies in Section 2.4 and present the field evidence in Section 2.5. Section 2.6 presents the first experiment and Section 2.7 our revised hypotheses based on that experiment. Section 2.8 presents the second experiment and Section 2.9 our exploratory analyses. We discuss our findings in Section 2.10.

2.2. Related Literature

Evidence on how technology affects learning outcomes remains mixed (for reviews, see Bulman and Fairlie 2016, Chatterji 2018). On the one hand, self-paced and individualized computer-aided instruction promises to extend the benefits of small class sizes and one-on-one interaction with instructors to more students. In addition, technology also supports traditional learning activities, for example by allowing instructors to closely track students'

learning progress or by making instruction more engaging. On the other hand, technology may be distracting, because it draws students' attention away from the subject matter and toward understanding the technology itself. Technology that reduces job demands in work settings is sometimes found to reduce learning (Wang et al. 2020) and education technology that similarly lowers effort may thus also reduce learning. Moreover, investment in technology may take away from investment in traditional learning activities. Despite the importance of technology in education, rigorous evaluations of its efficacy remain scarce.

Large language models (LLMs) have demonstrated astounding capabilities in educational assessments (e.g., Drori et al. 2022, Terwiesch 2023). Teachers who collaboratively "think" with LLMs about teaching activities report being more productive when preparing classes, while teachers who let LLMs generate teaching material report no change in their productivity (Keppler et al. 2024). For students, however, the emerging research has found mixed results. Students perceive LLM tutors to be helpful (Liu et al. 2024b), but prefer human explanations over LLM explanations (Pardos and Bhandari 2023, Prihar et al. 2023). Still, LLM explanations improve students' learning outcomes (Kumar et al. 2023, Pardos and Bhandari 2023).

In the above studies, students were shown LLM-generated content, but they could not interact with an LLM themselves. Bastani et al. (2024) run a field experiment in math classes at a high school in which students interact with an LLM. They find that the LLM improves student performance on practice problems but reduces performance in the final exam when students no longer have access to it. In addition, they show that this harmful effect on learning can be mitigated by modifying the LLM so that it produces fewer errors and does not directly provide solutions to students. Nie et al. (2024) conduct a similar field experiment in an online programming course. They observe very limited usage of the LLM and document that it reduces engagement with the course materials and the likelihood of completing the course. Still, the small group of students who do use the LLM performs better on the final exam. Kestin et al. (2024) conduct a field experiment in university-level

physics classes, finding that students achieve higher learning outcomes when learning with an LLM-based personal tutor than when attending traditional active learning classes. Nie et al. and Kestin et al. also implement safeguards against students asking an LLM for solutions to practice problems. Given these contrasting findings, we extend the nascent literature on LLMs in education by shedding light on the circumstances under which LLMs help or hurt learning.

2.3. Theory

We conceptualize learning outcomes as the value added by education, which is the difference in students' knowledge before and after education (Hanushek 2020). In a simple framework of a learning process, we decompose learning outcomes into topic volume and topic understanding. Students learn about a set of topics, such as the topics taught in a course. To learn about each topic, students engage in a variety of learning activities, for example reading textual explanations or solving practice problems. Students learn as they cover more topics and understand each topic better. Thus, students' learning outcomes are a function of topic volume and understanding.[2] Based on this framework, we develop two competing and pre-registered[3] hypotheses of the effect of LLMs on learning outcomes.

2.3.1. LLMs Improve Learning Outcomes

Personal tutoring improves student learning outcomes (Cohen et al. 1982, Bowman-Perrott et al. 2013). LLMs promise to provide personal tutoring at scale, to every student, at all hours of the day. Prior studies have found that even previous rule-based, non-LLM chatbots improve learning (Abbasi and Kazi 2014, Chen et al. 2020, Yin et al. 2021, Chang et al.

[2] The distinction between topic volume and topic understanding bears some resemblance to the distinction between education quantity and quality in economics (Hoekstra 2020), but it is not exactly the same. Education quantity usually refers to the number of years people spend in education, while education quality refers to the efficacy of educational institutions in imparting knowledge. In our case, the distinction is similarly about the volume of knowledge that students are exposed to and the efficacy of internalizing that knowledge, but the level of analysis is at the individual learning process rather than at the institution.

[3] https://aspredicted.org/2LM_DP3

2022, Essel et al. 2022, Ait Baha et al. 2023). LLMs improve on rule-based chatbots in several ways.

First, the large-scale pretraining of LLMs equips them with vast knowledge across a wide range of domains (Brown et al. 2020, Touvron et al. 2023). LLMs also exhibit strong coding abilities, which includes problem-solving and the ability to reason about self or user-generated code (Touvron et al. 2023, Gemini Team et al. 2023, Liu et al. 2024a). In this domain, LLMs have surpassed the average performance of human professionals (AlphaCode Team 2023).

Second, modern LLMs are fine-tuned to act as helpful assistants (OpenAI 2022, Touvron et al. 2023). This allows students to interact with them flexibly and iteratively like they would with a human tutor. LLMs understand user queries even if they do not use the correct technical jargon and tailor their responses to the level of knowledge that the user exhibits in their message. The versatility of LLMs means that they can give hints on problems, provide examples, explain concepts, and write code.

Third, LLMs can act as a thought partner for students and enhance understanding by providing detailed explanations and step-by-step solutions (Mollick and Mollick 2022, Kasneci et al. 2023, Meyer et al. 2023, Extance 2023). Several studies have found that LLM-generated explanations improve learning in mathematics (Pardos and Bhandari 2023, Prihar et al. 2023, Kumar et al. 2023) and physics (Kestin et al. 2024). Similarly, learning improved in programming courses in which students had access to LLMs (Liu et al. 2024b, Nie et al. 2024).

Interacting with LLMs while learning may affect both components of learning: volume and understanding. LLMs may allow students to cover more topics, because students can quickly ask questions if they do not understand something. For the same reason, LLMs may also improve students' understanding of each topic. For instance, an LLM may explain a topic that a student has not understood before. Thus, LLMs may improve learning outcomes through both topic volume and topic understanding.

Hypothesis 1 *Having access to LLMs when learning to code increases the learning outcome of students.*

2.3.2. LLMs Harm Learning Outcomes

Learning and application often go hand in hand: To fully understand a topic, students need to apply their knowledge to a related problem and learn from their experience of solving that problem. Such learning-by-doing is especially important when learning to code (Narayanan et al. 2009). As we mentioned above, LLMs can generate working code and students may be tempted to use this functionality to solve practice exercises more quickly. If students use an LLM to solve exercises (fully or partially), they reduce their own mental effort and do not engage in the problem solving necessary to internalize and understand a topic. Hence, LLM usage may inhibit learning-by-doing and thus decrease learning outcomes.

This argument extends beyond learning-by-doing, because LLMs may reduce time and effort spent on learning more broadly. Successful learning requires investments of time and effort and students are easily lured into reducing their investments if given the opportunity (Bishop 2006, Lavecchia et al. 2016). For example, by relying on LLMs to talk about and explain concepts, students may inadvertently reduce their own mental effort during the acquisition of new knowledge. Previous technologies from computers, to virtual reality, to pre-LLM artificial intelligence have similarly reduced the mental effort required of their users and thereby decreased their learning (e.g., Mueller and Oppenheimer 2014, Wuttke et al. 2022, Dell'Acqua 2022, Miola et al. 2024; for a review, see Wang et al. 2020).

Aside from reducing mental effort, LLMs may also harm learning by producing falsehoods or "hallucinations" (Kasneci et al. 2023, Meyer et al. 2023, Extance 2023). While experts may be able to spot such errors, students are easily deceived because they do not know the topic yet. Once put on the wrong path, students may fundamentally misunderstand a new topic. In addition, LLMs may engage students in tangential or peripheral conversations that reduce the volume of topics students have time to cover. Thus, LLMs may harm learning outcomes and Bastani et al. (2024) provide first empirical evidence that high school students learning mathematics with the help of LLMs perform worse in exams.

Hypothesis 2 *Having access to LLMs when learning to code decreases the learning outcome of students.*

2.4. Overview of Studies

We test the hypothesized effects of LLMs on learning in three studies. We combine field data (Study 1) and two lab experiments (Studies 2 and 3) to provide evidence of both internal and external validity (e.g. Lee et al. 2021, Buell 2021, Wiltermuth et al. 2023). In Study 1, we show how the use of LLMs during weekly assignments affects learning in two graduate programming courses. In Studies 2 and 3, we replicate the field setting in two incentivized and pre-registered laboratory experiments. In Study 2, we manipulate LLM access and compare learning outcomes between treated (with LLM) and controls (without LLM). Study 3 is an exact replication of Study 2 except for the availability of copy and paste for all participants (treated and controls). In Study 2, participants could not copy and paste text from or to the LLM and—as we discuss in Section 2.7—this may have affected how the LLM was used. In the next sections, we describe each study and report its results.

2.5. Study 1: Field Evidence

We begin with a field study of two graduate programming courses delivered at roughly the same time in the Spring of 2023 at a public Dutch university. The courses started with an introduction to Python programming (e.g., variables, data types) and finished with simple machine learning models (e.g., random forests). The two courses were delivered in different programs, one in information systems (IS) and the other in business analytics (BA). The courses were identical in terms of content and assessment except that the BA course had one less lecture on Python because students in that program had already received an introduction to Python in another course. Hence, the IS course comprised five lectures over five weeks and the BA course comprised four lectures over four weeks. 56 students took the IS course, and 57 students took the BA course; no student took both courses.

Each lecture was accompanied by a coding homework assignment on an online coding platform. Students had three days to complete each assignment and their assignment grade

counted towards their course grade. Each assignment consisted of a different number of coding questions and in total there were 65 questions across five assignments in the IS course and 55 questions across four assignments in the BA course. (There is one less assignment in the BA course because there is also one less lecture.) On the online coding platform, students could submit code for each question as often as they liked and for each submission they immediately saw their grade and any potential error message that their code generated. Only their last code submission was graded. While the lecture imparted programming theory, students had to work through the assignments to fully internalize the content. Thus, students were learning by doing, that is by working through the assignments and receiving feedback in form of their grade and potential error messages.

The data has a panel structure and the unit of analysis is student–questions. There are 56 $\times$ 65 = 3,640 student–question observations in the IS course and 57 $\times$ 55 = 3,135 student–question observations in the BA course, yielding 6,775 student–question observations in total.

2.5.1. Variables

2.5.1.1. Dependent Variable Students' grade for each question is the dependent variable. The grade is normalized such that for student i and question q, $\text{Grade}_{iq} \in [0, 1]$. At the focal institution, the average normalized course grade is between 0.7 and 0.8. An average (across all courses) of at least 0.825 is considered a distinction and an average of at least 0.9 is considered an exceptional distinction. A grade below 0.55 is considered a fail. Most students fall between 0.6 and 0.85.

2.5.1.2. Explanatory Variable We measure students' use of generative AI as the similarity between students' final code submission and ChatGPT-generated code for the same question. ChatGPT was the main LLM chatbot available during the courses. We collected ChatGPT-generated code by submitting the same question descriptions that students saw to the OpenAI API. We exactly replicate the ChatGPT environment that was available to students at the time of the courses (`gpt-3.5-turbo-0613`). Because ChatGPT generates solutions

stochastically, which can therefore differ across queries, and we do not know which solution a student received, we query ChatGPT 50 times per question and take the maximum similarity across the 50 ChatGPT solutions:

$$\begin{aligned}&\text{ChatGPT Similarity}_{iq} = \\ &\quad \max\left\{\text{sim}\left(\text{Code}_{iq}, \text{Code}_{jq}^{\text{LLM}}\right) \,\middle|\, j = 1, \ldots, 50\right\},\end{aligned}$$

for student i and question q, where Code_{iq} is the final student code, $\text{Code}_{jq}^{\text{LLM}}$ is one of the 50 ChatGPT generated solutions, and $\text{sim}(\cdot, \cdot) \in [0, 1]$ is the normalized Damerau–Levenshtein similarity (available from RapidFuzz). In addition to *ChatGPT Similarity* for a focal question, we compute the *Cumulative ChatGPT Similarity* over all questions answered before the focal question to estimate the effect of past LLM usage on learning.

Since we cannot directly observe whether a student used ChatGPT, *ChatGPT Similarity* is a proxy for this behavior: A higher value maps into a higher probability that ChatGPT was used. This measurement of treatment is similar to intention-to-treat analyses in experimental designs, in which treatment assignment as the main explanatory variable also corresponds to a higher probability of actual treatment (e.g., Bastani et al. 2024).

We exclude 181 observations of questions that students did not attempt to answer. For such unanswered questions, we cannot measure *ChatGPT Similarity*, because no student code was submitted. Unanswered questions are conceptually different from answered questions with a true *ChatGPT Similarity* of 0, so imputing a *ChatGPT Similarity* of 0 for unanswered questions would introduce a measurement bias.

2.5.1.3. Control Variables We control for the following potential confounders:

Plagiarism. Because the homework assignments are not invigilated, students may copy solutions from one another. If a student whose solution was plagiarized used ChatGPT, the plagiarizing student's *ChatGPT Similarity* will also be higher even if they themselves did

Table 1 Study 1: Descriptive Statistics.

	Mean	Min.	Max.	S.D.
Grade	0.95	0.00	1.00	0.19
ChatGPT Similarity	0.42	0.00	0.99	0.14
Cum. ChatGPT Similarity	12.19	0.00	37.87	7.93
Plagiarism	0.71	0.00	1.00	0.19
Cum. Plagiarism	22.15	0.00	54.58	12.89
Questions Answered	29.07	0.00	64.00	17.45
Previous Grade	0.96	0.00	1.00	0.15
Time Taken	47.08	0.10	1,595.7	103.02
Cum. Time Taken	795.72	0.00	7,153.07	900.25
ChatGPT Outages	882.21	0.00	58,500.00	5,735.40
Cum. ChatGPT Outages	14,748.96	0.00	827,092.00	75318.07

Note. N = 6,594.

not use ChatGPT. For student i and question q we measure Plagiarism_{iq} as the maximum normalized Damerau–Levenshtein similarity with any other student solution for the same question in the same course. We also compute *Cumulative Plagiarism* over all questions answered before the focal question.

Questions Answered. Students' performance and behavior may change as they solve more questions. For example, they may become more confident and use ChatGPT less. We thus control for the cumulative number of questions students have answered before the focal question.

Previous Grade. Students' performance and behavior may change depending on their performance in previous questions. For example, students who are doing well may become more confident and use less ChatGPT. We thus control for the average grade across all previously answered questions.

2.5.2. Identification

Our baseline model is a two-way fixed effects (FE) model. For student i and question q,

$$\begin{aligned}\text{Grade}_{iq} = \beta_1 &\text{ChatGPT Similarity}_{iq} \\ &+ \beta_2 \text{Cum. ChatGPT Similarity}_{iq} \\ &+ \boldsymbol{X}'_{\boldsymbol{iq}}\boldsymbol{\beta_3} + c_i + c_q + \varepsilon_{iq},\end{aligned} \tag{1}$$

where $\boldsymbol{X}_{iq}$ is a vector of control variables, c_i are student fixed effects, and c_q are question fixed effects. This setup controls for constant additive student and question confounders as well as for dynamic linear confounders in the form of students' prior performance. The parameter β_2 captures the effect of past LLM use on the grade of the current question. It estimates how past LLM use has affected students' overall learning (both in terms of topic volume and understanding). Since we can only observe whether students submitted LLM-generated solutions and not whether they used the LLM in any other way that supported their learning, β_2 only estimates the effect of students substituting some of their learning activities with the LLM.

In addition and to rule out further confounding factors (e.g., measurement error), we estimate the FE model in Equation 1 with instrumental variables for the two similarity measures. As instruments, we use ChatGPT service interruptions as reported on the OpenAI Status Blog[4] (OpenAI is the vendor of ChatGPT). We include all interruptions that mention a ChatGPT outage in the incident report or as an affected service. We match each interruption to students who were solving a question at the same time, that is to all student–questions whose first submission occurred before the end of an interruption and whose last submission occurred after the start of an interruption. *ChatGPT Outages* is then the amount of time (in seconds) that an interruption overlaps with the time between the first and last submission. *Cum. ChatGPT Outages* is the corresponding cumulative measure over all previous questions. For instance, if a student worked on a question between 9am and 11am, and there was an

[4] https://status.openai.com

outage from 9.30am until 10am, then for that student–question we calculate 30 minutes of outage. As students who take longer to solve a question may be exposed to more outages, we control for the time between first and last submission to a question as the *(Cumulative) Time Taken* (in minutes). *ChatGPT Outages* exogenously vary the availability of ChatGPT and thus allows us to estimate quasi-experimental effect sizes.

We estimate the fixed effects 2SLS (FE2SLS) estimator (Wooldridge 2010), implemented as `ivreghdfe` in Stata (Correia 2017). In the first stages, we estimate

$$\begin{aligned}&\text{ChatGPT Similarity}_{iq}\\&\quad=\gamma_1\text{ChatGPT Outages}_{iq}\\&\qquad+\gamma_2\text{Cum. ChatGPT Outages}_{iq}\\&\qquad+\boldsymbol{X_{iq}'\gamma_3}+c_i+c_q+u_{iq}\end{aligned}\tag{2}$$

and

$$\begin{aligned}&\text{Cum. ChatGPT Similarity}_{iq}\\&\quad=\delta_1\text{ChatGPT Outages}_{iq}\\&\qquad+\delta_2\text{Cum. ChatGPT Outages}_{iq}\\&\qquad+\boldsymbol{X_{iq}'\delta_3}+c_i+c_q+v_{iq}.\end{aligned}\tag{3}$$

In the second stage, we estimate

Table 2 Study 1: Two-way Fixed Effects (FE) Results.

	Grade
ChatGPT Similarity	0.18*** (0.05)
Cum. ChatGPT Similarity	-0.02*** (0.00)
Plagiarism	0.15*** (0.03)
Cum. Plagiarism	0.00 (0.00)
Questions Answered	0.00 (0.00)
Previous Grade	0.04 (0.04)
Time Taken	0.00* (0.00)
Cum. Time Taken	-0.00 (0.00)
Constant	0.79*** (0.05)
F-statistic	7.64***
R^2	0.29
Adjusted R^2	0.27
Students	113
Observations	6,594

Notes. Student-clustered standard errors in parentheses. Asterisks indicate significance: *p <0.10; **p <0.05; ***p <0.01.

$$\begin{aligned}\text{Grade}_{iq} &= \beta_1 \widehat{\text{ChatGPT Similarity}}_{iq} \\ &\quad + \beta_2 \text{Cum. } \widehat{\text{ChatGPT Similarity}}_{iq} \\ &\quad + \boldsymbol{X'_{iq}\beta_3} + c_i + c_q + \varepsilon_{iq}.\end{aligned} \tag{4}$$

As before, β_2 is the parameter of interest and estimates the effect of substitutive LLM use on overall learning outcomes.

2.5.3. Results

Table 1 presents descriptive statistics of the panel data. The reported average grade is larger than the actual course grade, because we do not apply question weights in the statistical analyses. Table 2 shows the results of the two-way fixed effects model in Equation 1 (FE). The effect of *ChatGPT Similarity* on the *Grade* of the current question is positive and significant ($p < 0.001$), while the effect of *Cum. ChatGPT Similarity* is negative and significant (p

Table 3 Study 1: Instrumental Variable Fixed Effects (FE2SLS) Results.

	First Stages		Second Stage
	ChatGPT Similarity	*Cum. ChatGPT Similarity*	*Grade*
ChatGPT Outages	-0.00*** (0.00)	-0.00 (0.00)	
Cum. ChatGPT Outages	-0.00 (0.00)	-0.00*** (0.00)	
ChatGPT Similarity			-0.35 (0.56)
Cum. ChatGPT Similarity			-0.06*** (0.02)
Plagiarism	0.16*** (0.02)	0.64*** (0.24)	0.26** (0.10)
Cum. Plagiarism	0.00** (0.00)	0.13** (0.06)	0.01** (0.01)
Questions Answered	-0.00 (0.00)	0.40*** (0.06)	0.02*** (0.01)
Previous Grade	0.02 (0.02)	-0.27 (0.35)	0.04 (0.04)
Time Taken	0.00 (0.00)	-0.00 (0.00)	0.00 (0.00)
Cum. Time Taken	-0.00* (0.00)	-0.00*** (0.00)	-0.00** (0.00)
F-statistic	13.91***	5.60***	4.74***
Students	113	113	113
Observations	6,594	6,594	6,594

Notes. Student-clustered standard errors in parentheses. The constant term is partialled out. Asterisks indicate significance: *p <0.10; **p <0.05; ***p <0.01.

< 0.001). If students submit code that is identical to a ChatGPT-generated solution, their grade for the current question increases by 0.18, but their grade on all subsequent questions decreases by 0.02.

Table 3 shows the results of the instrumental variable fixed effects model in Equations 2–4 (FE2SLS). In the first stages, both instruments significantly affect the instrumented variables in the expected directions: *ChatGPT Outages* reduces *ChatGPT Similarity* (p = 0.005) and *Cum. ChatGPT Outages* reduces *Cum. ChatGPT Similarity* (p = 0.002). In the second stage, the effect of *ChatGPT Similarity* on the *Grade* of the current question is not significant anymore (p = 0.334), while the effect of *Cum. ChatGPT Similarity* remains negative and significant (p = 0.002). If students submit code that is identical to a ChatGPT-generated solution, their grade for the current question does not change, but their grade on all subsequent questions decreases by 0.06.

Across the FE and FE2SLS models we find evidence in support of a negative effect of LLMs on learning outcomes. The more students substitute for their own learning by using LLMs to generate solutions, the less they learn and the worse they perform on subsequent

questions. The benefit of using an LLM to get a better grade on the current question is only significant in the FE model, but not in the FE2SLS model. One explanation may be the correlation between high *ChatGPT Similarity* and a correct solution, such that conditional on student ability (c_i) and prior performance (*Previous Grade*), high *ChatGPT Similarity* indicates a chance correct solution rather than LLM use. This is a form of measurement error and by using an instrumental variable, we remove the correlation between *ChatGPT Similarity* and a correct solution and the estimate hence turns not significant.

2.6. Study 2: Laboratory Experiment

In Study 1, we find a negative effect of substitutive LLM use on learning in the field. In Study 2, we further investigate the impact of LLMs on learning in an incentivized and pre-registered[5] laboratory experiment. We closely replicate the field study context by teaching experimental subjects Python programming in a manner similar to typical online courses (e.g., Udemy, Coursera). We manipulate the availability of an LLM and compare learning outcomes between subjects. The controlled setting in a laboratory allows us to ensure that subjects only use the provided materials for learning (e.g., no smartphones).

2.6.1. Experimental Design

The main task of the subjects is to learn to code in Python. We choose Python as the programming language since it is among the most popular programming languages (Vailshery 2024) and is particularly beginner-friendly compared to other languages (Lemonaki 2022).

The experiment consists of three distinct phases: pre-test, learning phase, and post-test. During the learning phase, which lasts for 45 minutes, subjects follow an introduction into basic Python programming, covering topics such as the print function, strings, and if-statements. The learning phase replicates the learning-by-doing approach of the assignments in Study 1. The learning phase is broken down into 24 parts. Each part comprises an explanation of a concept (e.g., if-statements), an example of how to implement it in Python,

[5] https://aspredicted.org/2LM_DP3

and a question in the form of a coding exercise for subjects to practice. The variable *Learning Phase* counts the number of these practice questions that subjects answered correctly. Referring to our learning framework from Section 2.3, it measures the volume of topics that subjects were able to cover during the learning phase. To remove time effects, participants cannot continue to the post-test before the 45 minutes have elapsed.

The pre-test and post-test assess subjects' coding abilities before and after the learning phase. Both tests contain 20 programming questions and subjects have 20 minutes to solve them. The questions resemble those of the learning phase and cover the same topics. For each question in the pre-test, there exists a similar question in the post-test, which requires knowledge of the same topic, such that we can directly compare the results of the two tests. The pre-test allows us to control for random differences in initial coding abilities between the control and treatment conditions and estimate learning in terms of "value added" (Hanushek 2020). The variables *Pre-test* and *Post-test* count the number of questions that subjects answered correctly in the respective tests. Regarding learning outcomes, *Post-test* measures students overall learning (i.e., through both topic volume and understanding).

During each of the three phases, subjects have access to a fully functional code editor and are able to submit code as often as they like. When submitting code, subjects immediately see their code's output and any potential error message. If subjects solve a question correctly (in the learning phase or in the tests), they see a short message confirming their result. Within each phase, subjects can move freely between the different questions. We describe the user interface in detail in Appendix A.

Before the pre-test, we survey participants' demographics, studiousness, prior coding experience, and LLM experience (we describe all variables in Table 26 in Appendix B). After the post-test, we ask subjects to rate their perceived learning progress on a five-point Likert scale and to describe their learning experience in an open text field. All sessions took place in June 2024. The experiment was designed using oTree (Chen et al. 2016).

2.6.2. Treatment Condition

We manipulate LLM availability by randomly assigning subjects to one of two conditions. In the control condition, subjects work through the experiment exactly as described above without access to an LLM. In the treatment condition, subjects have access to an LLM during the learning phase only (not in the pre or post-test). The LLM is available in a chat window next to the question description (see Figure 6 in Appendix A). We use OpenAI's ChatGPT (`gpt-3.5-turbo-0125`; Brown et al. 2020, OpenAI 2022) as the LLM, which we query through an API. This model is trained to act as a helpful assistant and has sophisticated coding abilities (Liu et al. 2024a). ChatGPT can solve all programming questions used in the experiment correctly when prompted with their descriptions. Subjects in the treatment condition can chat with the LLM as often as they like and with arbitrary message contents. They can reset the conversation with a button.

2.6.3. Subjects

Subjects for the experiment were recruited from the laboratory participant pool at a public German university. The experiment was approved by the university's ethics committee.[6] All participants are enrolled students and had no knowledge of the experimental content beforehand. Subjects were incentivized with a fixed compensation of 10 euros and a performance-based compensation depending on the number of solved questions in the post-test, where they received 1 euro per solved question. The incentive structure is in accordance with the laboratory's minimum wage requirements.

We exclude participants based on two pre-registered filters. We exclude those who did not complete the entire experiment. In addition, we specifically designed the final question of the post-test such that is not solvable solely based on the knowledge acquired in the learning phase as it requires more advanced programming concepts. Thus, we use this question to filter out subjects with significant prior experience in Python as these are not the target group of our experiment.

[6] Ethics Committee of the Faculty of Management, Economics and Social Sciences (ERC-FMES) at the University of Cologne. Reference number: 240020ML.

Table 4 Study 2: Correctly Answered Questions per Experimental Phase.

	Pre-test	*Learning Phase*	*Post-test*	*Post-test − Pre-test*
Control Condition	3.1 (3.0)	15.1 (3.8)	7.9 (4.2)	4.8 (3.3)
Treatment Condition	3.6 (3.5)	16.3 (4.8)	9.0 (4.7)	5.4 (3.5)

Note. Standard deviations are in parenthesis.

2.6.4. Results

We recruited 108 subjects, all of which completed the experiment. One subject solved the final question in the post-test and was thus excluded according to our pre-registered filter criteria. Of the remaining 107 subjects, almost everyone had either no prior coding experience in Python (79%) or was a beginner (17%). On average, subjects took 89 minutes to complete the experiment and earned 18.50 euros. Table 27 in Appendix B provides summary statistics of all pre-treatment variables.

Subjects in the treatment condition sent 4.45 (median = 3, S.D. = 3.94) messages on average to the LLM and six subjects did not use the LLM at all.[7] Table 4 shows the average performance of the treatment and control conditions in each of the three experimental phases. The treatment condition outperformed the control condition in each of the three phases, solving half a question more in the *Pre-test* and one question more in both the *Learning Phase* and the *Post-test*. None of the three differences are statistically significant in two-tailed Welch's t-tests ($p = 0.425$, $p = 0.175$, $p = 0.206$). The difference of *Post-test − Pre-test*, which adjusts for differences in prior coding experience, is half a question larger in the treatment condition, but also not significant ($p = 0.376$).

In addition, we report the results of regressions in Table 5, in which we adjust for observable pre-treatment characteristics to control for random differences between the control and treatment conditions. As in the model-free analysis, the treatment effect is not significant (column 1, $p = 0.580$).

[7] The six subjects who did not use the LLM performed better than other treated (*Post-test − Pre-test* = 7.8). Results are similar if we exclude these six subjects (see Table 29 in Appendix D)

Table 5 Study 2: Regression Analyses.

	Post-test		*Learning Phase*
	(1)	(2)	(3)
Treatment (LLM access = 1)	0.396 (0.713)	-0.276 (0.474)	0.828 (0.662)
Gender (male = 1)	0.271 (0.714)	-0.978** (0.484)	1.540** (0.663)
Level of Studies	1.025** (0.512)	-0.092 (0.353)	1.377*** (0.476)
GPA	-0.594 (0.567)	0.017 (0.378)	-0.753 (0.527)
Age	-0.578 (0.462)	0.526 (0.320)	-1.359*** (0.429)
Coding Experience	0.639 (0.606)	0.276 (0.401)	0.447 (0.563)
Python Experience	-0.161 (0.803)	0.119 (0.531)	-0.344 (0.747)
Studiousness	-0.158 (0.259)	-0.066 (0.171)	-0.114 (0.241)
LLM Used Before (yes = 1)	0.537 (1.105)	0.925 (0.730)	-0.478 (1.027)
LLM Experience	-0.309 (0.290)	0.081 (0.195)	-0.481* (0.270)
Pre-test	0.761*** (0.141)	0.192* (0.106)	0.701*** (0.131)
Learning Phase		0.812*** (0.073)	
Constant	5.994** (2.564)	-6.834*** (2.047)	15.804*** (2.383)
Observations	107	107	107
R^2	0.472	0.772	0.512
Adjusted R^2	0.411	0.743	0.455

Notes. Standard errors are in parenthesis. *: p<0.1; **: p<0.05; ***: p<0.01.

In our theoretical framework, LLMs can affect learning outcomes by changing the volume of topics subjects can study and/or by enhancing or harming their understanding of each topic. While we cannot observe whether subjects have understood a topic directly, we can estimate the effect of the LLM on understanding by controlling for the LLM's effect on the volume of topics covered during the *Learning Phase*. Any remaining treatment effect must be due to subjects' increased understanding. In the second column of Table 5, we thus control for subjects' progress during the *Learning Phase*. In this setup, too, the treatment effect remains not significant (p = 0.562). In the third column, we can see that LLMs also do not affect volume (p = 0.214). Hence, we do not find support for a positive or a negative effect of LLMs on learning.

2.7. Revised Hypotheses

In Study 2, we do not find support for either Hypothesis 1 or Hypothesis 2. While conducting the experiment, we noticed that the software on the computers in the laboratory blocked all

options to copy and paste text. This mechanism was unintended by us and we hypothesize that this affects LLM usage and thus also learning outcomes.

Without copy and paste, subjects cannot easily copy task descriptions into the LLM or code snippets generated by the LLM back into the code editor. Instead, they have to manually copy text by typing it themselves. Such small increases in transaction costs are known to change behaviors (Lavecchia et al. 2016, Ericson and Laibson 2019). Here, the inability to copy and paste poses a barrier to LLM usage and thereby also limits its effect on learning. For one, subjects may forgo the LLM entirely because they perceive the effort of manually copying questions and LLM-generated solutions as too high. Or, as we observed among several subjects in Study 2, they make mistakes and thus the LLM does not provide the correct answer, or they copy a solution incorrectly. Thus, LLM usage and consequentially its effect on learning are reduced without copy and paste. Consistent with behavioral research that incorporated initially unintended experimental conditions into their theorizing (e.g., Lee and Sosa 2024), we revise our hypotheses to include copy and paste:

Revised Hypothesis 1 *When copy and paste are enabled, having access to LLMs while learning to code will increase the learning outcome of students.*

Revised Hypothesis 2 *When copy and paste are enabled, having access to LLMs while learning to code will decrease the learning outcome of students.*

Further, we add a direct effect of copy and paste on LLM usage:

Hypothesis 3 *When copy and paste are enabled, people use LLMs more during learning.*

2.8. Study 3: Laboratory Experiment with Copy and Paste

To test the revised hypotheses, we conducted a second experiment with copy and paste. Study 3 is an incentivized and pre-registered[8] laboratory experiment identical in its design to Study 2, except for the availability of copy and paste. Subjects could copy and paste text

[8] https://aspredicted.org/SXV_DLL

Table 6 Study 3: Correctly Answered Questions per Experimental Phase.

	Pre-test	*Learning Phase*	*Post-test*	*Post-test − Pre-test*
Control Condition	2.6 (2.4)	14.3 (5.0)	6.8 (4.4)	4.2 (3.5)
Treatment Condition	3.7 (3.9)	18.0 (4.8)	8.8 (5.1)	5.0 (3.5)

Note. Standard deviations are in parenthesis.

from question descriptions, code, and their conversations with the LLM (in the treatment condition) by right-clicking or with keyboard shortcuts. Beyond interacting with the LLM, the ability to copy and paste yielded no other discernible advantage in the experiment. All sessions took place in June 2024 three weeks after Study 2.

Of 72 enrolled participants, two were excluded for not completing the experiment and one for solving the final question (see Section 2.6.3), resulting in 69 subjects. On average, subjects took 90 minutes and earned 17.88 euros. Prior Python coding experience was similar to Study 2 (78% no prior experience and 14% beginners). Table 27 in Appendix B provides summary statistics of all pre-treatment variables.

Table 6 shows the average performance of the treatment and control conditions in each of the experimental phases. The treatment condition outperforms the control condition in all three phases. In the *Pre-test*, treated subjects solved one more question than controls ($p = 0.157$); in the *Learning Phase*, treated subjects solved four more questions ($p = 0.003$); and in the *Post-test*, treated subjects solved two more questions ($p = 0.092$). The difference of *Post-test − Pre-test* is on average one question larger for the treatment condition, but not statistically significant ($p = 0.312$). In Table 7, the treatment effect on overall learning outcomes remains insignificant if we control for observed pre-treatment covariates (column 1, $p = 0.298$). As in the model-free analysis, LLM access increased the volume of topics covered in the *Learning Phase* (column 3, $p = 0.013$), but it did not affect understanding when we control for subjects' progress during the *Learning Phase* (column 2, $p = 0.359$).[9]

[9] Results are similar if we exclude two subjects who did not use the LLM (see Table 29 in Appendix D).

Table 7 Study 3: Regression Analyses.

	Post-test		*Learning Phase*
	(1)	(2)	(3)
Treatment (LLM access = 1)	0.959 (0.912)	-0.651 (0.705)	2.614** (1.020)
Gender (male = 1)	1.919** (0.901)	1.010 (0.672)	1.474 (1.007)
Level of Studies	0.195 (0.587)	0.410 (0.431)	-0.348 (0.656)
GPA	-2.216** (0.834)	-0.489 (0.657)	-2.804*** (0.933)
Age	-0.333 (0.376)	-0.283 (0.276)	-0.081 (0.421)
Coding Experience	1.780** (0.681)	1.081** (0.508)	1.134 (0.761)
Python Experience	-0.676 (0.730)	-0.479 (0.535)	-0.319 (0.815)
Studiousness	-0.262 (0.275)	-0.187 (0.201)	-0.120 (0.307)
LLM Used Before (yes = 1)	1.266 (1.273)	0.923 (0.933)	0.557 (1.423)
LLM Experience	-0.333 (0.347)	-0.367 (0.254)	0.056 (0.388)
Pre-test	0.769*** (0.145)	0.325** (0.123)	0.721*** (0.162)
Learning Phase		0.616*** (0.087)	
Constant	7.360** (3.085)	-2.987 (2.686)	16.795*** (3.448)
Observations	69	69	69
R^2	0.638	0.810	0.600
Adjusted R^2	0.569	0.769	0.522

Notes. Standard errors are in parenthesis. *: p<0.1; **: p<0.05; ***: p<0.01.

2.9. Exploratory Analyses

To further investigate how LLMs affect learning, we conduct a number of exploratory analyses in the combined sample of Studies 2 and 3.[10] The studies are comparable because they were run within three weeks of each other in June 2024, both drew from the same subject pool and used the same lab, both followed the same experimental procedure, and the control conditions did not perform significantly different in any of the three experimental phases ($p = 0.433$, $p = 0.437$, $p = 0.260$).

2.9.1. Usage Behaviors

In both experiments, we observe all messages subjects sent to the LLM. We manually code these messages based on what the subjects wanted the LLM to do. We distinguish between messages that asked for solutions (e.g., "complete the function hypotenuse...") and messages that asked for explanations (e.g., "explain what the str() function does"). When subjects

[10] We mention the following exploratory analyses in both Studies' pre-registrations.

Table 8 Distributions of Usage Behaviors.

	Solutions	*Explanations*	*Miscellaneous*	*Translations*	*User Errors*
Study 2	42.2%	31.3%	11.6%	2.4%	12.4%
Study 3	64.2%	18.4%	7.2%	6.1%	4.1%
Total	54.1%	24.4%	9.2%	4.4%	7.9%

asked for solutions, they substituted some of their learning activities (working through the practice exercises) with the LLM. In contrast, when subjects asked for explanations, they complemented their own learning activities with the LLM by requesting additional information on a subject. The two usage behaviors are similar to the output and input categories in Keppler et al. (2024). Any other messages were either translations,[11] unrelated to the coding questions (miscellaneous), or user errors. See Appendix C for details on the coding process and Table 8 for the distributions of the usage behaviors. In our analyses, we focus on the number of times subjects substituted their learning activities by asking for *Solutions* and the number of times they complemented their learning activities by asking for *Explanations*.

We start by analyzing how the availability of copy and paste in Study 3 impacted how subjects used the LLM. Consistent with Hypothesis 3, copy and paste increased LLM use. On average, treated subjects in Study 3 sent 7.71 (S.D. = 5.50) messages to the LLM, which is significantly more than the 4.45 messages sent in Study 2 ($p = 0.002$; Table 9, column 3: $p = 0.001$). Only two subjects in Study 3 did not use the LLM at all.

The availability of copy and paste not only changed how often subjects referred to the LLM, but also for which kinds of tasks they referred to it. As shown in columns 1 and 2 of Table 9, treated subjects in Study 3 asked for significantly more *Solutions* ($p < 0.001$) but not for more *Explanations* ($p = 0.936$) than treated subjects in Study 2. By making the LLM easier to use, copy and paste increased substitutive but not complementary use, suggesting that subjects intrinsically preferred the former. Furthermore, we note that prior experience

[11] All translations are from English to German. We did not ask for subjects' nationality, but given the laboratory's location in Germany, the vast majority of subjects will have been German.

Table 9 The Effect of Copy & Paste on Usage Behaviors.

	Solutions (1)	*Explanations* (2)	*Messages* (3)
Copy/Paste (enabled = 1)	3.066*** (0.755)	0.032 (0.403)	3.273*** (0.967)
Gender (male = 1)	1.337 (0.815)	-0.309 (0.435)	1.007 (1.044)
Level of Studies	-0.229 (0.547)	0.273 (0.292)	-0.280 (0.701)
GPA	0.735 (0.643)	0.028 (0.344)	1.028 (0.824)
Age	0.300 (0.451)	-0.205 (0.241)	0.299 (0.578)
Coding Experience	-0.123 (0.706)	0.778** (0.377)	0.691 (0.904)
Python Experience	0.384 (0.728)	-0.692* (0.389)	-0.466 (0.933)
Studiousness	-0.164 (0.279)	-0.148 (0.149)	-0.450 (0.358)
LLM Used Before (yes = 1)	-3.538*** (1.228)	0.581 (0.656)	-3.567** (1.573)
LLM Experience	0.890*** (0.328)	-0.138 (0.175)	0.954** (0.421)
Pre-test	-0.118 (0.126)	-0.005 (0.068)	-0.083 (0.162)
Constant	1.079 (2.447)	1.360 (1.307)	4.713 (3.136)
Observations	94	94	94
R^2	0.322	0.084	0.264
Adjusted R^2	0.231	-0.039	0.165

Notes. Regressions include treated subjects from Studies 2 and 3. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

with LLMs also determined usage behavior: subjects who had never used an LLM before (*LLM Used Before* = 0) and those who used LLMs frequently (*LLM Experience*) asked for *Solutions* more often ($p = 0.005$, $p = 0.008$).

We next use the availability of copy and paste in Study 3 to estimate how the associated change in usage behavior—specifically the increase in substitutive *Solutions* requests—affected learning outcomes.[12] In Table 10, column 1, copy and paste did not affect overall learning outcomes ($p = 0.737$). However, copy and paste decreased topic understanding (column 2, $p = 0.009$) and increased topic volume (column 3, $p = 0.033$). Treated subjects in Study 3 worked through 1.7 more question during the *Learning Phase*, but holding this progress constant they solved 1.4 fewer questions in the *Post-test*. Thus, when students use LLMs to substitute learning activities, there is a trade-off between volume and understanding.

[12] The availability of copy and paste did not affect learning outcomes other than through its effect on LLM usage behavior. The control condition in Study 3 (who could also use copy and paste) did not perform significantly different from the control condition in Study 2.

Table 10 The Effect of Copy & Paste on Learning Outcomes.

	Post-test		*Learning Phase*
	(1)	(2)	(3)
Copy/Paste (enabled = 1)	-0.249 (0.737)	-1.407*** (0.525)	1.733** (0.800)
Gender (male = 1)	2.143*** (0.796)	0.549 (0.576)	2.384*** (0.863)
Level of Studies	0.351 (0.534)	0.352 (0.370)	-0.001 (0.580)
GPA	-1.086* (0.629)	-0.259 (0.444)	-1.237* (0.682)
Age	-0.297 (0.440)	-0.188 (0.305)	-0.163 (0.478)
Coding Experience	1.086 (0.690)	1.027** (0.478)	0.088 (0.748)
Python Experience	-0.982 (0.712)	-0.646 (0.494)	-0.501 (0.772)
Studiousness	0.110 (0.273)	0.067 (0.189)	0.063 (0.296)
LLM Used Before (yes = 1)	-0.365 (1.200)	0.157 (0.832)	-0.780 (1.301)
LLM Experience	-0.146 (0.321)	-0.253 (0.222)	0.160 (0.348)
Pre-test	0.842*** (0.124)	0.326*** (0.101)	0.772*** (0.134)
Learning Phase		0.669*** (0.070)	
Constant	6.747*** (2.391)	-3.897* (2.000)	15.913*** (2.594)
Observations	94	94	94
R^2	0.571	0.797	0.482
Adjusted R^2	0.513	0.767	0.413

Notes. Regressions include treated subjects from Studies 2 and 3. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

On the one hand, students can increase the volume of topics they cover because they are faster, which increases learning outcomes. On the other hand, this increase in volume comes at the cost of a lesser understanding of each topic, which decreases learning outcomes. Although treated subjects in Study 3 covered more topics than in Study 2, they understood each topic less. If we combine both effects, they cancel each other out, which explains the insignificant effect in column 1. The negative effect of substitution on understanding is likely due to reduced effort: In Study 3, 42% of messages asking for a solution were sent without a single attempt to solve the corresponding question. Moreover, two participants remarked in the open text field at the end of the experiment that they did not learn anything as they relied too much on the LLM to solve the practice questions.

Lastly, to estimate how complementary use affected learning outcomes, we regress *Post-test* on the number of *Explanations* requested. Although we did not exogenously vary *Explanations*, *Pre-test* allows us to control for subjects' initial ability, which along with

Table 11 The Effect of Usage Behaviors on Learning Outcomes.

	Post-test	
	(1)	(2)
Solutions	-0.195* (0.107)	-0.312*** (0.066)
Explanations	-0.003 (0.200)	0.249** (0.125)
Copy/Paste (enabled = 1)	0.350 (0.802)	-0.574 (0.499)
Gender (male = 1)	2.404*** (0.805)	0.885* (0.511)
Level of Studies	0.307 (0.533)	0.213 (0.328)
GPA	-0.943 (0.629)	0.046 (0.395)
Age	-0.239 (0.440)	-0.032 (0.271)
Coding Experience	1.064 (0.702)	0.789* (0.432)
Python Experience	-0.908 (0.721)	-0.321 (0.445)
Studiousness	0.077 (0.273)	0.049 (0.167)
LLM Used Before (yes = 1)	-1.055 (1.254)	-1.040 (0.770)
LLM Experience	0.027 (0.333)	0.048 (0.205)
Pre-test	0.819*** (0.123)	0.239** (0.091)
Learning Phase		0.735*** (0.064)
Constant	6.961*** (2.391)	-4.955*** (1.795)
Observations	94	94
R^2	0.588	0.847
Adjusted R^2	0.521	0.819

Notes. Regressions include treated subjects from Studies 2 and 3. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

other pre-treatment observables is likely to be one of the main confounders. However, in this case we cannot estimate the effect of *Explanations* on the volume of topics due to the simultaneity of usage behavior and *Learning Phase* progress: Behavior affects progress but progress also affects behavior. In Table 11, we see that *Explanations* did not affect overall learning outcomes (column 1, $p = 0.990$), but it did increase topic understanding (column 2, $p = 0.05$). One reason for why the positive effect of *Explanations* on understanding did not carry over to overall learning outcomes may be that asking for more *Explanations* reduced topic volume. However, due to the aforementioned simultaneity, we cannot estimate this. We also include *Solutions* in the regressions in Table 11 and the results are the same as with copy and paste in Table 10. In Appendix D, we perform various robustness checks for this and the previous analyses and all results hold.

How the different usage behaviors affect learning links back to our original hypotheses.

As tutors, LLMs increase understanding if students ask for explanations and increase topic volume if students ask for solutions (Hypothesis 1). However, neither effect comes without a penalty. Asking for explanations does not increase overall learning outcomes, possibly because it reduces topic volume, while asking for solutions decreases understanding (Hypothesis 2). Therefore, whether LLMs help or hurt learning depends on how they are used and the context of the learning task. If topic volume matters more than understanding, using LLMs to speed up practice exercises may increase overall learning outcomes. If topic volume matters less than understanding (e.g., when students can learn without time constraints, as was the case in the field study), using LLMs to develop a deeper comprehension of topics may increase overall learning outcomes. Instead of either Hypothesis 1 or 2 being true, our exploratory results suggest that they may both be true, depending on the way people use LLMs and the learning context.

2.9.2. Heterogeneous Effects

Previous studies have found that LLMs increase productivity the most for low-performing workers (Brynjolfsson et al. 2023, Noy and Zhang 2023, Dell'Acqua et al. 2023). Here, we test whether LLMs similarly affect learning depending on people's initial knowledge. To do so, we partition subjects in Studies 2 and 3 along the median of the combined sample's *Pre-test* (e.g., Dell'Acqua et al. 2023).

Table 12 shows the results for overall learning outcomes (column 1), understanding (column 2), and volume (column 3) in Studies 2 and 3 combined. The interaction of *Treatment* and *Pre-test* > *Median* significantly affects understanding and overall learning outcomes (column 2: $p = 0.034$; column 1: $p = 0.028$). Treated subjects who scored higher on the *Pre-test* increased their understanding and overall learning outcomes more than those who scored lower. Moreover, treated subjects who scored lower on the pre-test understood significantly less than control subjects (column 2: $p = 0.043$). Topic volume is not affected by the interaction of *Treatment* and *Pre-test* > *Median* ($p = 0.290$). Unlike in workplaces, LLMs appear to increase inequality in education, especially with respect to understanding.

Table 12 Heterogeneous Effects.

	Post-test		*Learning Phase*
	(1)	(2)	(3)
Treatment × *Pre-test* > *Median*	2.528** (1.142)	1.613** (0.754)	1.249 (1.177)
Treatment (LLM access = 1)	0.043 (0.701)	-0.950** (0.466)	1.355* (0.722)
Copy/Paste (enabled = 1)	-0.827 (0.553)	-1.222*** (0.365)	0.539 (0.570)
Gender (male = 1)	0.872 (0.578)	-0.131 (0.386)	1.368** (0.595)
Level of Studies	0.666* (0.390)	0.212 (0.258)	0.620 (0.402)
GPA	-1.078** (0.466)	-0.004 (0.315)	-1.465*** (0.480)
Age	-0.468 (0.306)	0.037 (0.204)	-0.689** (0.316)
Coding Experience	1.083** (0.466)	0.724** (0.307)	0.490 (0.480)
Python Experience	0.296 (0.555)	0.013 (0.366)	0.387 (0.572)
Studiousness	-0.188 (0.191)	-0.138 (0.126)	-0.068 (0.197)
LLM Used Before (yes = 1)	0.958 (0.833)	0.798 (0.548)	0.217 (0.858)
LLM Experience	-0.169 (0.232)	-0.101 (0.153)	-0.092 (0.239)
Pre-test > *Median*	2.784*** (0.890)	0.342 (0.609)	3.333*** (0.917)
Learning Phase		0.733*** (0.050)	
Constant	6.544*** (1.958)	-4.541*** (1.494)	15.128*** (2.017)
Observations	176	176	176
R^2	0.467	0.771	0.444
Adjusted R^2	0.424	0.751	0.399

Notes. Regressions include subjects from Studies 2 and 3. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

2.9.3. Perceived learning

Lastly, we investigate how LLMs affect perceptions of learning. Such perceptions can be different from actual learning and have been shown to disadvantage more effective teachers and teaching methods (Carrell and West 2010, Deslauriers et al. 2019). Higher perceived learning affects choices (Jensen 2010) and in the case of LLMs may thus affect adoption. As we have shown above, such adoption is not always beneficial. LLMs increase perceptions of self-efficacy (Noy and Zhang 2023) and may thus increase perceived learning.

At the end of the experiments, we asked subjects to rate their *Perceived Learning* on a five-point Likert scale. Table 13 shows the results for the combined sample of Studies 2 and 3. In all regressions, we include the difference of subjects' *Post-test* − *Pre-test* to control for actual learning. The estimated *Treatment* effect therefore represents subjects' perceptions that differ from actual learning. The effect of LLM exposure (*Treatment*) on *Perceived*

Table 13 The Effect of LLM Usage on Perceived Learning.

	Perceived Learning	
	(1)	(2)
Treatment (LLM access = 1)	0.338** (0.166)	0.348** (0.170)
Copy/Paste (enabled = 1)	0.128 (0.164)	0.140 (0.170)
Post-test − Pre-test	0.154*** (0.024)	0.162*** (0.035)
Gender (male = 1)	-0.079 (0.171)	-0.071 (0.174)
Level of Studies	-0.079 (0.116)	-0.077 (0.117)
GPA	-0.192 (0.140)	-0.198 (0.142)
Age	-0.194** (0.091)	-0.197** (0.092)
Coding Experience	0.086 (0.138)	0.083 (0.139)
Python Experience	-0.101 (0.166)	-0.102 (0.166)
Studiousness	0.052 (0.057)	0.053 (0.057)
LLM Used Before (yes = 1)	-0.672*** (0.246)	-0.678*** (0.247)
LLM Experience	0.020 (0.068)	0.020 (0.068)
Pre-test	0.083*** (0.030)	0.092** (0.044)
Learning Phase		-0.010 (0.035)
Constant	2.975*** (0.591)	3.079*** (0.699)
Observations	176	176
R^2	0.336	0.336
Adjusted R^2	0.283	0.278

Notes. Regressions include subjects from Studies 2 and 3. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

Learning is positive and significant (column 1: $p = 0.044$). This effect is independent of volume (column 2: $p = 0.043$). Thus, LLMs increase perceived learning by more than can be explained by actual differences in learning.

2.10. Discussion

How do large language models (LLMs) affect student learning? We conceptualize learning outcomes as the value added by education (Hanushek 2020), and decompose them into the volume of topics covered and the depth of understanding of each topic. In two pre-registered and incentivized laboratory experiments, we find no evidence that LLMs affect learning outcomes: neither the volume of topics that students go through nor the depth of their understanding of each topic changes when students have access to an LLM.

In exploratory analyses, we show that the effect of LLMs on learning outcomes depends on how students use them and their initial knowledge. Students who ask LLMs for solutions

to practice exercises increase the number of topics they cover, but they understand each topic less. Students who ask LLMs for explanations of topics increase their understanding of each topic. In a field study of two programming course in which we only observe if students asked for solutions, we find a long-term and practically relevant negative effect of LLMs on learning. The ability to copy and paste when interacting with an LLM is a strong determinant of usage behavior as it substantially increases the number of times students prompt the LLM for solutions.

Unlike low-performing workers, who improve their productivity with LLMs more than high-performing workers (Brynjolfsson et al. 2023, Dell'Acqua et al. 2023, Noy and Zhang 2023), we find that students with less initial knowledge learn less when using LLMs. The unrestricted availability of LLMs thus increases inequality in learning outcomes. Moreover, students who have access to an LLM overestimate how much they have learned.

Our results have several theoretical implications for our understanding of how technology and LLMs in particular affect learning. First, artificial intelligence (AI) can substitute or complement human activities. Many of the reported productivity gains of AI have occurred in cases in which the technology complemented human work (Fügener et al. 2022, Brynjolfsson et al. 2023, Dell'Acqua et al. 2023, Keppler et al. 2024). In education, both substitution (e.g., asking for solutions) and complementarity (e.g., asking for explanations) can support learning: Substitution increases the volume of knowledge that students learn because it speeds up the learning process, but it comes at the cost of reduced understanding. On the other hand, complementarity increases understanding. Our results indicate that in practical settings in which the time available for study is not a strongly limiting factor, understanding is more important than volume for overall learning outcomes. In such settings, LLMs should be used to complement students' learning activities rather than substitute them.

Second, previous debates on the efficacy of technology in education have focused on technology features (e.g., costs or interactivity) and how technology replaces other learning

activities (Bulman and Fairlie 2016, Chatterji 2018), but not on student behavior. Our results suggest that students prefer to use LLMs to substitute rather than complement learning activities. There are two possible explanations. For one, students may expect that substitution increases overall learning outcomes because it allows them to cover more topics. However, we do not see empirical evidence of improved actual or perceived learning outcomes from greater volume. A more likely explanation is that students' self-control problems extend to technology use. Learning is an effortful task and students exhibit a present bias which makes them prefer immediate gratification over exerting effort for learning in anticipation of future rewards (Lavecchia et al. 2016, Damgaard and Nielsen 2020). Students may thus prefer substitution because it reduces their learning effort.

Third, a small change to the LLM interface can significantly alter student behavior. Specifically, the availability of copy and paste increases substitutive use but does not affect complementary use. Disabling copy and paste increases the immediate transaction costs of interacting with the LLM, thereby offsetting the decrease in learning effort.

Our study has a number of limitations that offer promising directions for future research. Despite the strong identification in our exploratory analyses, these findings should ideally be replicated in controlled experiments and with additional field data. Furthermore, we study learning outcomes that are a function of topic volume and topic understanding. We abstract away from the neurological building blocks of understanding, such as conceptual and factual understanding (Mueller and Oppenheimer 2014) and memory (Wuttke et al. 2022). Studying how LLMs influence the neurological building blocks of learning may offer interesting insights. Moreover, LLMs may also affect learning in workplaces. As employees accumulate experience with a task, their performance tends to increase—so-called learning-by-doing (Arrow 1962). LLMs may change employee learning in two ways. On the positive side, LLMs can teach employees how to become better at their jobs, and this learning may persist without LLMs (e.g., Brynjolfsson et al. 2023). On the negative side, LLMs may reduce how attentive employees work on a task or how much a task stimulates them mentally (e.g.,

Wang et al. 2020, Wuttke et al. 2022, Dell'Acqua 2022). In the latter case, employees may not learn from experience anymore and may even unlearn some of their skills. As we have shown in education, the effect of LLMs on learning in workplaces likely depends on several factors, such as the complexity of the task involved and how employees use the LLM.

In practice, educators have been encouraged to embrace LLMs and mitigate their potential harm by advising students to use them as complementary aids for learning. However, with users' difficulty in maintaining self-control and the growing power of LLMs, adherence to voluntary guidelines may be elusive. This presents an ongoing challenge for education managers in both academic and corporate settings: How can they ensure that LLM-supported learning programs meet educational and training objectives while promoting high-quality knowledge transfer and equitable access for diverse learners.

Our experiments introduce a simple design tweak to a real-world LLM interface that fosters more effective knowledge transfer and can improve learning outcomes. This design may thus serve as a proof-of-concept for education managers seeking to implement LLM technology in their programs. We hope that its core principle—stimulating learners' mental engagement with rich problem-solving support while discouraging passive or superficial use—will inspire the development of innovative LLM-supported training programs that are effective in both academic and business environments.

Don't Stop Sketching! Design Ideation Processes in the Age of Generative AI

Matthias Lehmann
University of Cologne, matthias.lehmann@wiso.uni-koeln.de

Henrik Franke
University of Cologne, franke@wiso.uni-koeln.de

Fabian J. Sting
University of Cologne, Rotterdam School of Management, Erasmus University, sting@wiso.uni-koeln.de

Abstract. Generative AI challenges existing idea generation processes with the capability to explore rapidly at low cost. Thus, how to effectively integrate AI into existing processes has become a core question for innovation and technology managers. We focus on design ideation, where it is unclear whether traditional sketching remains valuable in AI-augmented workflows. We address this gap through two preregistered, incentivized experiments – one with non-experts and one with prospective design professionals – which simulate logo-design contests where participants generate designs using generative AI. Participants are randomly assigned to sketch a logo prior to AI usage or to a control drawing task. We quantify idea variance via neural network-based similarity metrics on prompts and images, and assess design quality through blind pairwise crowd evaluations using an Elo rating system (a ranking method common in competitive games, e.g., ubiquitous in chess). Across both studies, sketching markedly reduces exploration breadth – fewer distinct design clusters and smaller idea variance – yet yields significantly higher average and best design quality. Exploratory analyses suggest that sketching induces frontloading of mental engagement which helps designers choose a better starting point for their design iterations and prevents off-task explorations, i.e., distractions afforded by the vast AI capabilities. These findings challenge conventional ideation theory that greater process variance uniformly enhances outcomes; instead, in AI-rich contexts, focused ideation anchored by deliberate planning can outperform unguided ideation. Practically, our results advise integrating traditional sketching or planning stages into AI-driven ideation to harness AI's potential without

sacrificing quality.

Key words: generative artificial intelligence, design, ideation, innovation management, technology management

3.1. Introduction

The effective management of ideation processes is a major determinant of innovation management outcomes (Girotra et al. 2010, Kavadias and Hutchison-Krupat 2020). With the rapid emergence and diffusion of generative artificial intelligence (AI) – including large language models (LLMs) such as ChatGPT and image generation models such as Dall·E – organizations need to decide how such novel technologies should be incorporated into ideation processes. This question is particularly pressing in sectors that have been disrupted by generative AI models, such as the design industry. The introduction of ChatGPT and its built-in image generation capabilities has drastically reduced the job postings in graphic design by 17% from 2021 to 2023 (Demirci et al. 2025). In other words, practice is rushing to replace visual designers with automation. At the same time, studies consistently suggest that AI is most productive when it restructures processes and redistributes tasks between humans and technology, rather than automating the entire workflow (Dell'Acqua et al. 2023, Spring et al. 2022, Boussioux et al. 2024). What is yet missing is a process-level understanding of how to sequence human and AI-augmented activities specifically. Our study addresses this gap by uncovering the role that sketching plays in the early stages of AI-supported design ideation workflows. Initial sketching is a standard methodology at the start of classical visual design ideation, but we know little about the interaction with AI technology. With insights on the effects of sketching on subsequent AI-enabled design, we shed light on how to combine human designers and generative AI effectively.

In design processes, designers develop and refine visual design ideas, such as logos for a company or entire marketing campaigns. Design processes are especially affected by generative AI (Demirci et al. 2025). The disruption of the industry stems from image generation models that enable near-instant design creation and can thus largely substitute

manual effort in design processes. However, their promise goes beyond simple substitution. First, AI democratizes design and levels the playing field by equipping the average person with sophisticated design abilities (Lau et al. 2024). It is becoming increasingly difficult to distinguish between designs created by professionals and those generated by amateurs using AI (Osadcha and Osadcha 2023). Second, AI enables more rapid iteration and ideation as the time required to produce each design is drastically reduced. Third, AI also allows for broader ideation. By quickly testing different designs – no longer limited by manual drawing skills – designers can explore a wider design space. Together, these affordances suggest that generative AI will continue to be integrated into design processes. What remains to be clarified is the ideal role of AI in the design process and whether established steps, such as sketching, help or hinder the AI-enabled design process. Specifically, it remains unclear whether sketching retains its value when generative image models drastically lower the cost of exploration. In this study, we therefore ask: *Should designers still sketch before AI-augmented design processes, or do AI's rapid exploration capabilities make sketching obsolete?*

To answer this question, we conduct two pre-registered and incentivized experiments with prospective design professionals and amateurs in which we manipulate whether subjects first manually sketch their idea before collaborating with generative AI to come up with a logo design. We find that sketching reduces process variance as designers modify their designs less across iterations and generate fewer distinct design ideas. Nonetheless, beginning the design process with sketching improves both the average quality and the quality of the best design, for professionals and amateurs alike. We assess quality through crowd evaluations following prior works and since logos are intended for the general public. In subsequent exploratory analyses, we argue that the positive effect of sketching on design quality is at least partially induced by designers choosing a better starting point in the generative design process and by being more focused on the core design task, whereas otherwise the capabilities of generative image models can have distracting effects. We find no evidence of interactions with prior generative AI experience.

We make several theoretical contributions. First, we contribute to the literature on innovation management (e.g., Van de Ven 1986, Garcia and Calantone 2002, Kavadias and Ulrich 2020). Variance in the generated ideas is regarded as generally helpful in ideation literature as it increases the likelihood of exceptional ideas (Girotra et al. 2010). We show that this no longer has to be the case in the realm of generative AI and that increased variance of ideas can surprisingly even co-occur with deteriorating performance. To reconcile this finding with existing theories, we suggest that – next to the known-to-be-useful process variance – the AI-enabled ideation process introduces unwanted process variance, which does not lead to the discovery of useful ideas and is visible as distractions from the original task. By revealing in which process configurations this troublesome distracting variance occurs, we help answer the currently pressing question of how to most effectively integrate AI into existing processes to yield effective AI-empowered ideation.

Second, we also add to the design literature (e.g., Jansson and Smith 1991, Buchanan 1992, Brown et al. 2008, Ulrich and Eppinger 2016, Chan et al. 2018) by showing that design fixation effects cannot only be induced by external stimuli (Jansson and Smith 1991, Smith et al. 1993, Youmans and Arciszewski 2014, Lee and Sosa 2024) but also by deliberately sequenced activities performed by designers themselves, e.g., their own sketching in advance of AI exploration. We add that such cognitive fixation must not always lead to negative results but can sharpen task focus if the anchor is deliberately chosen. Through sketching, designers can increase task focus and safeguard against AI-induced off-task explorations.

Finally, our study also relates to the search literature (e.g., March 1991, Levinthal 1997, Fleming 2001). We find that when humans search for the best design with the help of AI in a large-dimensional and rugged fitness landscape, there is a fine line between exploration and a loss of focus that in turn prompts tinkering with inferior solutions. By unearthing these micro-level foundations of search behavior and particularly on the perils of AI-supported exploration, we add to research on human search behavior (Billinger et al. 2014, Raisch and

Fomina 2023), that has argued for suboptimal balances between exploitation and exploration that humans choose. We show that this bias may be further amplified through the vast and potentially distracting capabilities of generative AI.

3.2. Related Literature

Empirical studies at the intersection of sketching and generative AI in design processes remain scarce. We approach our review of related works from two angles: first, we discuss works which analyze the effects of generative AI on design processes, then we focus on works which explicitly analyze the effects of sketching.

3.2.1. Generative AI in Design Processes

Generative AI is currently disrupting the design industry yet empirical research on how the integration of AI affects design processes and outcomes is still limited. Early works suggest that generative AI can expand exploration, foster creativity, and provide innovative ideas in design processes (Mountstephens and Teo 2020, Jin and Lee 2024, Liao et al. 2024, Li et al. 2024). This literature emphasizes a potential boost to the outcome quality of the final design through improving the effectiveness of the ideation process. At the same time, other studies suggest that generative models streamline design processes, enable faster iterations and allow for time savings (Tholander and Jonsson 2023, Jin and Lee 2024). This view emphasizes not effectiveness but efficiency: the same design quality can be achieved in a shorter time. Beyond effectiveness and efficiency, studies find that the democratizing element of generative AI allows non-experts to participate in design processes by alleviating skill-based entry barriers (Osadcha and Osadcha 2023, Lau et al. 2024). In total, generative AI may overhaul the structure of the entire design value chain.

Our study focuses on uncovering causal evidence on the design process itself, not on the broader value chain including customers and suppliers. Few studies experimentally tested this impact of AI on design processes. Tholander and Jonsson (2023) show that designers using AI for ideation save time while also creating more diverse ideas. Cai et al. (2023) conduct a laboratory experiment showing that designers rate generative AI as more

inspirational and enjoyable than traditional search but that high diversity achieved through AI has low added value.

3.2.2. The Role of Sketching in Design

Design literature has long investigated the role of sketching. Reviews on the role of sketching find that sketching serves as an external visual memory of ideas – an attempt of reproducing mental images – and aids design iteration, creativity and discovery of designs through the rapid representation of ideas (Purcell and Gero 1998, Tovey et al. 2003). These visualized ideas also enable quicker communication, retrieval and iteration. On the contrary, Goldschmidt (1991) sees sketching less as a representation of mental images but instead as visual reasoning steps towards a final design.

Empirical studies on the effects of sketching are limited. Song and Agogino (2004) conduct an empirical study of design students in new product development and find that the number of sketches and their level of details correlate with the final design outcome quality. In an experiment with architects, Bilda et al. (2006) allow only one experimental group to sketch. They find no difference in design outcomes, cognitive activity or idea links, concluding that sketching is not essential for experts in design processes. Finally, a case study of designers by Jonson (2005) reveals that sketching only infrequently provides breakthrough ideas. In addition, they find a discrepancy between designers and clients. Whereas designers prefer to sketch, clients expect photorealistic images already at the ideation stage. Overall, the research on sketching is divided and partly considers sketching useful but partly also questions its benefits.

No study has yet examined the effect that typical preparation exercises, like sketching, have on AI-enhanced creative design processes. A rare related example in Wadinambiarachchi et al. (2024) finds that using AI for inspiration leads to higher design fixation and less divergent thinking during sketching afterwards, reducing both the number of sketches and their originality. Our study focuses on the more common approach in practice, which is to use sketching before the iterative design process (Goldschmidt 1991, Purcell and Gero 1998,

Jonson 2005). Another related study, Zhang et al. (2023), focuses on the use of generative AI *while* sketching in an observational study with architectural design students. Anecdotal participant reports state that AI does not enhance the sketching itself but provides inspiration. Other participants reported that AI can improve the visual quality of hand-made sketches.

3.3. Theory

In this section, we theorize on the effect of sketching prior to AI use on idea variance and quality outcomes in AI-augmented design processes. Design processes are a form of ideation during which designers seek the best design through generating and iterating multiple ideas. According to fundamental design thinking principles (Brown et al. 2008), process variance – strategically creating a variety of sketches or concepts – is essential to identify promising solutions. Only with sufficiently diverse ideas can designers effectively select the most promising among these while idea variance also increases the likelihood of exceptional ideas (Girotra et al. 2010). The need for exploration further increases when faced with novel design challenges, where the search space is yet less known to the designer (Fleming and Sorenson 2004).

Generative AI facilitates rapid testing of a variety of designs. Whereas traditionally designers had to sketch out each idea manually, a non-trivial and time-consuming process, designers can now visualize any design idea within seconds via a short text prompt to an image generation model. Simultaneously, AI opens areas of the design space previously inaccessible due to manual-drawing constraints. Designers no longer need high manual sketching skill to externalize ideas; instead, they can now create even hard-to-sketch designs through the capabilities of generative image models to visualize virtually anything that can be put into words. As a result, generative AI shifts the exploration landscape: exploration becomes broader, faster and cheaper. Thus, AI has the potential to greatly boost idea variance.

Given this facilitation of exploration, the question emerges whether sketching, which previously filled this exploratory role at the fuzzy front-end of design ideation (Goldschmidt

1991, Purcell and Gero 1998, Tovey et al. 2003), is still necessary in AI-augmented design processes. Indeed, first evidence suggests that generative AI can yield diverse design ideas while streamlining the ideation process and effectively producing high quality sketches (Tholander and Jonsson 2023, Zhang et al. 2023, Wang et al. 2024). Conversely, one can argue that sketching may even be harmful in AI-augmented design processes by impairing the aforementioned benefits of using generative AI. Specifically, we posit that sketching prior to using generative AI in the design process actively constraints the exploration potential of generative AI by inducing design fixation. Design fixation means the conscious or unconscious anchoring to latent designs (Youmans and Arciszewski 2014, Vasconcelos and Crilly 2016), such as existing designs (Jansson and Smith 1991, Smith et al. 1993) or external stimuli (Lee and Sosa 2024), to which the designer is exposed prior to or during the design process, and results in significantly reduced idea variance and creativity. Notably, design fixation is even prevalent among design experts (Crilly 2015). We argue that sketching prior to using generative AI induces similar design fixation effects. Designers may anchor to the design ideas outlined in their initial sketches when subsequently using generative AI. With these fixed points, idea variance in the vast AI-enabled space will be limited and sketching may inhibit exploration in AI-augmented design processes. We summarize our main hypothesis as:

Hypothesis 4 *Sketching before using generative AI to create design ideas will reduce the idea variance and exploration of the design space, and instead lead to a higher similarity across the created designs.*

According to the ideation theory discussed in the beginning that idea variance drives idea quality, the hypothesized negative effect of sketching on process variance also would imply a detrimental effect of sketching on design quality.

3.4. Methods

We test our theories via incentivized and pre-registered[13] online experiments. In a between-subject setup, we measure the participants' idea variance and design quality in an AI-augmented logo design task. The experimental setup replicates typical online logo design contests. We manipulate the design process by instructing a random subset of subjects to sketch before starting the main design process with the AI tool.

3.4.1. Experimental Design

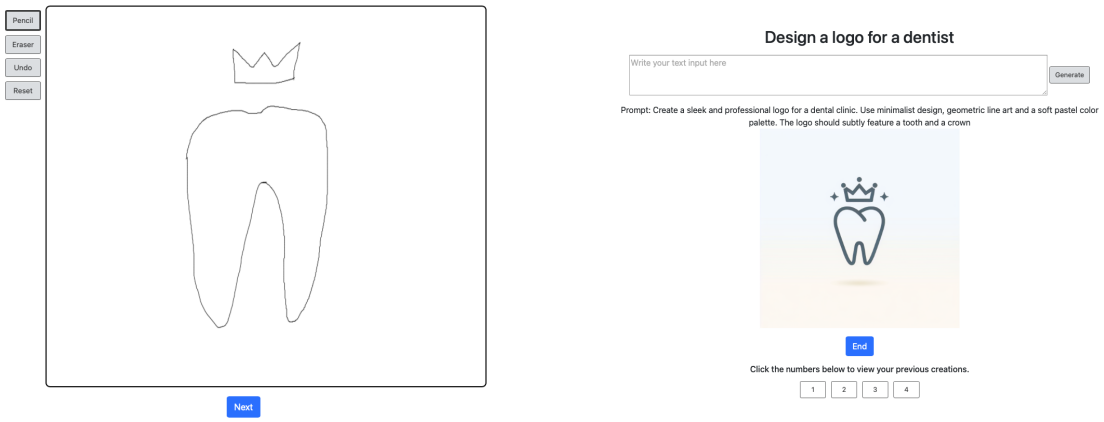
We implement a logo design contest: All experiment participants compete against each other to create the best design. They have up to 45 minutes to create as many designs as they want[14]. Afterwards, they select a final design which enters the contest. Subjects create designs by prompting a generative AI tool, which generates images based on these text inputs. The text prompts can contain arbitrary contents and lengths. We use DALL·E 3 (Betker et al. 2023) as the image generation model, which we query through the OpenAI API. During the design process, subjects can retrieve and re-view all their previously generated designs and the respective prompts through the interface. To set realistic design constraints, we instruct subjects to create a dentist logo. The experiment platform is built on oTree Chen et al. (2016). In Figure 1, we show screenshots of the experimental interface.

As the design task, we choose logo design for three main reasons: (1) logo design is among the most prevalent tasks in online design competitions (e.g., see `https://en.99designs.de/contests` or `https://www.designcrowd.com/contests`); (2) logo design is well suited for the use of generative AI as its output is a single image and current AI models can produce high-quality logos; (3) logo design is well suited for an experiment with non-expert subjects and under time constraints as subjects can reasonably create sufficiently many logos within 45 minutes, even non-experts can create simple logo sketches, and non-experts can sufficiently evaluate their own designs.

[13] `https://aspredicted.org/X2V_3QR`

[14] For practical reasons, we implement an upper limit of 100 designs. No subject reached this limit.

Figure 1 Experimental Interface.



Note. Left: Interface for sketching. Right: Interface during the main design phase.

To elicit effort, we incentivize subjects with a prize of 60 euros for the winner of the design competition. The subjects were informed that the winner would be chosen by an expert jury[15]. All subjects receive a fixed compensation of 7 euros in accordance with the laboratory's minimum wage requirements. We exclude subjects who do not complete the entire experiment. In addition, we exclude participants who only generated a single design. On the one hand, this suggests a clear lack of effort from the subjects. On the other hand, we cannot measure exploration behavior through similarities (see Section 3.4.3.1) as the single design represents only a single point in the design search space, essentially meaning that no exploration occurred. The experiment was approved by the ethics committee of a public German university.[16]

After the design process, we ask subjects to rate their prior experience with image generation models, their behavior during the experiment and the outcomes of the design process on five-point Likert scales. Among others, we ask for how much they think that they explored different designs, the perceived quality of the designs, their overall satisfaction and whether they had a specific design in mind during the process.

[15] The expert jury consisted of four dentists. This choice is appropriate as in a realistic online design competition, these also would be the client who makes the final decision on which logo to use.

[16] Ethics Committee of the Faculty of Management, Economics and Social Sciences (ERC-FMES) at the University of Cologne. Reference number: 230052ML.

3.4.2. Manipulation

We manipulate whether subjects sketch by randomly assigning them to one of two conditions. Both conditions have to complete an additional task prior to the main AI design phase.

Treatment condition: Subjects have 13 minutes to create sketches of their envisioned dentist logo design using a minimalistic mouse-based drawing tool in the experimental interface. As they would in an actual design process, subjects can also see their sketch during the main design phase. We use a simplistic sketching tool (rather than sketching on a paper) due to its normalizing effects such that no subject is particularly skilled and creates better (i.e. more advanced or detailed) sketches. This also allows non-experts who never sketched before to participate in the experiment.

Control Condition: Subjects solve an alternative task which is unrelated to the main design task. They have 13 minutes to complete a connect-the-dots image using the same drawing tool. We selected this alternative task as it is comparable in nature to sketching (eliminating potential effects by the drawing process itself). Importantly however, this task requires no creative ideation as they have to simply connect as sequence of dots and uses imagery unrelated to the main design task of sketching a dentist logo to avoid design fixation effects (Jansson and Smith 1991, Smith et al. 1993). Connecting the dots yields a UFO piloted by aliens, elements which would be highly uncommon in a dentist logo.

3.4.3. Performance Measures

In the experiment, we measure various metrics to capture user behavior. Our focus is twofold. First, we study how much subjects explore the design space. For this purpose, we discuss how we measure similarities between designs and how we quantify idea variance in Section 3.4.3.1. Second, we measure design quality (Section 3.4.3.2) to assess the impact of sketching and idea variance on the final outcome of the design process.

3.4.3.1. Measuring Search Behavior

In order to quantify subjects' idea variance, we map the created designs to metric spaces such that we can measure distances and similarities. For robustness, we independently employ multiple vectorization techniques. First, we use

the term frequency-inverse document frequency algorithm (tf-idf, Sparck Jones 1972) from scikit-learn (Pedregosa et al. 2011). Let P be the set of all prompts (across all subjects) given to the image generation model and each prompt $D \in P$ be a multiset of words contained in the respective prompt. Let the vocabulary V be the set of words across all prompts, i.e. $V = \text{Set}\left(\bigcup_{D \in P} D\right)$. Then, we vectorize each prompt D as $x_{\text{tf-idf}} \in \mathbb{R}^{|V|}$, where each component of $x_{\text{tf-idf}}$ corresponds to a word $w \in V$ and is given as

$$\text{tf-idf}(w, D) = \frac{|\{\{x \mid x \in D, x = w\}\}|}{|D|} \cdot \log \frac{|P|}{|\{\{D' \mid D' \in P, w \in D'\}\}|}.$$

To increase the robustness of the algorithm, we use word stemming (Lovins 1968) by removing form specific endings from all words prior to applying tf-idf (e.g., *shine*, *shines* and *shining* are all be treated the same as they are stemmed towards the infinitive form).

The tf-idf algorithm has shortcomings however. As a count-based technique, it captures differences in word distributions but does not accurately reflect semantic similarities (e.g., synonyms are treated as different words just as antonyms are). Additionally, we can only use it to vectorize text-based prompts (i.e., the inputs of the design process) but not to vectorize images (outputs of the design process). To remedy these shortcomings, we also resort to a more advanced vectorization technique using neural networks. Specifically, we use a neural network $f_\theta \colon W \cup I \longrightarrow \mathbb{R}^n$ with parameters θ which projects prompts from the natural language space W or images from the image space I into an n-dimensional joint embedding space.[17] As the neural network, we use a pretrained CLIP (Radford et al. 2021) model, which is specifically trained contrastively, i.e. to maximize cosine similarities in the embedding space between similar inputs and minimize similarities for different inputs.

[17] Such neural network models are also referred to as vision-language models (VLMs) as the operate on both natural language and images.

We refer to Radford et al. (2021) for details. From CLIP vectorization, we obtain vectors $x_{\text{CLIP}} = f_\theta(p)$ for each prompt p and $x_{\text{CLIP-Image}} = f_\theta(o)$ for each design image o. We also remark that it is particularly interesting to analyze both the similarities of prompts as the inputs to the design process as well as the generated images as the outputs since there is a potential disconnect due to the stochasticity and performance limitations of the image generation models. Similar prompts can yield drastically different image outputs while conversely changes in the prompts not necessarily translate to the generated designs.

To quantify subjects' exploration behavior using the obtained vectors, we construct higher level metrics based on cosine similarities between the vectors. Due to the richness and complexity of designer behavior, we employ various metrics. Let $x_i^k,\ i = 1, \ldots, m$ be the vectors of the m designs created by subject k, obtained through any of the above vectorization techniques, and $\text{sim}(\cdot, \cdot)$ be the cosine similarity.

$$\textit{Mean Similarity}_k = \frac{2}{m(m-1)} \sum_{1 \le i < j \le m} \text{sim}(x_i^k, x_j^k)$$

is the mean pairwise similarity of the designs. It serves as our primary measure of process variance.

$$\textit{Trajectory Similarity}_k = \frac{1}{m-1} \sum_{i=1}^{m-1} \text{sim}(x_i^k, x_{i+1}^k)$$

is the mean pairwise similarity between immediately consecutive designs. Thereby, it measures how much subjects vary the design in each iteration.

$$\textit{Long Jumps}_k = |\{i \mid \text{sim}(x_i^k, x_{i-1}^k) \le t,\ i = 2, \ldots, m\}|$$

is the number of designs for which the similarity to the previous design falls below a threshold t. This is the number of iterations in which subjects diverged significantly from the previous design. *Clusters* is the number of clusters which a subject covers. We pool designs by all subjects and use hierarchical clustering based on cosine similarities and average linkage with a threshold c to assign a cluster to each design. $\textit{Clusters}_k$ measures the variety

of distinct design ideas that a subject explores. Finally, $\textit{First–Submitted}_k = \text{sim}(x_1^k, x_\omega^k)$ and $\textit{First–Last}_k = \text{sim}(x_1^k, x_m^k)$ are the similarities between the first design and respectively the final design selected by the subject in the competition and the last design created. Thus, we measure how distant the outcome of the design process is from the starting point. In our main analyses, we focus on metrics using text-based CLIP-vectorization since prompt-based similarity may matter most for user-driven process variance whereas image similarity is obscured by the image generation model's stochasticity. We set $t = 0.6$ for *Long Jumps* and $c = 0.4$ for *Clusters*. We report results for the other vectorization techniques and thresholds in Appendix E as robustness checks.

3.4.3.2. Measuring Design Quality Similar to previous work, we use crowd evaluation via non-experts to assess the quality of all generated designs (e.g., Girotra et al. 2010, Hooshangi and Loewenstein 2018, Jin and Chua 2024). We argue that this is especially suitable for our context of logo design as the target group of the (dentist) logos is the average person, i.e. the general crowd. We recruit independent subjects, who did not participate in the main experiment, through the online experiment platform Prolific. Raters receive a fixed compensation of approximately 1.5 pounds in line with the minimum wage standards on Prolific.

Following the evaluation procedures of the creators of the employed image generation models, we use pairwise comparisons between designs (Rombach et al. 2022, Ramesh et al. 2022, Saharia et al. 2022, Betker et al. 2023). This has several advantages over letting subjects rate each design independently on fixed scales. First, eliciting binary preferences is much easier for subjects than assigning absolute scores. Second, this procedure avoids any measurement scale-related issues or inconsistencies. Third, the pairwise comparisons allow subjects to rate designs much faster by simply clicking on the preferred design. Raters evaluate the logos blindly, i.e. they receive no additional information about the logo or its creator but merely see the image.

We calculate logo quality scores based on the elicited pairwise preferences via an Elo rating system (Elo and Sloan 1978). We pool all logos by all designers. Each logo design is

assigned an initial (Elo) quality score of 1,200. Ratings are updated as follows. Let r_a and r_b be the current ratings of logos a and b and $s_a \in \{0, 1\}$ be a binary indicator if the rater preferred a over b. Then, we obtain the new rating for a as

$$r_a^{\text{new}} = r_a + k * (s_a - e_a),$$

where k is the k-factor, which is typically set based on the number of comparisons a was already part of, and

$$e_a = \frac{1}{1 + 10^{(r_b - r_a)/400}}$$

is the apriori probability of a new rater preferring logo a over logo b based on the current ratings. We decay k from 40 to 20 over the course of evaluation as is standard in other contexts (e.g., chess).

Each rater assesses 250 randomly selected design pairs. Each design is assessed by multiple raters. Our pairing algorithm selects the first of the two designs for the next comparison in a round-robin fashion among all designs and across all raters, who work in parallel. This ensures that all designs are evaluated sufficiently many times. Now, let a be the first selected design and D be the set of all designs. Then, we select the second design b for the next comparison by drawing n candidate designs b_i, $i = 1, \ldots, n$ uniformly at random from $D \setminus \{a\}$ and choosing

$$b = \arg\min_{b_i} \left\{ |r_a - r_{b_i}| \,\middle|\, i = 1, \ldots, n \right\}.$$

This ensures randomness while aiding the convergence of the ratings by requesting comparisons among designs where the outcome is not yet clear. We set $n = 5$ in all experiments. We randomize the left-right positioning of the two designs in each comparison. For each designer's set of designs, we compute the *Mean Quality* across all their designs, the standard deviation (*Std Quality*), minimum (*Min Quality*), and maximum (*Max Quality*), as well as the score of the ultimately chosen design (*Quality Submitted*).

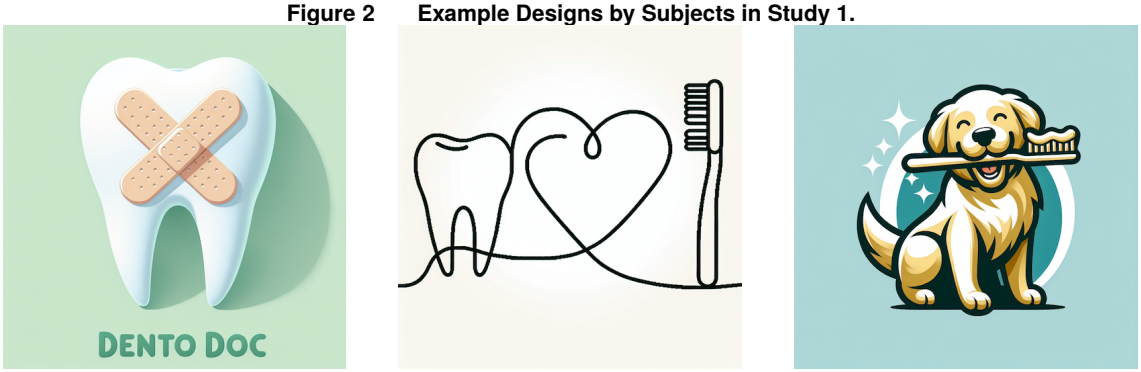
We exclude raters who fail multiple attention checks or show clear bias in their selection (e.g. always selecting the left design in each comparison). Specifically, we exclude raters who choose the left design in under 35% or over 65% of comparisons. Due to randomized left-right positioning, the probability of a rater preferring the left design is 50% in each comparison. Then, over the entire evaluation process of one rater, the number of times L that the left image is chosen is a binomially distributed random variable $L \sim B(n, 0.5)$, where n is the number of comparisons. For our selected thresholds of 35% and 65% and our number of comparisons $n = 250$, it follows that $P(0.35 \cdot 250 \leq L \leq 0.65 \cdot 250) > 0.999$. In other words, any rater falling outside these thresholds is highly unlikely due to chance and strongly suggests biased selections. Therefore, we exclude such raters.

3.5. Study 1

In Study 1, we investigate the interaction between sketching and generative AI for non-design experts in an incentivized and pre-registered online experiment[18] as outlined in the previous section. We manipulate whether the design process includes an initial sketching phase. We recruited 138 subjects from the participant pool of the economic research laboratory at a public German university, out of which 28 subjects did not complete the experiment and additional 3 subjects were excluded for generating only a single image (control: 1, treatment: 2). After exclusions our sample contains 107 subjects. All participants were university students and had no knowledge of the experiment contents before starting the experiment. No participant studies any design-related topic. All sessions took place in spring 2024. On average, subjects took 26 minutes to complete the experiment. Subjects in the experiment have limited prior experience with image generation models (*AI Experience*, controls: 1.96, treated: 1.76, as reported on a five-point Likert scale from 1 to 5). The difference between conditions is not significant ($p = 0.315$), indicating that our randomization was successful. In Figure 2 we present exemplary designs generated by subjects.

[18] https://aspredicted.org/X2V_3QR

Figure 2 Example Designs by Subjects in Study 1.



3.5.1. Design Fixation and Exploration Behavior

Here, our aim is to show how sketching impacts process variance and the exploration of the design space. For our analyses, we use two-tailed Welch's t-tests and OLS regressions controlling for *AI Experience*.

First, subjects in the treatment condition self-report a significantly higher degree of design fixation ($p = 0.014$) in the final surveys. They had a much clearer image in mind of what they wanted to created while using the image generation model, confirming that sketching indeed creates a design fixation. Despite this fixation, there is no significant difference in the number of designs that treated and controls create (treated: 12.1, controls: 12.9, $p = 0.692$). Sketching does not influence the length of the design process.

In the following, we provide evidence that the self-reported fixation also translates to a measurably narrower search behavior with less process variance as quantified by the metrics introduced in Section 3.4.3.1. Here, we focus on metrics based on CLIP-vectorization of the prompts. We present results for other vectorization techniques in Appendix E, which are inline with the results reported here. The *Mean Similarity* among the designs created by users who sketched is significantly higher than in the control condition ($p = 0.022$). This indicates that their idea variance is lower and, in conjunction with the indifference in the number of designs, means that they explored a smaller region of the design search space. Similarly, we find that treated subjects cover fewer *Clusters* ($p = 0.001$), meaning they explore fewer distinct design ideas. We also see reduced variance across consecutive design

Table 14 Study 1 – Designer Behavior: Treatment Effects.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Submitted*
Treatment (sketch = 1)	0.050** (0.020)	0.043** (0.019)	-0.808*** (0.227)	-0.969** (0.430)	0.080*** (0.028)
AI Experience	-0.006 (0.009)	-0.009 (0.009)	-0.081 (0.107)	0.058 (0.202)	-0.014 (0.013)
Num. Designs	-0.001 (0.001)	0.003*** (0.001)	0.054*** (0.011)	0.072*** (0.021)	-0.006*** (0.001)
Constant	0.743*** (0.027)	0.773*** (0.026)	2.233*** (0.301)	0.914 (0.572)	0.735*** (0.037)
Observations	107	107	107	107	107
Mean	0.738	0.809	2.570	1.449	0.679
Std.	0.106	0.104	1.565	2.353	0.16
R^2	0.082	0.119	0.274	0.148	0.208
Adjusted R^2	0.056	0.093	0.253	0.123	0.185

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

iterations as sketching has a significant positive effect on the mean similarity of consecutive designs (*Trajectory Similarity*, $p = 0.032$). Treated subjects change their prompts much less across iterations, i.e., they search more locally than the control condition. Not only is the average change across iterations lower, but sketching also reduces the number of times that subjects radically change their design compared to the previous one (*Long Jumps*, $p = 0.022$). Lastly, the end points of the design search process are also much closer to the starting points (i.e., the first generated design) when sketching. The similarity between the first designs and the designs ultimately selected by the subjects is significantly higher when sketching (*First–Submitted*, $p = 0.005$). All our results hold when controlling for *AI Experience* in the regressions in Table 14 ($p = 0.016$, $p = 0.027$, $p = 0.001$, $p = 0.027$, $p = 0.006$). In Appendix E, we provide additional robustness analyses confirming our results.

In Appendix F, we qualitatively illustrate the idea variance of the treatment and control condition by presenting examples. Treated subjects typically remain very close to their initial sketch during the entire design process. Even though the sketches are very minimalistic, the main design ideas within the sketches are often recognizable throughout the design iterations. Many sketched elements reappear in the generated designs. In contrast, the control condition diverges more across designs. Visually, we observe different design ideas and a less coherent search behavior. Note, that this does not imply an erratic search behavior

however. Controls still engage in local search and iterate on specific design ideas, just noticeably less so than the treated.

Using the survey data collected after the design process, we also study the self-perceived process variance. Subjects were asked to indicate on five-point Likert scales how much they varied their prompts throughout the design process and how much the generated designs differed from each other. Notably, while perceived variation and *Mean Similarity* is negatively correlated for both prompts ($\rho = -0.305$, $p = 0.001$) and images ($\rho = -0.195$, $p = 0.044$), we find no difference in the perceived variation between treated and controls (prompts: $p = 0.952$, images: $p = 0.425$). For the images, perceived variation is on average even higher when sketching contrary to our findings regarding objectively measured variation. This indicates discrepancies in the designers' perception of their own behavior and the actual design process and suggests sketching designers might unconsciously reduce process variance.

Finally, we also investigate whether subjects are affected asymmetrically based on prior experience with image generation models as prior experience has been found to moderate effects of generative AI in other contexts (e.g., Brynjolfsson et al. 2023, Noy and Zhang 2023, Dell'Acqua et al. 2023). For our analysis, we partition subjects along the median of *AI Experience* (e.g., Dell'Acqua et al. 2023, Lehmann et al. 2024). We show the results in Table 15. Only for *First–Submitted* do we find heterogeneous effects. Prior experience increases the distance between first and submitted design ($p = 0.029$) for treated subjects and thereby induces more divergence of the final outcome from the starting point, offsetting the effects of sketching.

We summarize the findings from this section as follows. Sketching prior to an iterative design process with AI reduces the design process variance. Sketching designers have a clearer target design in mind. As a result, they explore fewer distinct design ideas, incorporate fewer radical design changes, and instead search more locally around the design search space area corresponding to their sketch. Sketching appears to result in design fixations and consequentially reduced idea variance, thus supporting Hypothesis 4.

Table 15 Study 1 – Designer Behavior: Heterogeneous Effects.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Submitted*
Treatment × *AI Exp.* > Median	-0.060 (0.048)	-0.043 (0.046)	0.717 (0.542)	-0.536 (1.031)	-0.146** (0.066)
Treatment (sketch = 1)	0.061*** (0.023)	0.052** (0.022)	-0.956*** (0.258)	-0.829* (0.491)	0.112*** (0.032)
AI Experience > Median	-0.009 (0.031)	-0.017 (0.030)	-0.323 (0.355)	0.567 (0.675)	0.023 (0.043)
Num. Designs	-0.001 (0.001)	0.003*** (0.001)	0.054*** (0.011)	0.073*** (0.021)	-0.005*** (0.001)
Constant	0.732*** (0.021)	0.759*** (0.020)	2.173*** (0.238)	0.859* (0.454)	0.700*** (0.029)
Observations	107	107	107	107	107
Mean	0.738	0.809	2.570	1.449	0.679
Std.	0.106	0.104	1.565	2.353	0.16
R^2	0.110	0.138	0.282	0.153	0.247
Adjusted R^2	0.076	0.105	0.254	0.120	0.217

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

3.5.2. Effects on Design Quality

In our study, sketching reduces idea variance. However, it is yet unclear what this means for the quality of the generated designs. We shed more light on this question using crowd evaluations following Section 3.4.3.2. We recruited 160 subjects as raters from Prolific. Of these, 8 raters were excluded for failing multiple attention checks and an additional 5 for showing clearly biased selections (see our discussion in Section 3.4.3.2). The remaining 147 raters rated a total of 34 457 design pairs, taking 17.6 minutes on average. In Appendix F, we visualize the results of the rating procedure by presenting the quality scores of exemplary designs.

Clear differences in the quality of the designs are apparent as the treated subjects generate designs with a significantly higher *Mean Quality* than controls (treatment: 1207.1, controls: 1154.3, $p = 0.003$). Sketching results in better designs on average. Likewise, sketching also increases the quality of both the best created design per subject (*Max Quality*, treatment: 1329.8, controls: 1282.4, $p = 0.037$) as well as the quality of the selected design (*Quality Submitted*, treatment: 1243.4, controls: 1196.0, $p = 0.037$). Note that these are not necessarily identical as the designers' own evaluation may differ from the crowd evaluation. We find no difference in the quality variance or the minimum quality ($p = 0.816$, $p = 0.092$). In regressions in Table 16, we show that findings are identical when controlling for *AI Experience* (*Mean Quality*: $p = 0.001$, *Quality Std*: $p = 0.751$, *Max Quality*: $p = 0.008$, *Min*

Table 16 Study 1 – Design Quality: Treatment Effects.

	Mean Quality	*Std Quality*	*Max Quality*	*Min Quality*	*Quality Submitted*
Treatment (sketch = 1)	56.730*** (16.534)	2.115 (6.657)	52.833*** (19.585)	32.362* (18.853)	53.036** (21.962)
AI Experience	9.561 (7.773)	0.552 (3.130)	3.911 (9.207)	7.142 (8.863)	17.514* (10.325)
Num. Designs	2.463*** (0.804)	0.556* (0.324)	5.782*** (0.953)	-2.481*** (0.917)	2.502** (1.068)
Constant	1103.773*** (21.956)	75.931*** (8.840)	1200.092*** (26.007)	1039.414*** (25.034)	1129.323*** (29.163)
Observations	107	107	107	107	107
Mean	1180.478	84.96	1305.888	1037.682	1219.474
Std.	91.545	34.077	117.588	100.138	117.541
R^2	0.170	0.028	0.294	0.098	0.111
Adjusted R^2	0.145	0.000	0.273	0.071	0.085

Note. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 17 Study 1 – Design Quality: Heterogeneous Effects.

	Mean Quality
Treatment × *AI Experience* > Median	-31.700 (39.903)
Treatment (sketch = 1)	62.149*** (19.012)
AI Experience > Median	15.887 (26.118)
Num. Designs	2.480*** (0.813)
Constant	1117.913*** (17.566)
Observations	107
Mean	1180.478
Std.	91.545
R^2	0.163
Adjusted R^2	0.130

Notes. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Quality: $p = 0.089$, *Quality Submitted*: $p = 0.018$). The advantage through sketching is also visible on the design contest-level. When pooling the designs of all subjects, 80.3% of the top 5% of logos and 8 out of the 10 best designs were created when sketching. Out of the top 10% of designers (as measured by the *Quality Submitted*), 60.0% sketched. We find no evidence for heterogeneous effects on quality with respects to prior *AI Experience* (Table 17).

As eluded to earlier, the subjects' own perception of quality may differ from the crowd evaluation. In the post-experimental survey, we collect the subjects' own evaluation of their designs, both on average and of their selected design, their overall satisfaction with the

design process as well as the perceived adherence of the image generation model to the prompts. Interestingly, while sketching has a (strongly) significant effect on *Mean Quality*, *Max Quality*, and *Quality Submitted*, there is no significant difference in the respective perceived measures (*Perceived Mean Quality*: $p = 0.097$, *Perceived Max Quality*: $p = 0.332$). Even more surprisingly, the effect direction is actually reversed, i.e., on average treated subjects like their own designs less than the control condition (treated: 3.132, controls: 3.500). This lower satisfaction extends to both the perceived adherence and the overall satisfaction, which are lower, albeit statistically insignificant ($p = 0.132$, $p = 0.284$), in the treatment condition. To further investigate the relation between own evaluations and (objective) crowd ratings, we also conduct a correlation analysis finding that evaluations are uncorrelated for both for the mean ($\rho = -0.101$, $p = 0.301$) and the max rating ($\rho = 0.053$, $p = 0.591$) per subject. Despite their better objective performance, sketching designers do not perceive their design to be superior and, if anything, tend to dislike their designs and the process more.

In conclusion, we find that sketching increases the quality of AI-generated designs. This applies to all design iterations as well as the final output of the design process, i.e., the final design selected by the designer. Strikingly, quality rises despite reduced idea variance (see Section 3.5.1). Thereby, our results break the classic variance-quality assumption (Girotra et al. 2010), highlighting that focused ideation can surpass broader exploration.

3.6. Study 2

In Study 1, we studied the effects of sketching on AI design processes using non-design experts. In Study 2, we replicate the exact experiment from Study 1 following the experimental procedures outlined in Section 3.4 but now recruited prospective design experts. Study 2 too is incentivized and pre-registered[19]. Subjects had no knowledge of the experiment content before starting the experiment. We recruited subjects from two sources, from Prolific and from a public German design school. On Prolific, we recruited 300 subjects who are

[19] https://aspredicted.org/nt58-tby5.pdf

Figure 3 Example Designs by Subjects in Study 2.

currently or have finished studying any form of design. At the design school, we recruited 37 subjects, who were all currently enrolled students (graduate and undergraduate) studying design. We exclude subjects based on the same filters as in Study 1. We exclude 188 subjects (all Prolific) who did not finish the experiment and 10 subjects (Prolific: 8, design school: 2) who only created a single design. The final subject pool consists of 139 (104 + 35) subjects. On average, subjects took 21 minutes to complete the experiment.

Subjects in Study 2 too have limited prior experience with image generation models (controls: 2.826, treated: 2.986). The difference between conditions is not significant ($p = 0.443$), indicating that our randomization was successful. The increase in *AI Experience* compared to Study 1 may be for two reasons. On the one hand, designers may be particularly inclined to use AI tools as they directly relate to their profession. On the other hand, Study 2 took place several months after Study 1 such that part of the increase may also be due to the increasing adoption rates of AI in general. Figure 3 displays exemplary designs created in Study 2.

3.6.1. Design Fixation and Exploration Behavior

In the survey, treated subjects indicate a significantly higher degree of design fixation than controls ($p < 0.001$). While this design fixation does not yield a difference in the iteration length (*Number of Designs*, $p = 0.216$, controls 10.0, treated 8.4), it significantly impacts the idea variance during the design process. The design space explored by treated subjects is smaller and denser (*Mean Similarity*, $p = 0.002$, regression: $p = 0.003$). Sketching also

Table 18 Study 2 – Designer Behavior: Treatment Effects.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Submitted*
Treatment (sketch = 1)	0.066*** (0.022)	0.058*** (0.022)	-0.619** (0.253)	-0.700** (0.270)	0.098*** (0.029)
AI Experience	0.005 (0.009)	-0.003 (0.009)	0.026 (0.104)	0.021 (0.111)	0.011 (0.012)
Num. Designs	-0.001 (0.001)	0.004*** (0.001)	0.059*** (0.017)	0.059*** (0.018)	-0.006*** (0.002)
Constant	0.704*** (0.032)	0.739*** (0.032)	2.052*** (0.367)	0.774* (0.392)	0.698*** (0.041)
Observations	139	139	139	139	139
Mean	0.746	0.797	2.353	1.022	0.723
Std.	0.131	0.133	1.568	1.668	0.180
R^2	0.069	0.095	0.135	0.130	0.161
Adjusted R^2	0.048	0.075	0.116	0.111	0.142

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

results in exploring fewer distinct design ideas as treated subjects cover significantly fewer *Clusters* ($p = 0.008$, regression: $p = 0.016$). Design changes across consecutive iterations are smaller when sketching, as the mean similarity between consecutive iterations is significantly higher in the treatment condition (*Trajectory Similarity*, $p = 0.025$, regression: $p = 0.009$), and the number of *Long Jumps*, i.e. larger design changes, is significantly smaller ($p = 0.005$, regression: $p = 0.011$). This local search behavior is also apparent from comparing the start and end points of the process. In the treatment condition, the first generated design is significantly more similar to the submitted design than in the control condition (*First–Submitted*, $p < 0.001$, regression: $p = 0.001$). Overall, we find evidence that sketching limits how much professional designers explore different ideas when using AI, thus again supporting Hypothesis 4. In Appendix E, we provide additional robustness analyses confirming our results.

As in Study 1, we also find qualitative anecdotic evidence of design fixations in the treatment group by visualizing sketches and search trajectories in Appendix F. We see clear continuations of the design ideas in the sketches in the subsequently generated designs. Divergence from this outlined path during the design process is very limited. Although the control condition too exhibits local search behavior, we find more concrete changes across design iterations and see that these subjects explore more distinct design ideas.

Looking at the self-perceived process variance among designers, we find that *Perceived*

Table 19 Study 2 – Designer Behavior: Heterogeneous Effects.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Submitted*
Treatment × AI Exp. > Median	-0.032 (0.045)	-0.021 (0.045)	1.271** (0.511)	0.758 (0.555)	-0.085 (0.059)
Treatment (sketch = 1)	0.078*** (0.027)	0.064** (0.027)	-1.073*** (0.309)	-0.970*** (0.336)	0.131*** (0.036)
AI Experience > Median	0.002 (0.032)	-0.022 (0.032)	-0.408 (0.362)	-0.250 (0.393)	0.047 (0.042)
Num. Designs	-0.000 (0.001)	0.004*** (0.001)	0.061*** (0.016)	0.060*** (0.018)	-0.006*** (0.002)
Constant	0.716*** (0.024)	0.737*** (0.023)	2.258*** (0.266)	0.915*** (0.289)	0.710*** (0.031)
Observations	139	139	139	139	139
Mean	0.746	0.797	2.353	1.022	0.723
Std.	0.131	0.133	1.568	1.668	0.180
R^2	0.073	0.110	0.177	0.143	0.169
Adjusted R^2	0.046	0.083	0.153	0.117	0.144

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Variation and *Mean Similarity* are no longer negatively correlated for prompts ($\rho = -0.163$, $p = 0.056$) or images ($\rho = -0.048$, $p = 0.582$). As for non-designers (Study 1), we do not find a difference in perception between treated and controls (prompts: $p = 0.558$, images: $p = 0.902$) despite the earlier found, objectively quantified differences in behavior. The effects of sketching on search behavior appear to be subconscious.

For the majority of our metrics, we find no evidence for heterogeneous effects regarding prior experience with image generation models (Table 19). *AI Experience* only mitigates the effects of sketching on the number of distinct design ideas (*Clusters*, $p = 0.034$). Beyond this, *AI Experience* does not moderate the effect on behaviors for experienced designers.

We conclude that for experienced designers the same effects on idea variance occur as observed for design novices in Study 1. Sketching before using AI creates design fixations and subsequently inhibits the exploration of diverse ideas. Instead, designers who sketch focus on local variations around their sketch.

3.6.2. Effects on Design Quality

Here, we study quality implications for professional designers due to the reduced idea variance induced by sketching. For the crowd evaluation (see Section 3.4.3.2), we recruited 120 raters from Prolific. 3 raters were excluded for failing multiple attention checks and 2 for making biased selections (see our discussion in Section 3.4.3.2). In total, we collected

Table 20 Study 2 – Design Quality: Treatment Effects.

	Mean Quality	*Std Quality*	*Max Quality*	*Min Quality*	*Quality Submitted*
Treatment (sketch = 1)	64.143*** (22.201)	-8.679 (10.815)	46.411** (23.065)	75.378*** (26.477)	78.729*** (29.637)
AI Experience	-7.261 (9.119)	2.316 (4.442)	-3.003 (9.474)	-2.176 (10.875)	0.590 (12.173)
Num. Designs	1.149 (1.447)	-0.121 (0.705)	5.374*** (1.503)	-3.994** (1.726)	0.596 (1.932)
Constant	1175.336*** (32.178)	123.595*** (15.676)	1297.160*** (33.430)	1030.022*** (38.376)	1203.170*** (42.957)
Observations	139	139	139	139	139
Mean	1197.062	124.848	1361.062	1025.045	1250.000
Std.	131.99	62.489	140.227	160.855	175.141
R^2	0.062	0.006	0.103	0.101	0.050
Adjusted R^2	0.041	-0.016	0.083	0.081	0.029

Note. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

28 209 pairwise comparisons. Raters took 21.6 minutes on average. We present exemplary quality scores in Appendix F.

Similarly to Study 1, we find that the *Mean Quality* of designs is significantly higher when sketching (treatment: 1227.4, controls: 1166.3, $p = 0.006$, regression: $p = 0.005$). For experienced designers too, sketching improves the *Max Quality* (treatment: 1379.5, controls: 1342.3, $p = 0.121$, regression: $p = 0.046$) as well as the *Quality Submitted* (treatment: 1288.6, controls: 1210.8, $p = 0.009$, regression: $p = 0.009$). In contrast to Study 1, the increased quality through sketching is now also visible in the worst design per subject, where sketching raises the bottom line by increasing the *Min Quality* score ($p = 0.003$, regression: $p = 0.005$). The quality variance remains the same in the treatment and control conditions for experienced designers ($p = 0.447$, regression: $p = 0.424$). On contest-level, the benefits of sketching are not as visible as in Study 1. While still 55.6% of the top 5% of logos stem from sketching designers, only 3 out of the 10 best logos do. Yet, 69.25% of the top 10% of designers sketched. In conclusion, sketching improves the outputs of AI-powered design processes also for professional designers despite the decrease in idea variance.

When looking at the designers' own perception of quality, we find similar results as in Study 1. Despite having significant positive effects on crowd-determined quality, there is no difference in the average perceived quality between the treatment and control conditions

Table 21 Study 2 – Design Quality: Heterogeneous Effects.

	Mean Quality
Treatment × AI Experience > Median	78.311* (44.873)
Treatment (sketch = 1)	33.749 (27.152)
AI Experience > Median	-84.555*** (31.750)
Num. Designs	1.392 (1.417)
Constant	1184.249*** (23.340)
Observations	139
Mean	1197.062
Std.	131.99
R^2	0.105
Adjusted R^2	0.078

Notes. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

(mean own rating: $p = 0.732$, max own rating: $p = 0.811$). For designers, sketching has no influence on the overall satisfaction with the process ($p = 0.303$) or the perceived adherence of the model to the prompts ($p = 0.143$). Designers do not perceive the advantage with respects to quality that sketching actually gives them.

3.7. Exploratory Analysis: Why Does Sketching Improve Quality?

Building on our main results, we provide additional post-hoc analyses in this section in an effort to further explain our findings and unravel the relationships between sketching, process variance and design quality.

Across Studies 1 and 2, we have shown that sketching improves the average and maximum quality of designs despite its negative effect on idea variance. This finding is in contrast with ideation literature which generally sees variance during ideation and design processes as beneficial (see our discussion in Section 3.3). In this section, we explore potential ex-post explanations for the divergence between our findings and theory. We begin by ruling out two obvious lines of argumentation before then presenting evidence for potential mechanisms. We will conclude by jointly regressing quality on the respective explanatory variables.

3.7.1. Learning Effects

One potential explanation is differential learning. Sketching induces less exploratory global search and instead designers engage in more local search. As they adjust prompts step-by-step, rather than trying completely new ones, they may receive a much clearer learning signal of how to prompt the image generation model successfully to yield the desired design. This argumentation is consistent with search literature where local search in small steps is beneficial to reveal the search landscape (e.g., Billinger et al. 2014). To investigate learning effects, we compute linear trends of the quality scores across iterations for each participant. We find no evidence that sketching influences the slope of quality improvement across iterations (Study 1: $p = 0.818$, Study 2: $p = 0.611$). To isolate learning effects through local search, we similarly fit linear approximations of quality across iterations within each design cluster per participant. The average within-cluster slope does not differ by condition (Study 1: $p = 0.172$, Study 2: $p = 0.378$). Interestingly, we do not find any evidence of learning through local search in Study 1 as the mean linear trends within clusters is negative for both treated and controls. Finally, we note that two factors may limit learning effects in our experiment. Firstly, participants do not iterate much. Within only about ten iterations on average, the amount of learning that can occur is naturally limited. Secondly, the stochasticity of the image generation models may obscure improvements in prompts such that the learning signal that participants receive is distorted.

3.7.2. Disconnect Between Process and Outcome Variance

Theory suggests that process variance (i.e., idea variance) is desirable in ideation processes as it promotes outcome variance and thus the likelihood of exceptional results. This may not hold in our setting if process and outcome variance are disconnected due to the stochasticity of the image generation models. For example, even with identical prompts (i.e. no process variance), the quality of the outputs can vary significantly. To test the relationship between process and outcome variance, we estimate the effect of our primary exploration behavior metric *Mean Similarity* on *Std Quality* in regressions in Table 22. The effect of our process

Table 22 Process & Outcome Variance.

	Std Quality	
	Study 1	Study 2
Treatment (sketch = 1)	8.167 (6.391)	3.111 (10.442)
Mean Similarity	-122.236*** (30.310)	-179.046*** (39.589)
AI Experience	-0.237 (2.927)	3.185 (4.158)
Num. Designs	0.381 (0.305)	-0.222 (0.659)
Constant	166.748*** (23.983)	249.587*** (31.478)
Observations	107	139
R^2	0.162	0.138
Adjusted R^2	0.129	0.112

Note. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

measure on the outcome measure is negative and strongly significant (Study 1: $p < 0.001$, Study 2: $p < 0.001$) suggesting that process variance does translate to outcome variance. Thus, we reject this line of argumentation.

3.7.3. Better Starting Point

Sketching is a thoughtful and deliberate process, during which designers explicitly contemplate their design preferences. This may result in designers choosing a better starting point in the rugged design landscape compared to when directly jumping into working with the AI without any planning. In Study 2, sketching yields a significantly higher quality on the first AI-generated design ($p = 0.012$, regression: $p = 0.009$, Table 23). Professional designers who sketch are better right from the start. In Study 1, the difference trended positive but did not reach significance ($p = 0.142$, regression: $p = 0.136$, Table 23). Thus, sketching may enable finding a better initial design from which the search then branches off.

3.7.4. Generative AI Distracts From The Task

Based on our exploratory data analysis, we posit an additional explanation. Generative AI might have distracting effects while sketching-induced design fixation safeguards designers against these. The strong capabilities of AI enable users to generate designs beyond their own skills, opening vast new areas of the design space. This can result in off-task exploration: designers testing the capabilities of the AI rather than focusing on the actual

Table 23 Quality of the First Design.

	Quality First	
	Study 1	Study 2
Treatment (sketch = 1)	33.787 (22.513)	78.900*** (29.644)
AI Experience	-1.045 (10.584)	-12.844 (12.176)
Num. Designs	1.298 (1.095)	1.108 (1.932)
Constant	1111.748*** (29.895)	1169.057*** (42.966)
Observations	107	139
R^2	0.034	0.055
Adjusted R^2	0.006	0.034

Note. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

task. Specifically, we observe that participants occasionally produce designs that do not fit the design requirements, i.e. they generate images which cannot be considered dentist logos (e.g., photorealistic scenes or non-dentist-related contents; see Figure 4 for examples) and consequentially are rated poorly. This effect is similar to choice overload in psychology (e.g., Chernev et al. 2015), where an abundance of options (here: AI-enabled design subspaces) can have adverse effects. In our analysis in Table 24, we classify each image, blind to condition, as 'logo-like' or not and find that sketching tends to reduce the amount of non-logo outputs from 26.9 % to 16.9 % in Study 1 ($p = 0.098$, regression: $p = 0.055$) and from 24.6 % to 11.3 % in Study 2 ($p = 0.004$, regression: $p = 0.002$). This suggests a twist on our earlier results. Rather than just process variance, our metrics may capture both the known-to-be-useful productive variance as well as a distracting variance, which induces a loss of focus in the control group. Only considering the logo-designs, the treatment condition still generates better designs, albeit not quite statistically significant in Study 2 (Study 1: $p = 0.006$, Study 2: $p = 0.077$).

3.7.5. Synopsis

Having outlined potential explanations, we quantify these effects via a joint regression analysis in Table 25. Choosing a better starting point (*Quality First*), as facilitated by sketching, has a significant positive effect on *Mean Quality* (Study 1: $p < 0.001$, Study 2: $p < 0.001$) and *Max Quality* (Study 1: $p < 0.013$, Study 2: $p < 0.001$). We find that local

Figure 4 **Examples of Outputs Not Classified as Dentist Logos.**

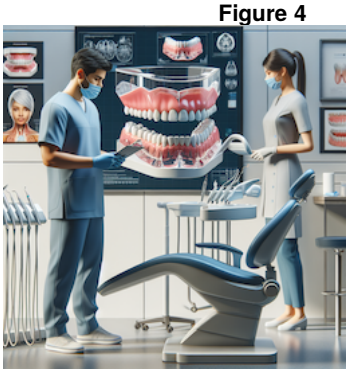
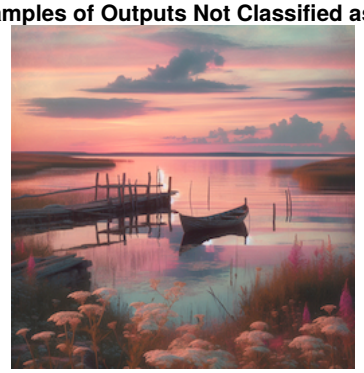
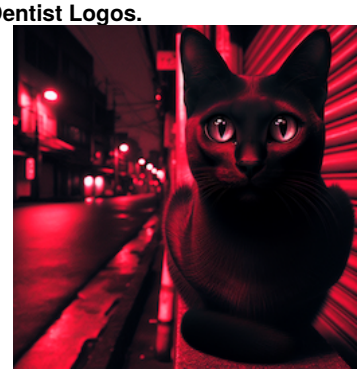




Table 24 **Treatment Effect on Task Focus.**

	Percentage of Logos	
	Study 1	Study 2
Treatment (sketch = 1)	0.115* (0.059)	0.145*** (0.045)
AI Experience	0.049* (0.028)	-0.018 (0.019)
Num. Designs	0.006** (0.003)	0.006* (0.003)
Constant	0.561*** (0.078)	0.747*** (0.065)
Observations	107	139
R^2	0.087	0.089
Adjusted R^2	0.060	0.069

Notes. S2. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

learning, measured by average within-cluster slopes as introduced earlier, only affects *Mean Quality* in Study 2 (*Local Improvements*, Study 1: $p = 0.236$, $p = 0.783$; Study 2: $p < 0.001$, $p = 0.067$). Also recall our earlier findings that sketching does not induce learning. Losing focus, manifesting in participants creating non-logo images, has a negative effect on quality (Study 1: $p < 0.001$, $p < 0.001$; Study 2: $p < 0.001$, $p < 0.001$). Finally, the effect of our treatment largely vanishes when including first-design quality and non-logo proportion (Study 1: $p = 0.011$, $p = 0.073$, Study 2: $p = 0.826$, $p = 0.340$), suggesting these factors mediate much of the treatment effect. Sketching may improve quality primarily by yielding a better starting point and preventing distracting effects of AI in form of off-task explorations.

Table 25 Design Quality: Explanatory Analysis.

	Mean Quality		*Max Quality*	
	Study 1	Study 2	Study 1	Study 2
Treatment (sketch = 1)	29.877** (11.481)	-3.083 (14.024)	29.936* (16.543)	-17.355 (18.104)
AI Experience	3.116 (5.347)	1.911 (5.540)	-3.007 (7.705)	6.016 (7.152)
Num. Designs	1.310** (0.553)	-0.197 (0.893)	4.691*** (0.797)	3.955*** (1.153)
Logo Percentage	135.926*** (21.015)	166.677*** (30.447)	152.289*** (30.281)	171.212*** (39.305)
Local Improvements	0.049 (0.041)	0.260*** (0.058)	-0.016 (0.059)	0.139* (0.075)
Quality First	0.286*** (0.055)	0.441*** (0.049)	0.200** (0.079)	0.305*** (0.063)
Constant	711.365*** (58.251)	538.220*** (52.810)	889.986*** (83.935)	816.634*** (68.173)
Observations	106	133	106	133
R^2	0.642	0.663	0.549	0.468
Adjusted R^2	0.620	0.647	0.522	0.443

Notes. S2. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

3.8. Discussion

Should ideators frontload mental engagement (e.g., by sketching) before AI-augmented ideation? Across two pre-registered and incentivized experiments, we unravel the effects of sketching prior to using generative AI on designer behavior and outcomes. We find that sketching significantly reduces designers' design idea variance, thereby mitigating a core benefit of this new technology to rapidly produce varieties of designs. Although low idea variance is, in theory, undesirable (Girotra et al. 2010), we further show that sketching has a strong positive effect on the design quality despite reducing idea variance. Designers who sketch both deliver a better final design and produce better designs in the process. The effects are consistent across levels of expertise. Amateurs as well as prospective design professionals try fewer different designs and yet benefit in terms of quality when sketching.

In exploratory analyses, we propose that these benefits of sketching may be linked to cognitive fixation effects. On the one hand, sketching is a thoughtful and deliberate process. We provide evidence that frontloading mental engagement through sketching, rather than jumping into design generation without preparation, yields a better starting point for the design process. On the other hand, we find that the design fixation on the sketch, although limiting idea variance, increases focus on the specific task. Without sketching, the unbound

capabilities of generative AI appear to distract designers who in turn produce more designs which do not meet the design specifications. We find no evidence for learning effects due to more local search.

Our results have several theoretical implications for our understanding of ideation processes, particularly in the presence of generative AI. First, our results challenge classic design thinking and ideation theory, which posits that greater idea variance increases output quality (Brown et al. 2008, Girotra et al. 2010). In AI-augmented contexts, however, process variance does not always translate to improved results and can incur distraction costs, which may even diminish quality. Sketching provides an anchor that narrows ideation yet improves outcomes.

Second, our results suggest that there is a fine line between exploration and loss of focus in ideation processes. Exploratory behavior by ideators can lead to tapping into irrelevant ideas outside of the given constraints. We posit that this is especially relevant when using generative AI for ideation since AI does not intrinsically adhere to such constraints and instead can broaden the ideation space. This finding also relates to search literature, where it is established that humans are prone to breaking off local search to early (i.e. lose focus) and rather over-explore less relevant regions in the search space (Billinger et al. 2014) – a human fallacy that may be amplified by generative AI.

Third, our results further suggest that design fixation, i.e. anchoring, can yield a beneficial focus of efforts on the task. Fixation – while traditionally having negative connotations in fixation literature (e.g., Jansson and Smith 1991, Smith et al. 1993, Youmans and Arciszewski 2014, Lee and Sosa 2024) – can be useful when the initial anchor is chosen deliberately through reflection (e.g., via sketching), rather than through random ("bad") stimuli which derail focus. This nuance contributes to fixation theory: fixation is not inherently harmful but depends on the quality and relevance of the anchor.

Fourth, design literature is yet unclear about the impact of sketching, providing mixed evidence on its effect on quality (Goldschmidt 1991, Song and Agogino 2004, Jonson

2005, Bilda et al. 2006). We find that sketching improves the design process outcomes. In addition, we shed light on potential mechanisms that induce this improvement. Frontloading of mental engagement through sketching enables initiating the design search process from a better starting point. Simultaneously, frontloading also seems to prevent the aforementioned phenomenon of breaking off local search too early in favor of exploration.

In practice, generative AI has started to transform design processes. Designers begin to tap into the potential of these models to accelerate previously manual steps and make use of their exploration abilities. Yet, it remains unclear how to best structure and sequence the activities in AI-augmented design processes. Our experiments reveal that as these transformations occur, designers should continue to sketch as part of the process. Despite the prima facie potential of generative AI to replace sketching, we show that latent effects of sketching have significant positive impact on design quality – even under the presence of AI. This implication also applies to a broader innovation management audience: Managers should consider preserving certain traditional steps in ideation processes even when deploying generative AI. Particularly, planning steps before AI-driven ideation can help unlock the potential of AI without unwanted detrimental effects on quality.

However, our findings also highlight that sketching may be double-edged: While sketching generally benefits quality, practitioners must be aware of the fixating effects induced by the initial sketches. This is especially relevant as our findings suggest these fixations to be subconscious. Thus, we expand upon common practical design guidelines which caution against design fixation through external stimuli. We believe that this knowledge should be incorporated into standard design trainings. Designers and educators should be aware of both the focusing advantages and the risk of tunnel vision into training and practice, for instance by combining initial sketches with deliberate divergence exercises.

Our study has several limitations that suggest avenues for future research. First, we focus on a logo-design task with a simple mouse-based sketch tool and DALL·E 3; results may differ for more complex design briefs (e.g., product design, architecture etc.), richer

sketch media, or other modalities, i.e. non-visual ideation domains with large language models as the generative AI. Logo design is a bounded, visual task. In more open-ended or strategic ideation (e.g., business-model brainstorming), the balance between planning and AI suggestions may differ. It remains an open question, how to structure AI-augmented non-visual ideation processes. We encourage future research to investigate interactions between AI-usage and traditional process steps in these domains. We also note that mouse-based simplistic sketching may not replicate professional sketch practices. In real settings, richer sketching (e.g., pen/tablet) might yield different anchoring strength. However, we argue that better sketches will result in even stronger anchors, thus making our observed effects conservative estimates. Second, we rely on crowd-based quality ratings rather than expert or stakeholder evaluations, which may not capture all dimensions of design success. We especially remark that process variance may still be relevant despite not inducing increases in crowd-based ratings. Real world clients often have hidden preferences which may or may not align with crowd preferences such that exploration is necessary to uncover these. Third, our time-limited setting only allows capturing immediate behavior. As designers adopt AI tools, they might learn and adapt strategies to use AI effectively without sketching. Longitudinal studies are required to observe such long-term effects over repeated projects.

Acknowledgements

My sincere gratitude goes to all the people whose support made this thesis possible. First and foremost, I thank my supervisor, Professor Fabian Sting, for giving me the opportunity to pursue doctoral studies and for the continuous support, guidance and freedom throughout. I am also very grateful to have had the opportunity to work with my co-authors, Professor Henrik Franke and Professor Philipp Cornelius, who helped shape my research into its final form, which you are here reading. To my fellow PhD students – Eva, Felix, Joe, Johannes and Timo – thank you for the camaraderie, shared frustrations, and laughter that made the journey very enjoyable. Many thanks go to Dominik for all the insightful discussions and for walking the largest part of my academic journey with me. Thank you to Hannah, Nele and Reem for lending their infinite dental wisdom and serving as jury for the logo competition. To my parents, who I could always lean on and who happily tested early experiment versions. To my brother, Max, and maybe even to my sister, Anne. A big thanks also to Gearoid Murphy, Feargus Pendlebury and Tam Hoang, who helped to make things right and led my academic journey to an unforgettable ending. Finally, all this would not have been possible without Valerie. Her undying support on and off the pitch is unparalleled. Thank you.

References

Abbasi S, Kazi H (2014) Measuring effectiveness of learning chatbot systems on student's learning outcome and memory retention. *Asian Journal of Applied Science and Engineering* 3(2):251–260.

Acemoglu D, Restrepo P (2018) Artificial intelligence, automation, and work. *The economics of artificial intelligence: An agenda*, 197–236 (University of Chicago Press).

Ait Baha T, El Hajji M, Es-Saady Y, Fadili H (2023) The impact of educational chatbot on student learning experience. *Education and Information Technologies* 1–24.

AlphaCode Team G (2023) Alphacode 2 technical report. Technical report, URL https://storage.googleapis.com/deepmind-media/AlphaCode2/AlphaCode2_Tech_Report.pdf.

Arrow KJ (1962) The economic implications of learning by doing. *The Review of Economic Studies* 29(3):155–173.

Bastani H, Bastani O, Sungu A, Ge H, Kabakcı Ö, Mariman R (2024) Generative ai can harm learning. *Available at SSRN 4895486* .

Bell JJ, Pescher C, Tellis GJ, Füller J (2024) Can ai help in ideation? a theory-based model for idea screening in crowdsourcing contests. *Marketing Science* 43(1):54–72.

Betker J, Goh G, Jing L, Brooks T, Wang J, Li L, Ouyang L, Zhuang J, Lee J, Guo Y, et al. (2023) Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf* 2(3):8.

Bilda Z, Gero JS, Purcell T (2006) To sketch or not to sketch? that is the question. *Design studies* 27(5):587–613.

Billinger S, Stieglitz N, Schumacher TR (2014) Search on rugged landscapes: An experimental study. *Organization Science* 25(1):93–108.

Bishop J (2006) Drinking from the fountain of knowledge: Student incentive to study and learn–externalities, information problems and peer pressure. *Handbook of the Economics of Education* 2:909–944.

Boussioux L, Lane JN, Zhang M, Jacimovic V, Lakhani KR (2024) The crowdless future? generative ai and creative problem-solving. *Organization Science* 35(5):1589–1607.

Bowman-Perrott L, Davis H, Vannest K, Williams L, Greenwood C, Parker R (2013) Academic benefits of peer tutoring: A meta-analytic review of single-case research. *School Psychology Review* 42(1):39–55.

Bray RL (2024) A tutorial on teaching data analytics with generative ai. *arXiv preprint arXiv:2411.07244* .

Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, et al. (2020) Language models are few-shot learners. *Advances in neural information processing systems* 33:1877–1901.

Brown T, et al. (2008) Design thinking. *Harvard business review* 86(6):84.

Brynjolfsson E, Li D, Raymond L (2025) Generative ai at work. *The Quarterly Journal of Economics* 140(2):889–942.

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

Buchanan R (1992) Wicked problems in design thinking. *Design issues* 8(2):5–21.

Buell RW (2021) Last-place aversion in queues. *Management Science* 67(3):1430–1452.

Bulman G, Fairlie RW (2016) Technology and education: Computers, software, and the internet. *Handbook of the Economics of Education*, volume 5, 239–280 (Elsevier).

Cai A, Rick SR, Heyman JL, Zhang Y, Filipowicz A, Hong M, Klenk M, Malone T (2023) Designaid: Using generative ai and semantic diversity for design inspiration. *Proceedings of The ACM Collective Intelligence Conference*, 1–11.

Carrell SE, West JE (2010) Does professor quality matter? evidence from random assignment of students to professors. *Journal of Political Economy* 118(3):409–432.

Chan TH, Mihm J, Sosa ME (2018) On styles in product design: An analysis of us design patents. *Management Science* 64(3):1230–1249.

Chang CY, Hwang GJ, Gau ML (2022) Promoting students' learning achievement and self-efficacy: A mobile chatbot approach for nursing training. *British Journal of Educational Technology* 53(1):171–188.

Chatterji AK (2018) Innovation and american k–12 education. *Innovation Policy and the Economy* 18(1):27–51.

Chen DL, Schonger M, Wickens C (2016) otree - an open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance* 9:88–97.

Chen HL, Vicki Widarso G, Sutrisno H (2020) A chatbot for learning chinese: Learning achievement and technology acceptance. *Journal of Educational Computing Research* 58(6):1161–1189.

Chen Z, Chan J (2024) Large language model in creative work: The role of collaboration modality and user expertise. *Management Science* 70(12):9101–9117.

Chernev A, Böckenholt U, Goodman J (2015) Choice overload: A conceptual review and meta-analysis. *Journal of Consumer Psychology* 25(2):333–358.

Cohen PA, Kulik JA, Kulik CLC (1982) Educational outcomes of tutoring: A meta-analysis of findings. *American Educational Research Journal* 19(2):237–248.

Correia S (2017) Linear Models with High-Dimensional Fixed Effects: An Efficient and Feasible Estimator. URL http://scorreia.com/research/hdfe.pdf.

Corsten D, Skousen BR (2023) 'get ai answer?'–exploring the effects of embedding chatgpt in exams to improve student learning outcomes. *Available at SSRN 4409035* .

Crilly N (2015) Fixation and creativity in concept development: The attitudes and practices of expert designers. *Design studies* 38:54–91.

Damgaard MT, Nielsen HS (2020) Behavioral economics and nudging in education: Evidence from the field. *The Economics of Education*, 21–35 (Elsevier).

Dell'Acqua F (2022) Falling asleep at the wheel: Human/ai collaboration in a field experiment on hr recruiters. Technical report.

Dell'Acqua F, McFowland III 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. *Harvard Business School Technology & Operations Mgt. Unit Working Paper* (24-013).

Demirci O, Hannane J, Zhu X (2025) Who is ai replacing? the impact of generative ai on online freelancing platforms. *Management Science* .

Deslauriers L, McCarty LS, Miller K, Callaghan K, Kestin G (2019) Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom. *Proceedings of the National Academy of Sciences* 116(39):19251–19257.

Drori I, Zhang S, Shuttleworth R, Tang L, Lu A, Ke E, Liu K, Chen L, Tran S, Cheng N, et al. (2022) A neural network solves, explains, and generates university math problems by program synthesis and few-shot learning at human level. *Proceedings of the National Academy of Sciences* 119(32):e2123433119.

Dubey A, Jauhri A, Pandey A, Kadian A, Al-Dahle A, Letman A, Mathur A, Schelten A, Yang A, Fan A, et al. (2024) The llama 3 herd of models. *arXiv e-prints* arXiv–2407.

Elo AE, Sloan S (1978) The rating of chessplayers: Past and present. *(No Title)* .

Ericson KM, Laibson D (2019) Intertemporal choice. *Handbook of Behavioral Economics: Applications and Foundations 1*, volume 2, 1–67 (Elsevier).

Essel HB, Vlachopoulos D, Tachie-Menson A, Johnson EE, Baah PK (2022) The impact of a virtual teaching assistant (chatbot) on students' learning in ghanaian higher education. *International Journal of Educational Technology in Higher Education* 19(1):57.

Extance A (2023) Chatgpt has entered the classroom: how llms could transform education. *Nature* 623(7987):474–477.

Fleming L (2001) Recombinant uncertainty in technological search. *Management science* 47(1):117–132.

Fleming L, Sorenson O (2004) Science as a map in technological search. *Strategic management journal* 25(8-9):909–928.

Freeman J (2025) Student generative ai survey 2025. Technical report, HEPI, URL https://www.hepi.ac.uk/2025/02/26/student-generative-ai-survey-2025/.

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.

Garcia R, Calantone R (2002) A critical look at technological innovation typology and innovativeness terminology: a literature review. *Journal of Product Innovation Management: An international publication of the product development & management association* 19(2):110–132.

Gemini Team G, Anil R, Borgeaud S, Wu Y, Alayrac JB, Yu J, Soricut R, Schalkwyk J, Dai AM, Hauth A, et al. (2023) Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805* .

Girotra K, Terwiesch C, Ulrich KT (2010) Idea generation and the quality of the best idea. *Management science* 56(4):591–605.

Goldschmidt G (1991) The dialectics of sketching. *Creativity research journal* 4(2):123–143.

Gottweis J, Weng WH, Daryin A, Tu T, Palepu A, Sirkovic P, Myaskovsky A, Weissenberger F, Rong K, Tanno R, et al. (2025) Towards an ai co-scientist. *arXiv preprint arXiv:2502.18864* .

Haefner N, Wincent J, Parida V, Gassmann O (2021) Artificial intelligence and innovation management: A review, framework, and research agenda. *Technological Forecasting and Social Change* 162:120392.

Hanushek EA (2020) Education production functions. *The Economics of Education*, 161–170 (Elsevier).

Hoekstra M (2020) Returns to education quality. *The Economics of Education*, 65–73 (Elsevier).

Hooshangi S, Loewenstein G (2018) The impact of idea generation and potential appropriation on entrepreneurship: an experimental study. *Management Science* 64(1):64–82.

Jansson DG, Smith SM (1991) Design fixation. *Design studies* 12(1):3–11.

Jensen R (2010) The (perceived) returns to education and the demand for schooling. *The Quarterly Journal of Economics* 125(2):515–548.

Jin M, Chua R (2024) Which idea to pursue? gender differences in novelty avoidance during creative idea selection. *Organization Science* .

Jin Y, Lee K (2024) Human-ai co-creation in fashion design ideation and sketching: an empirical study. *Proceedings of IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR), CVFAD Workshop, Seattle, USA*.

Jonson B (2005) Design ideation: the conceptual sketch in the digital age. *Design studies* 26(6):613–624.

Kasneci E, Seßler K, Küchemann S, Bannert M, Dementieva D, Fischer F, Gasser U, Groh G, Günnemann S, Hüllermeier E, et al. (2023) Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences* 103:102274.

Kavadias S, Hutchison-Krupat J (2020) A framework for managing innovation. *Pushing the boundaries: Frontiers in impactful OR/OM research*, 202–228 (INFORMS).

Kavadias S, Ulrich KT (2020) Innovation and new product development: Reflections and insights from the research published in the first 20 years of manufacturing & service operations management. *Manufacturing & Service Operations Management* 22(1):84–92.

Keppler S, Sinchaisri WP, Snyder C (2024) Backwards planning with generative ai: Case study evidence from us k12 teachers. *Available at SSRN 4924786* .

Keppler SM (2024) Little's law and educational inequality: A comparative case study of teacher workaround productivity. *Management Science* 70(5):2756–2778.

Keppler SM, Li J, Wu D (2022) Crowdfunding the front lines: An empirical study of teacher-driven school improvement. *Management Science* 68(12):8809–8828.

Kestin G, Miller K, Klales A, Milbourne T, Ponti G (2024) Ai tutoring outperforms active learning. *Preprint. Available at* `math.rochester.edu/people/faculty/cohf` .

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25.

Kumar H, Rothschild DM, Goldstein DG, Hofman JM (2023) Math education with large language models: Peril or promise? *Available at SSRN 4641653* .

Lau T, Carter S, Chen F, Huynh B, Kimani E, Lee ML, Sieck KA (2024) Democratizing design through generative ai. *Companion Publication of the 2024 ACM Designing Interactive Systems Conference*, 239–244.

Lavecchia AM, Liu H, Oreopoulos P (2016) Behavioral economics of education: Progress and possibilities. *Handbook of the Economics of Education*, volume 5, 1–74 (Elsevier).

Law KK, Shen M (2025) How does artificial intelligence shape audit firms? *Management Science* 71(5):3641–3666.

Lee HS, Kesavan S, Deshpande V (2021) Managing the impact of fitting room traffic on retail sales: Using labor to reduce phantom stockouts. *Manufacturing & Service Operations Management* 23(6):1580–1596.

Lee S, Sosa M (2024) Spaces for creativity: Unconventional workspaces and divergent thinking. *Management Science* (forthcoming).

Lehmann M, Cornelius PB, Sting FJ (2024) Ai meets the classroom: When does chatgpt harm learning? *arXiv preprint arXiv:2409.09047* .

Lemonaki D (2022) The best programming language to learn - beginners guide to coding. URL https://www.freecodecamp.org/news/the-best-programming-language-to-learn-beginners-guide-to-coding/.

Levinthal DA (1997) Adaptation on rugged landscapes. *Management science* 43(7):934–950.

Li C, Zhang T, Du X, Zhang Y, Xie H (2024) Generative ai for architectural design: A literature review. *arXiv preprint arXiv:2404.01335* .

Liao W, Lu X, Fei Y, Gu Y, Huang Y (2024) Generative ai design for building structures. *Automation in Construction* 157:105187.

Lichtenthaler U (2018) Substitute or synthesis: the interplay between human and artificial intelligence. *Research-technology management* 61(5):12–14.

Liu J, Xia CS, Wang Y, Zhang L (2024a) Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems* 36.

Liu R, Zenke C, Liu C, Holmes A, Thornton P, Malan DJ (2024b) Teaching cs50 with ai: leveraging generative artificial intelligence in computer science education. *Proceedings of the 55th ACM Technical Symposium on Computer Science Education*, volume 1, 750–756.

Loch CH (2017) Creativity and risk taking aren't rational: Behavioral operations in MOT. *Production and Operations Management* 26(4):591–604.

Lovins JB (1968) Development of a stemming algorithm. *Mech. Transl. Comput. Linguistics* 11(1-2):22–31.

March JG (1991) Exploration and exploitation in organizational learning. *Organization science* 2(1):71–87.

Meincke L, Girotra K, Nave G, Terwiesch C, Ulrich KT (2024) Using large language models for idea generation in innovation. *The Wharton School Research Paper Forthcoming* 9:2024.

Meyer JG, Urbanowicz RJ, Martin PC, O'Connor K, Li R, Peng PC, Bright TJ, Tatonetti N, Won KJ, Gonzalez-Hernandez G, et al. (2023) Chatgpt and large language models in academia: opportunities and challenges. *BioData Mining* 16(1):20.

Miola L, Muffato V, Sella E, Meneghetti C, Pazzaglia F (2024) Gps use and navigation ability: A systematic review and meta-analysis. *Journal of Environmental Psychology* 102417.

Mollick ER, Mollick L (2022) New modes of learning enabled by ai chatbots: Three methods and assignments. *Available at SSRN 4300783* .

Mountstephens J, Teo J (2020) Progress and challenges in generative product design: A review of systems. *Computers* 9(4):80.

Mueller PA, Oppenheimer DM (2014) The pen is mightier than the keyboard: Advantages of longhand over laptop note taking. *Psychological Science* 25(6):1159–1168.

Narayanan S, Balasubramanian S, Swaminathan JM (2009) A matter of balance: Specialization, task variety, and individual learning in a software maintenance environment. *Management Science* 55(11):1861–1876.

Nie A, Chandak Y, Suzara M, Malik A, Woodrow J, Peng M, Sahami M, Brunskill E, Piech C (2024) The gpt surprise: Offering large language model chat in a massive coding class reduced engagement but increased adopters' exam performances. Technical report, Center for Open Science.

Noy S, Zhang W (2023) Experimental evidence on the productivity effects of generative artificial intelligence. *Science* 381(6654):187–192.

O'brien RM (2007) A caution regarding rules of thumb for variance inflation factors. *Quality & Quantity* 41:673–690.

OpenAI (2022) Introducing chatgpt. URL https://openai.com/index/chatgpt/.

Osadcha KP, Osadcha MV (2023) Generative artificial intelligence vs humans in the process of creating corporate identity elements. *Information Technologies and Learning Tools* 98(6):212.

Pardos ZA, Bhandari S (2023) Learning gain differences between chatgpt and human tutor generated algebra hints. *arXiv preprint arXiv:2302.06871* .

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Prihar E, Lee M, Hopman M, Kalai AT, Vempala S, Wang A, Wickline G, Murray A, Heffernan N (2023) Comparing different approaches to generating mathematics explanations using large language models. *International Conference on Artificial Intelligence in Education*, 290–295 (Springer).

Purcell AT, Gero JS (1998) Drawings and the design process: A review of protocol studies in design and other disciplines and related research in cognitive psychology. *Design studies* 19(4):389–430.

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).

Raisch S, Fomina K (2023) Combining human and artificial intelligence: Hybrid problem-solving in organizations. *Academy of Management Review* (ja):amr–2021.

Raisch S, Krakowski S (2021) Artificial intelligence and management: The automation–augmentation paradox. *Academy of management review* 46(1):192–210.

Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M (2022) Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* 1(2):3.

Rombach R, Blattmann A, Lorenz D, Esser P, Ommer B (2022) High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Saharia C, Chan W, Saxena S, Li L, Whang J, Denton EL, Ghasemipour K, Gontijo Lopes R, Karagol Ayan B, Salimans T, et al. (2022) Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems* 35:36479–36494.

Shane SA, Ulrich KT (2004) 50th anniversary article: Technological innovation, product development, and entrepreneurship in management science. *Management Science* 50(2):133–144.

Silver D, Hubert T, Schrittwieser J, Antonoglou I, Lai M, Guez A, Lanctot M, Sifre L, Kumaran D, Graepel T, et al. (2018) A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science* 362(6419):1140–1144.

Silver D, Schrittwieser J, Simonyan K, Antonoglou I, Huang A, Guez A, Hubert T, Baker L, Lai M, Bolton A, et al. (2017) Mastering the game of go without human knowledge. *nature* 550(7676):354–359.

Smilowitz K, Keppler S (2020) On the use of operations research and management in public education systems. *Pushing the Boundaries: Frontiers in Impactful OR/OM Research* 84–105.

Smith SM, Ward TB, Schumacher JS (1993) Constraining effects of examples in a creative generation task. *Memory & cognition* 21:837–845.

Song S, Agogino AM (2004) Insights on designers' sketching activities in new product design teams. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 46962, 351–360.

Sparck Jones K (1972) A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28(1):11–21.

Spring M, Faulconbridge J, Sarwar A (2022) How information technology automates and augments processes: Insights from artificial-intelligence-based systems in professional service operations. *Journal of Operations Management* 68(6-7):592–618.

Teodoridis F (2018) Understanding team knowledge production: The interrelated roles of technology and expertise. *Management Science* 64(8):3625–3648.

Terwiesch C (2023) Would chat gpt3 get a wharton mba? a prediction based on its performance in the operations management course. Technical report, Mack Institute for Innovation Management at the Wharton School, University of Pennsylvania.

Tholander J, Jonsson M (2023) Design ideation with ai-sketching, thinking and talking with generative machine learning models. *Proceedings of the 2023 ACM designing interactive systems conference*, 1930–1940.

Touvron H, Lavril T, Izacard G, Martinet X, Lachaux MA, Lacroix T, Rozière B, Goyal N, Hambro E, Azhar F, et al. (2023) Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* .

Tovey M, Porter S, Newman R (2003) Sketching, concept development and automotive design. *Design studies* 24(2):135–153.

Ulrich KT, Eppinger SD (2016) *Product design and development* (McGraw-hill).

Vailshery LS (2024) Most used programming languages among developers worldwide as of 2023. URL https://www.statista.com/statistics/793628/worldwide-developer-survey-most-used-languages/.

Van de Ven AH (1986) Central problems in the management of innovation. *Management science* 32(5):590–607.

Vasconcelos LA, Crilly N (2016) Inspiration and fixation: Questions, methods, findings, and challenges. *Design Studies* 42:1–32.

Wadinambiarachchi S, Kelly RM, Pareek S, Zhou Q, Velloso E (2024) The effects of generative ai on design fixation and divergent thinking. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1–18.

Wang B, Liu Y, Parker SK (2020) How does the use of information communication technology affect individuals? a work design perspective. *Academy of Management Annals* 14(2):695–725.

Wang X, Wu L (2025) Artificial intelligence, lean startup method, and product innovations. *Management Science* .

Wang Y, Damen NB, Gale T, Seo V, Shayani H (2024) Inspired by ai? a novel generative ai system to assist conceptual automotive design. *arXiv preprint arXiv:2407.11991* .

Wiltermuth SS, Gubler T, Pierce L (2023) Anchoring on historical round number reference points: Evidence from durable goods resale prices. *Organization Science* 34(5):1839–1863.

Wooldridge JM (2010) *Econometric analysis of cross section and panel data* (Cambridge, Mass.: MIT Press).

Wu L, Lou B, Hitt LM (2025) Innovation strategy after ipo: How ai analytics spurs innovation after ipo. *Management Science* 71(3):2360–2389.

Wuttke D, Upadhyay A, Siemsen E, Wuttke-Linnemann A (2022) Seeing the bigger picture? ramping up production with the use of augmented reality. *Manufacturing & Service Operations Management* 24(4):2349–2366.

Yin J, Goh TT, Yang B, Xiaobin Y (2021) Conversation technology with micro-learning: The impact of chatbot-based learning on students' learning motivation and performance. *Journal of Educational Computing Research* 59(1):154–177.

Yoo OS, Zhan D (2023) Economic behavior of information acquisition: Impact on peer grading in massive open online courses. *Operations Research* 71(4):1277–1297.

Youmans RJ, Arciszewski T (2014) Design fixation: Classifications and modern methods of prevention. *AI EDAM* 28(2):129–137.

Zhang C, Wang W, Pangaro P, Martelaro N, Byrne D (2023) Generative image ai using design sketches as input: Opportunities and challenges. *Proceedings of the 15th Conference on Creativity and Cognition*, 254–261.

Zhang DJ, Allon G, Van Mieghem JA (2017) Does social interaction improve learning outcomes? evidence from field experiments on massive open online courses. *Manufacturing & Service Operations Management* 19(3):347–367.

Appendices

A. Experimental Interface

Here, we describe the experimental interface with which subjects interacted in the pre-test, learning phase, and post-test. Figure 5 shows a screenshot of the interface for the control condition during the pre-test. In the top left quadrant, subjects see the description of the current question and, in the learning phase, helpful information to learn how to solve this question. In the top right quadrant, subjects have access to a fully functioning code editor in which they can enter their Python code to solve the respective question. They can run code as often as they like. When running code, subjects see the full output below the editor including potential error messages and a short message if the question was solved correctly. We determine the correctness of solutions with predefined test cases. In addition to the above, the treatment condition also has access to an LLM during the learning phase in a chat window in the bottom left quadrant (see Figure 6).

Figure 5 Main experiment interface.

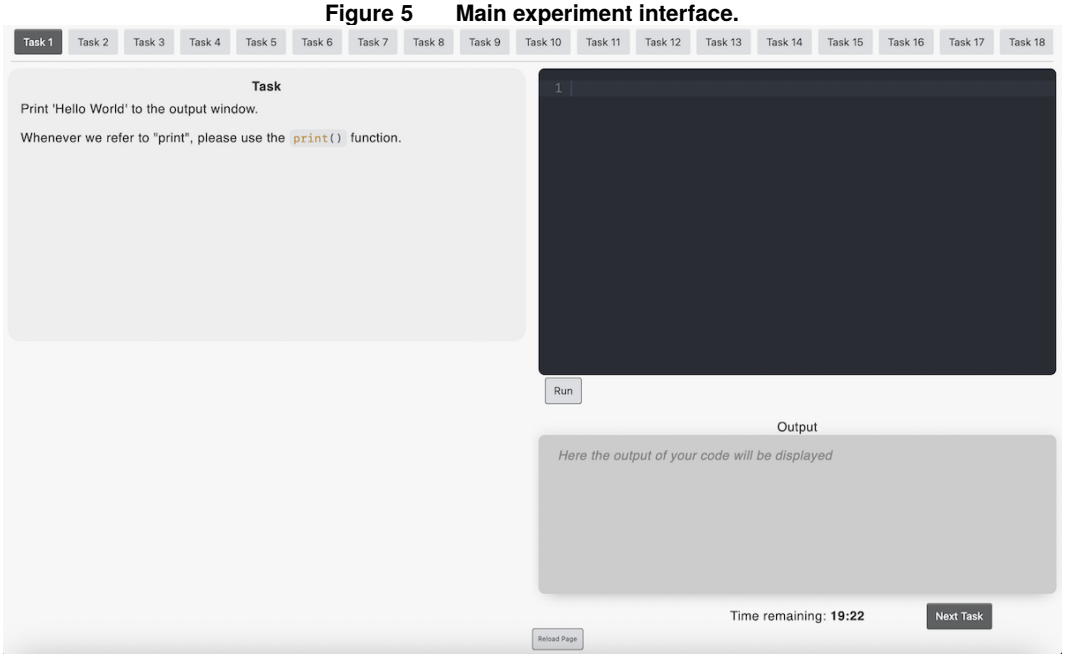


Figure 6 Treatment Condition Interface.

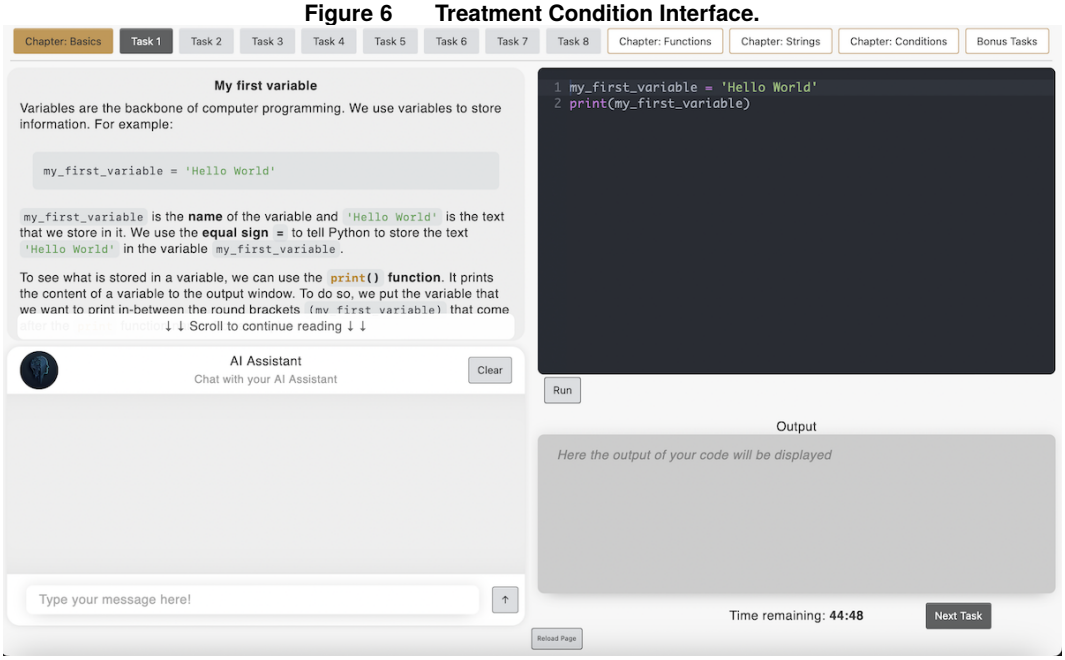


Table 26 Study 2 and 3: Description of Covariates.

Covariate	Description
Gender	Indicator whether the participant is male.
Level of Studies	The current level of studies, i.e. bachelor's, master's etc., of the participant.
GPA	Average grade of the participant on a German grading scale from 1 (best) to 4 (worst).
Age	The age of the participant, discretized into buckets labeled from 1 to 5.
Coding Experience	Prior experience in any programming language on a five-point Likert scale.
Python Experience	Prior experience in Python on a five-point Likert scale.
Studiousness	How much students learn on average for an exam on a five-point Likert scale.
LLM Used Before	Binary indicator whether the participant has used LLMs before.
LLM Experience	Experience with LLMs on a five-point Likert scale.

B. Pre-treatment Variables

In Studies 2 and 3, we measure a variety of pre-treatment variables. Table 26 summarizes these variables and Table 27 presents summary statistics.

Table 27 Studies 2 and 3: Descriptive Statistics.

	Study 2		Study 3	
	Control	Treatment	Control	Treatment
Gender (male = 1)	0.510	0.375	0.548	0.316
	(0.505)	(0.489)	(0.506)	(0.471)
Level of Studies	2.255	2.357	2.161	2.158
	(0.771)	(0.750)	(0.779)	(0.718)
GPA	1.978	1.880	2.132	1.847
	(0.570)	(0.645)	(0.664)	(0.473)
Age	2.569	2.643	2.839	2.395
	(0.878)	(0.796)	(1.241)	(1.054)
Coding Experience	1.471	1.767	1.806	1.711
	(0.674)	(0.809)	(0.910)	(0.732)
Python Experience	1.176	1.339	1.290	1.342
	(0.434)	(0.611)	(0.588)	(0.781)
Studiousness	4.373	4.375	4.000	4.105
	(1.312)	(1.342)	(1.789)	(1.448)
LLM Used Before (yes = 1)	0.804	0.911	0.871	0.816
	(0.401)	(0.288)	(0.341)	(0.393)
LLM Experience	2.353	2.321	2.387	2.000
	(1.415)	(1.336)	(1.476)	(1.252)

Note. We report the means and in brackets the standard deviations.

C. Coding of Chat Messages

We code each message sent to the LLM. We blindly assess the messages and assign them to a category based on the intention of the subject. Categories were created inductively during the coding process whenever we found a message that did not fit into a previously created category. Table 28 summarizes the resulting categories of this coding procedure. The categories are clearly distinct from each other. *Solutions* are easily identifiable by containing phrases such as "solve this task" or the copied question description. *Explanations* are identifiable by asking other problem related questions without explicitly asking for a solution.

D. Robustness Checks

In this section, we provide additional robustness checks of our results.

First, our primary analyses focus on intention-to-treat effects since subjects in the

Table 28 Chat Messages Types.

Category	Description	Example
Solutions	User asks the LLM to solve an exercise, most commonly by pasting in the entire task description.	*"Complete the function hypotenuse..."*
Explanations	User asks the LLM to explain a programming concept.	*"Explain what the str() function does"*
Miscalleneous	Messages not related to the tasks.	*"hows your day mate?"*
Translations	User asks for a translation from English to German.	*"what is the german word for concatenate"*
User Errors	Most commonly users ask questions without providing context and assume the LLM has access to everything onscreen.	*"What am I doing wrong here?"*

treatment group only have access to the LLM but must not necessarily use it. We repeat our analyses on the effects of treatment in Study 2 (Table 5) and Study 3 (Table 7) in Table 29 after excluding unsuccessfully treated subjects, i.e. those who did not use the LLM at all (six in Study 2 and two in Study 3). The results conform to our previous findings as we find no effect of LLM usage (*Treatment*) on learning outcomes (*Post-test*, column 1: $p = 0.868$, column 4: $p = 0.333$) or on understanding (column 2: $p = 0.319$, column 5: $p = 0.342$) and no effect on topic volume in Study 2 (column 3, $p = 0.262$) but a positive effect in Study 3 (column 6, $p = 0.015$).

Second, we replicate the analyses in Table 11 (Section 2.9) with a relative measure of usage behavior by including the total number of *Messages* sent to the LLM as an additional control. The results are shown in Table 30. Including this control subsumes the effect of *Solutions* across all specifications ($p = 0.906$, $p = 0.090$). However, we remark that *Messages* is highly correlated with *Solutions* with a Pearson correlation coefficient of $\rho = 0.81$ ($p < 0.001$) and produces a Variable Inflation Factor greater than ten, indicating significant multicollinearity and providing grounds for exclusion (O'brien 2007).

Finally, we also consider an alternative dependent variable to measure understanding. Previously, we inferred understanding by measuring learning outcomes (*Post-test*) while controlling for volume (*Learning Phase*). In Table 31, we replicate our main analyses on understanding with another variable. *Covered Post-test* measures the fraction of *Post-test*

Table 29 Studies 2 and 3: Treatment Effects Without Unsuccessfully Treated.

	Study 2			Study 3		
	Post-test		*Learning Phase*	*Post-test*		*Learning Phase*
	(1)	(2)	(3)	(4)	(5)	(6)
Treatment (LLM access = 1)	0.122 (0.733)	-0.495 (0.494)	0.760 (0.674)	0.920 (0.941)	-0.701 (0.731)	2.623** (1.043)
Gender (male = 1)	0.274 (0.733)	-0.986* (0.505)	1.553** (0.674)	1.936** (0.913)	1.019 (0.685)	1.484 (1.011)
Level of Studies	1.129** (0.530)	-0.072 (0.373)	1.479*** (0.487)	0.193 (0.597)	0.425 (0.440)	-0.375 (0.661)
GPA	-0.443 (0.573)	0.106 (0.387)	-0.676 (0.526)	-2.234** (0.846)	-0.495 (0.671)	-2.814*** (0.937)
Age	-0.439 (0.486)	0.600* (0.340)	-1.280*** (0.447)	-0.305 (0.389)	-0.301 (0.286)	-0.006 (0.431)
Coding Experience	0.621 (0.609)	0.334 (0.408)	0.354 (0.559)	1.807** (0.690)	1.086** (0.519)	1.166 (0.765)
Python Experience	-0.266 (0.812)	0.066 (0.545)	-0.409 (0.746)	-0.682 (0.738)	-0.481 (0.544)	-0.325 (0.818)
Studiousness	-0.176 (0.267)	-0.099 (0.179)	-0.094 (0.245)	-0.264 (0.278)	-0.188 (0.205)	-0.124 (0.308)
LLM Used Before (yes = 1)	0.470 (1.117)	0.748 (0.749)	-0.342 (1.027)	1.245 (1.309)	0.864 (0.965)	0.617 (1.450)
LLM Experience	-0.280 (0.293)	0.145 (0.201)	-0.523* (0.270)	-0.342 (0.351)	-0.364 (0.259)	0.035 (0.389)
Pre-test	0.788*** (0.142)	0.211* (0.110)	0.710*** (0.131)	0.773*** (0.148)	0.329** (0.127)	0.718*** (0.164)
Learning Phase		0.812*** (0.077)			0.618*** (0.090)	
Constant	5.233* (2.662)	-7.192*** (2.138)	15.301*** (2.446)	7.317** (3.131)	-2.957 (2.743)	16.622*** (3.467)
Observations	101	101	101	67	67	67
R^2	0.484	0.771	0.523	0.637	0.807	0.604
Adjusted R^2	0.420	0.740	0.464	0.565	0.765	0.525

Notes. We exclude subjects in the treatment condition who did not use the LLM at all. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

Table 30 Studies 2 and 3: Additional Analyses of the Effect of Usage Behaviors on Learning Outcomes.

	Post-test	
	(1)	(2)
Solutions	0.025 (0.207)	-0.223* (0.130)
Explanations	0.264 (0.294)	0.355* (0.182)
Copy/Paste (enabled = 1)	0.415 (0.801)	-0.541 (0.501)
Gender (male = 1)	2.422*** (0.802)	0.903* (0.513)
Level of Studies	0.221 (0.536)	0.179 (0.331)
GPA	-0.877 (0.629)	0.065 (0.397)
Age	-0.182 (0.441)	-0.011 (0.273)
Coding Experience	1.041 (0.700)	0.782* (0.433)
Python Experience	-0.914 (0.718)	-0.328 (0.446)
Studiousness	0.050 (0.273)	0.038 (0.168)
LLM Used Before (yes = 1)	-1.246 (1.259)	-1.117 (0.778)
LLM Experience	0.086 (0.336)	0.072 (0.207)
Pre-test	0.828*** (0.123)	0.246*** (0.092)
Learning Phase		0.730*** (0.064)
Messages	-0.228 (0.184)	-0.092 (0.115)
Constant	7.437*** (2.414)	-4.677** (1.832)
Observations	94	94
R^2	0.596	0.848
Adjusted R^2	0.524	0.819

Notes. Regressions only include treated subjects. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

questions solved whose topics the subject covered during the learning phase. In other words, *Covered Post-test* is normalized by learning phase progress. As in Table 5 and Table 7, LLM access has no effect on understanding (columns 1 and 2, $p = 0.527$, $p = 0.580$). Also similarly to our earlier analyses in Table 10 and Table 11 in Section 2.9, we find a negative (albeit now weaker) effect of copy and paste on understanding ($p = 0.086$). This effect vanishes once we include *Solutions* and *Explanations* ($p = 0.458$). Conforming to our main analysis, *Solutions* decrease *Covered Post-test* ($p = 0.018$) whereas *Explanations* increase *Covered Post-test* ($p = 0.003$).

E. Robustness Checks

To verify that our main behavioral results in section 3.5 and section 3.6 are not driven by the particular choice of vectorization technique or thresholds for constructing *Clusters*

Table 31 Additional Analyses of the Effects on Understanding.

	Solved of Covered			
	Study 2	Study 3	Studies 2 & 3	
	(1)	(2)	(3)	(4)
Solutions				-0.016** (0.007)
Explanations				0.038*** (0.012)
Treatment (LLM access = 1)	-0.034 (0.054)	-0.036 (0.065)		
Copy/Paste (enabled = 1)			-0.085* (0.049)	-0.037 (0.050)
Gender (male = 1)	-0.127** (0.054)	0.024 (0.064)	0.001 (0.053)	0.034 (0.050)
Level of Studies	0.067* (0.039)	0.002 (0.042)	0.086** (0.035)	0.071** (0.033)
GPA	-0.010 (0.043)	-0.064 (0.059)	-0.036 (0.042)	-0.025 (0.039)
Age	-0.010 (0.035)	-0.060** (0.027)	-0.064** (0.029)	-0.051* (0.027)
Coding Experience	0.045 (0.046)	0.061 (0.048)	0.115** (0.046)	0.084* (0.043)
Python Experience	-0.039 (0.061)	-0.001 (0.052)	-0.070 (0.047)	-0.038 (0.044)
Studiousness	0.011 (0.020)	-0.039** (0.019)	0.017 (0.018)	0.020 (0.017)
LLM Used Before (yes = 1)	0.190** (0.084)	0.187** (0.090)	0.109 (0.079)	0.030 (0.077)
LLM Experience	-0.019 (0.022)	-0.020 (0.025)	-0.033 (0.021)	-0.013 (0.021)
Pre-test	0.024** (0.011)	0.026** (0.010)	0.022** (0.008)	0.020** (0.008)
Constant	0.378* (0.194)	0.740*** (0.218)	0.451*** (0.158)	0.417*** (0.148)
Observations	107	69	94	94
R^2	0.222	0.403	0.361	0.465
Adjusted R^2	0.132	0.287	0.276	0.378

Notes. Columns 1 and 2 include all subjects, columns 3 and 4 include only treated subjects. Standard errors are in parenthesis. *: $p<0.1$; **: $p<0.05$; ***: $p<0.01$.

and *Long Jumps* (see Section 3.4.3.1), we conduct a series of robustness checks for both Study 1 and Study 2. We re-compute our core exploration metrics using altered metrics and confirm that the treatment effects of sketching remain substantively unchanged. Across alternative vectorization techniques (CLIP-image, TF-IDF) and a range of similarity cutoffs for both clustering and long-jump detection, the core behavioral patterns – sketching reduces idea variance (higher similarity metrics) and global exploration (fewer clusters, fewer long jumps) – persist. These robustness checks affirm that our main findings are not artifacts of specific measuring choices.

E.1. Alternative Vectorization Techniques

Converting the non-numerical products of the design processes in our experiments to measurable points, for which we can compute similarities, can be done in a variety of ways.

Table 32 Designer Behavior With Different Vectorization Techniques – T-Tests.

		Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
CLIP-Image	Study 1	0.003	0.005	0.060	0.058	0.007	0.080
	Study 2	0.002	0.21	0.017	0.009	0.001	0.004
TF-IDF	Study 1	< 0.001	0.002	0.007	0.002	< 0.001	< 0.001
	Study 2	0.001	0.014	0.005	0.010	< 0.001	< 0.001

Note. P-values for two-tailed Welch's T-Tests.

Table 33 Designer Behavior With Different Vectorization Techniques – Regression P-Values.

		Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
CLIP-Image	Study 1	0.004	0.005	0.042	0.030	0.008	0.091
	Study 2	< 0.001	0.004	0.045	0.017	< 0.001	0.004
TF-IDF	Study 1	< 0.001	0.001	0.003	0.001	< 0.001	< 0.001
	Study 2	0.001	0.003	0.009	0.021	< 0.001	< 0.001

Note. P-values of the treatment indicator from the respective regression analyses in Tables 34 – 37.

While we argue that CLIP-based vectorization of prompts is the most appropriate choice due to the capability to capture semantic changes across designer inputs without being obscured by the stochasticity of the image generation AI, we here also explore alternative vectorization techniques. First, we embed the actual logo images generated at each iteration by the subjects using a pre-trained CLIP image encoder. We then recomputed all pairwise-similarity-based metrics (*Mean Similarity*, *Trajectory Similarity*, *First-Last*, *First-Submitted*, *Cluster* counts, and *Long Jump* counts using these image embeddings. Across both studies, the direction and statistical significance of the treatment coefficients in two-tailed Welch's *t*-tests (Table 32) and OLS regressions (summary in Table 33, individual regressions in Table 34 and Table 35) remain virtually identical to those reported in Sections 3.5.1 and 3.6.1. Next, as a purely text-based alternative, we re-vectorize each prompt via TF-IDF over the vocabulary of prompts. Again, all exploration-related treatment effects replicate: sketching significantly increases *Mean Similarity* and *Trajectory Similarity*, reduces the number of *Clusters* and *Long Jumps*, and yields higher *First-Last* and *First-Submitted* similarity (all $p < 0.05$, regressions in Table 36 and Table 37).

Table 34 Study 1 – Designer Behavior: CLIP-based image vectorization.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
Treatment (sketch = 1)	0.054*** (0.018)	0.054*** (0.019)	-1.069** (0.520)	-1.637** (0.745)	0.063*** (0.023)	0.062* (0.036)
AI Experience	-0.001 (0.009)	0.000 (0.009)	-0.327 (0.244)	-0.321 (0.350)	0.002 (0.011)	-0.003 (0.017)
Num. Designs	0.002* (0.001)	0.002** (0.001)	0.135*** (0.025)	0.292*** (0.036)	0.000 (0.001)	-0.001 (0.002)
Constant	0.311*** (0.025)	0.311*** (0.026)	3.196*** (0.690)	2.320** (0.989)	0.296*** (0.031)	0.349*** (0.049)
Observations	94	94	107	107	105	105
R^2	0.120	0.141	0.255	0.412	0.069	0.032
Adjusted R^2	0.091	0.113	0.233	0.395	0.042	0.003

Notes. All metrics use image-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 35 Study 2 – Designer Behavior: CLIP-based image vectorization.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
Treatment (sketch = 1)	0.061*** (0.017)	0.054*** (0.018)	-0.503** (0.248)	-0.977** (0.403)	0.088*** (0.022)	0.111*** (0.038)
AI Experience	-0.006 (0.007)	-0.008 (0.008)	-0.054 (0.102)	0.217 (0.165)	0.001 (0.009)	-0.004 (0.016)
Num. Designs	0.004*** (0.001)	0.005*** (0.001)	0.078*** (0.016)	0.149*** (0.026)	0.005*** (0.001)	0.000 (0.002)
Constant	0.346*** (0.024)	0.349*** (0.027)	2.030*** (0.360)	0.871 (0.584)	0.300*** (0.033)	0.409*** (0.055)
Observations	132	132	139	139	137	139
R^2	0.151	0.142	0.181	0.249	0.149	0.059
Adjusted R^2	0.132	0.122	0.162	0.232	0.129	0.038

Notes. All metrics use image-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 36 Study 1 – Designer Behavior: TF-IDF prompt vectorization.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
Treatment (sketch = 1)	0.156*** (0.042)	0.139*** (0.042)	-1.795*** (0.599)	-2.225*** (0.666)	0.176*** (0.046)	0.221*** (0.053)
AI Experience	-0.011 (0.020)	-0.014 (0.020)	0.024 (0.281)	0.100 (0.313)	0.007 (0.022)	-0.009 (0.025)
Num. Designs	-0.002 (0.002)	0.006*** (0.002)	0.205*** (0.029)	0.175*** (0.032)	-0.008*** (0.002)	-0.008*** (0.003)
Constant	0.410*** (0.056)	0.463*** (0.056)	3.294*** (0.795)	2.155** (0.885)	0.308*** (0.061)	0.369*** (0.071)
Observations	107	107	107	107	107	107
R^2	0.129	0.168	0.369	0.290	0.223	0.218
Adjusted R^2	0.104	0.143	0.351	0.270	0.200	0.195

Notes. All metrics use prompt-based TF-IDF-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 37 Study 2 – Designer Behavior: TF-IDF prompt vectorization.

	Mean Similarity	*Trajectory Similarity*	*Clusters*	*Long Jumps*	*First–Last*	*First–Submitted*
Treatment (sketch = 1)	0.156*** (0.046)	0.139*** (0.046)	-1.238*** (0.467)	-1.063** (0.454)	0.182*** (0.050)	0.202*** (0.056)
AI Experience	-0.009 (0.019)	-0.020 (0.019)	0.211 (0.192)	0.133 (0.187)	-0.010 (0.021)	0.006 (0.023)
Num. Designs	0.002 (0.003)	0.011*** (0.003)	0.215*** (0.030)	0.196*** (0.030)	-0.002 (0.003)	-0.004 (0.004)
Constant	0.384*** (0.066)	0.446*** (0.067)	2.110*** (0.677)	0.916 (0.658)	0.300*** (0.073)	0.341*** (0.081)
Observations	139	139	139	139	139	139
R^2	0.080	0.130	0.324	0.289	0.097	0.105
Adjusted R^2	0.059	0.110	0.309	0.274	0.077	0.085

Notes. All metrics use prompt-based TF-iDF-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 38 Designer Behavior: Clusters – Sensitivity Analysis.

	Threshold	0.25	0.30	0.35	0.40	0.45
Study 1	T-test	0.002	0.001	0.001	0.001	0.018
	Regression	0.001	0.000	0.000	0.001	0.022
Study 2	T-test	0.002	0.002	0.002	0.008	0.006
	Regression	0.004	0.004	0.004	0.016	0.013

Note. P-values for two-tailed Welch's T-Tests and regression for varying thresholds.

Table 39 Study 1 – Designer Behavior: Treatment Effects Cluster Threshold Sensitivity.

	Clusters				
	$c = 0.25$	$c = 0.30$	$c = 0.35$	$c = 0.40$	$c = 0.45$
Treatment (sketch = 1)	-1.502*** (0.437)	-1.387*** (0.365)	-1.136*** (0.314)	-0.934*** (0.261)	-0.437** (0.188)
AI Experience	-0.025 (0.205)	-0.011 (0.172)	-0.005 (0.147)	-0.034 (0.123)	0.066 (0.088)
Num. Designs	0.143*** (0.021)	0.108*** (0.018)	0.080*** (0.015)	0.067*** (0.013)	0.041*** (0.009)
Constant	2.872*** (0.580)	2.681*** (0.485)	2.500*** (0.417)	2.254*** (0.347)	1.754*** (0.249)
Observations	107	107	107	107	107
R^2	0.364	0.342	0.290	0.292	0.207
Adjusted R^2	0.345	0.323	0.269	0.271	0.183

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

E.2. Clustering and Long-Jump Threshold Sensitivity

To ensure that our *Clusters* and *Long Jumps* results are not driven by the particular choice of similarity cutoff, we run a sensitivity analysis for both clustering and long-jump analyses across a range of thresholds. For each threshold, we recompute the number of *Cluster* and the count of *Long Jumps* per subject, then test the treatment effect of sketching via Welch's *t*-tests and OLS regressions (controlling for AI experience and number of designs). In all threshold configurations across both studies, sketching yields significantly fewer *Clusters* (Table 38, $p < 0.05$ in all cases) and fewer *Long Jumps* (Table 41, $p < 0.05$ for all thresholds except the lowest in regressions for Study 1 ($p = 0.060$)). Tables 39 – 43 present the respective regressions analyses.

E.3. Additional Metrics

In our main analyses, we show that sketching has a significant negative effect on the similarity between the first design created and the design ultimately submitted (*First–Submitted*), thus

Table 40 Study 2 – Designer Behavior: Treatment Effects Cluster Threshold Sensitivity.

	Clusters				
Treatment (sketch = 1)	-1.082*** (0.374)	-0.954*** (0.321)	-0.877*** (0.295)	-0.619** (0.253)	-0.513** (0.205)
AI Experience	0.044 (0.154)	0.053 (0.132)	0.042 (0.121)	0.026 (0.104)	0.000 (0.084)
Num. Designs	0.129*** (0.024)	0.099*** (0.021)	0.081*** (0.019)	0.059*** (0.017)	0.048*** (0.013)
Constant	2.547*** (0.542)	2.324*** (0.465)	2.232*** (0.428)	2.052*** (0.367)	1.939*** (0.297)
Observations	139	139	139	139	139
R^2	0.231	0.208	0.182	0.135	0.138
Adjusted R^2	0.214	0.190	0.164	0.116	0.119

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 41 Designer Behavior: Long Jumps – Sensitivity Analysis.

	Threshold	0.55	0.60	0.65	0.70	0.75
Study 1	T-test	0.045	0.022	0.006	0.011	0.006
	Regression	0.060	0.027	0.006	0.009	0.005
Study 2	T-test	0.025	0.005	0.007	0.006	0.005
	Regression	0.045	0.011	0.016	0.014	0.010

Note. P-values for two-tailed Welch's T-Tests and regression for varying thresholds.

Table 42 Study 1 – Designer Behavior: Treatment Effects Long Jump Threshold Sensitivity.

	Long Jumps				
	$t = 0.55$	$t = 0.60$	$t = 0.65$	$t = 0.70$	$t = 0.75$
Treatment (sketch = 1)	-0.595* (0.313)	-0.969** (0.430)	-1.372*** (0.485)	-1.444*** (0.544)	-1.728*** (0.600)
AI Experience	0.098 (0.147)	0.058 (0.202)	-0.001 (0.228)	-0.059 (0.256)	0.056 (0.282)
Num. Designs	0.048*** (0.015)	0.072*** (0.021)	0.087*** (0.024)	0.118*** (0.026)	0.140*** (0.029)
Constant	0.577 (0.415)	0.914 (0.572)	1.487** (0.644)	1.757** (0.722)	2.011** (0.797)
Observations	107	107	107	107	107
R^2	0.125	0.148	0.180	0.214	0.240
Adjusted R^2	0.100	0.123	0.156	0.191	0.218

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

finding that the end point of the design search process is closer to the starting point than in the control condition. For robustness, we redo this analysis regarding the similarity between the first design and the last created design (*First–Last*). Note that the last created design and the submitted design are generally not identical as subjects were free to submit any of their designs after completing the design generation process. In both Studies 1 and 2, the

Table 43 Study 2 – Designer Behavior: Treatment Effects Long Jump Threshold Sensitivity.

	Long Jumps				
	$t = 0.55$	$t = 0.60$	$t = 0.65$	$t = 0.70$	$t = 0.75$
Treatment (sketch = 1)	-0.431** (0.213)	-0.700** (0.270)	-0.811** (0.331)	-0.937** (0.376)	-1.055** (0.405)
AI Experience	-0.018 (0.087)	0.021 (0.111)	0.041 (0.136)	0.051 (0.154)	0.049 (0.166)
Num. Designs	0.031** (0.014)	0.059*** (0.018)	0.102*** (0.022)	0.116*** (0.024)	0.172*** (0.026)
Constant	0.680** (0.309)	0.774* (0.392)	0.766 (0.480)	0.999* (0.545)	1.044* (0.587)
Observations	139	139	139	139	139
R^2	0.071	0.130	0.190	0.193	0.287
Adjusted R^2	0.051	0.111	0.172	0.175	0.271

Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

Table 44 Designer Behavior: Treatment Effects On Similarity Between First & Last Design.

	First–Last	
	Study 1	Study 2
Treatment (sketch = 1)	0.062** (0.026)	0.084*** (0.028)
AI Experience	-0.006 (0.012)	0.010 (0.012)
Num. Designs	-0.006*** (0.001)	-0.005*** (0.002)
Constant	0.712*** (0.035)	0.656*** (0.041)
Observations	107	139
Mean	0.653	0.680
Std.	0.151	0.174
R^2	0.232	0.124
Adjusted R^2	0.209	0.104

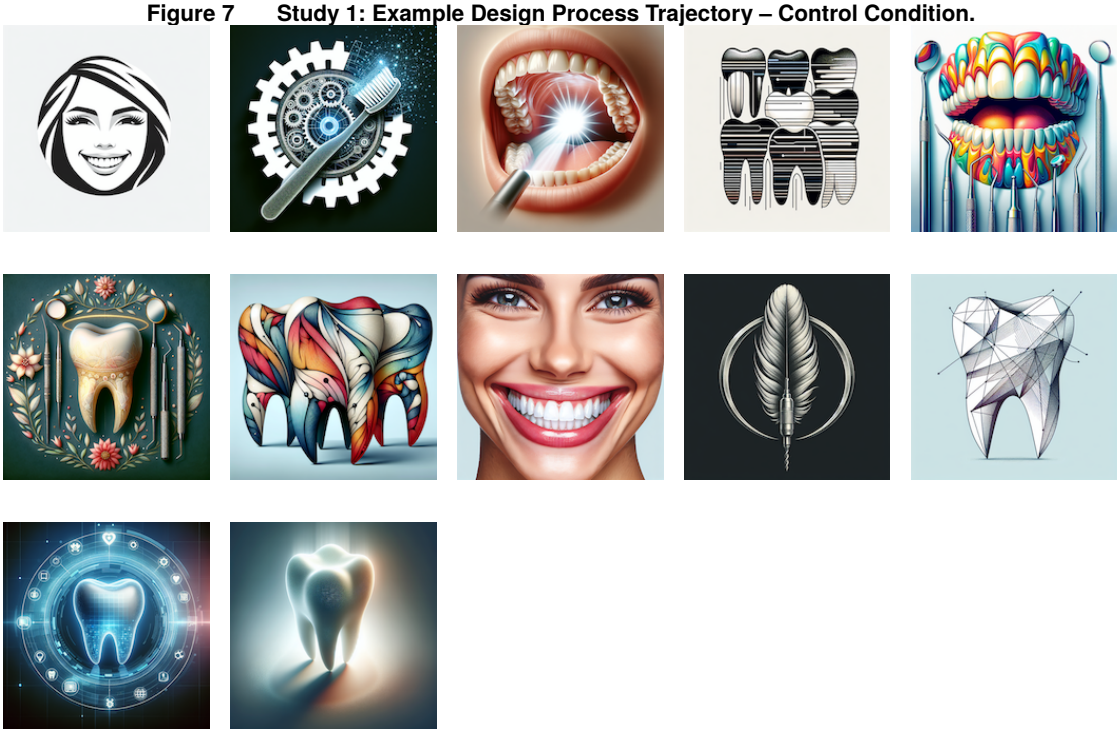
Notes. All metrics use prompt-based CLIP-vectorization. Standard errors are in parenthesis. *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$.

similarity *First–Last* is significantly higher in the treatment condition, i.e. when sketching, confirming our earlier results (Study 1: $p = 0.020$, Study 2: $p = 0.001$). This also holds when controlling for *AI Experience* in regressions in Table 44 (Study 1: $p = 0.021$, Study 2: $p = 0.003$)

F. Examples

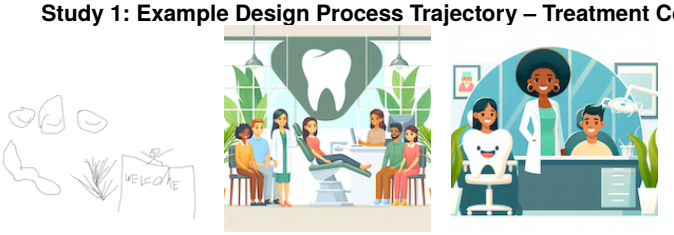
Here, we provide examples of the design process the outcomes. To illustrate the process, we display trajectories, i.e. the sequence of created designs for randomly sampled subjects in the treatment and control condition. For the treatment condition, we also display their sketch. Figure 7, Figure 8, Figure 9 and Figure 11 present trajectories from Study 1. Figure 12 and

Figure 7 **Study 1: Example Design Process Trajectory – Control Condition.**



Note. Designs by a randomly chosen subject, displayed in order of creation.

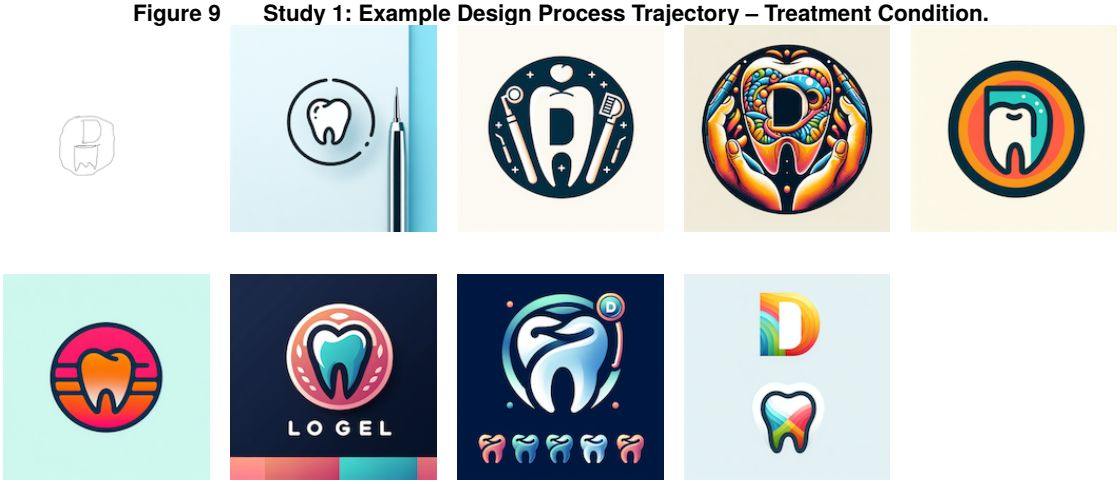
Figure 8 **Study 1: Example Design Process Trajectory – Treatment Condition.**



Note. Designs by a randomly chosen subject, displayed in order of creation.

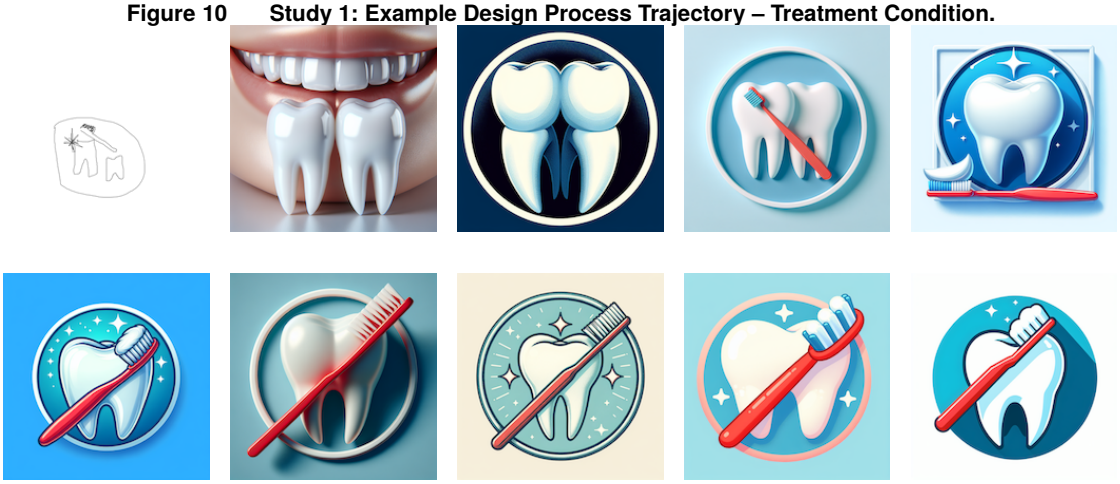
Figure 13 present trajectories from Study 2. To illustrate the products of the design process and the crowd evaluations, we display exemplary designs with their crowd-determined Elo rating in Figure 14 for Study 1 and in Figure 15 for Study 2. Note that here we show samples of the entire pools of created designs, i.e. not only designs that were ultimately chosen by their respective designers.

Figure 9 **Study 1: Example Design Process Trajectory – Treatment Condition.**



Note. Designs by a randomly chosen subject, displayed in order of creation.

Figure 10 **Study 1: Example Design Process Trajectory – Treatment Condition.**



Note. Designs by a randomly chosen subject, displayed in order of creation.

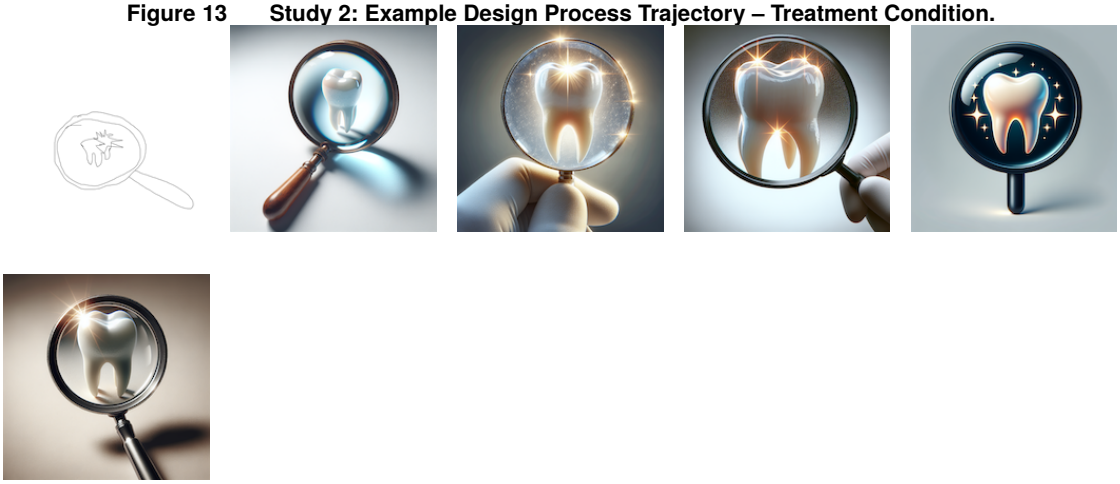
Figure 11 **Study 1: Example Design Process Trajectory – Control Condition.**

Note. Designs by a randomly chosen subject, displayed in order of creation.

Figure 12 **Study 2: Example Design Process Trajectory – Control Condition.**

Note. Designs by a randomly chosen subject, displayed in order of creation.

Figure 13 **Study 2: Example Design Process Trajectory – Treatment Condition.**



Note. Designs by a randomly chosen subject, displayed in order of creation.

Figure 14 Study 1: Elo Scores of Exemplary Designs.

(a) 1505 (Rank 1) (b) 1488 (Rank 2) (c) 1486 (Rank 3) (d) 1473 (Rank 4) (e) 1470 (Rank 5)

(f) 1468 (Rank 6) (g) 1465 (Rank 7) (h) 1460 (Rank 8) (i) 1458 (Rank 9) (j) 1453 (Rank 10)

(k) 1378 (Rank 102) (l) 1377 (Rank 107) (m) 1209 (Rank 658) (n) 1197 (Rank 705) (o) 1163 (Rank 834)

(p) 1144 (Rank 910) (q) 1133 (Rank 949) (r) 1049 (Rank 1165) (s) 1041 (Rank 1179) (t) 986 (Rank 1265)

(u) 865 (Rank 1335) (v) 864 (Rank 1336) (w) 859 (Rank 1337) (x) 855 (Rank 1338) (y) 831 (Rank 1339)

Note. We show the top 10 designs, 10 randomly sampled and the 5 worst designs by Elo. Elo scores and rankings are above each design.

Figure 15 Study 2: Elo Scores of Exemplary Designs.

(a) 1624 (Rank 1) (b) 1548 (Rank 2) (c) 1547 (Rank 3) (d) 1546 (Rank 4) (e) 1541 (Rank 5)

(f) 1539 (Rank 6) (g) 1533 (Rank 7) (h) 1531 (Rank 8) (i) 1531 (Rank 9) (j) 1531 (Rank 10)

(k) 1418 (Rank 117) (l) 1401 (Rank 161) (m) 1213 (Rank 651) (n) 1156 (Rank 799) (o) 1136 (Rank 838)

(p) 1130 (Rank 856) (q) 1118 (Rank 884) (r) 1074 (Rank 971) (s) 921 (Rank 1194) (t) 844 (Rank 1240)

(u) 713 (Rank 1270) (v) 705 (Rank 1271) (w) 694 (Rank 1272) (x) 675 (Rank 1273) (y) 630 (Rank 1274)

Note. We show the top 10 designs, 10 randomly sampled and the 5 worst designs by Elo. Elo scores and rankings are above each design.