Position Paper: Validatory Visualization for Human-GenAI Co-Creation

Yu Zhang* University of Oxford
Yuheng Zhao Fudan University
Yuhan Guo Peking University
Guozheng Li Beijing Inst. of Tech.
Siming Chen Fudan University
Xiaoru Yuan Peking University

Abstract

This paper focuses on the concept of validatory visualization in generative AI applications. As generative AI becomes increasingly widespread, it is essential to empower people to validate potentially large volumes of machine-generated content. We define validatory visualization as visualizations that support efficient validation of AI-generated content. To understand their benefits and guide their design, we introduce a model of interaction costs in validation and outline design considerations for creating effective validatory visualizations. Such visualizations have broad applicability, including in productivity tools and data analysis systems. We advocate for increased attention to validatory visualizations.

Index Terms: Generative AI, Validatory user interface, Validatory visualization, Human-AI co-creation, LLM, Agent.

1 Introduction

Machine learning algorithms inevitably make mistakes. Thus, the *validatory interaction paradigm*, where the machine generates content and the user validates it, deciding whether to accept or reject, has long been used in everyday applications such as grammar checkers [12]. With the rise of generative artificial intelligence (GenAI), the validatory interaction paradigm has been further popularized in applications that feature human-AI co-creation. An example is AI-assisted programming tools that allow the user to prompt the AI with natural language or partial code, which the AI then completes for the user to validate.

As GenAI becomes more powerful, attempts to automate tasks currently performed by humans are expected to grow. However, full automation is often infeasible due to intrinsic task properties, such as evolving requirements, subjective judgment, or task complexity. For many tasks, AI may produce useful outputs but fall short of fully satisfying human requirements. Therefore, human validation remains essential. Therefore, the validatory interaction paradigm will likely be a common solution across diverse applications.

The validatory interaction paradigm involves a loop of three steps: *steering* (user communicates intent), *generation* (AI generates output), and *validation* (user evaluates output). Among these steps, validation is the primary bottleneck in terms of time and cost, as it involves human judgment of AI-generated content. Designing efficient and effective *validatory user interfaces* is crucial.

A key challenge of validation is that GenAI can quickly generate large volumes of complex content. Visualizations, as a powerful tool for amplifying human information processing capabilities, can help address this challenge and play a crucial role in validatory user interfaces. We see rich opportunities to deploy *validatory visualizations* in GenAI-infused productivity tools facing mass consumers.

Beyond productivity tools, validatory user interfaces and visualizations may also benefit data analysis, the conventional focus of visualization research. There is growing interest in using GenAI to increase the level of automation in data analysis, through UI automation agents that operate on visualization systems [31], or data analysis agents that work directly in the data space (as exemplified in VIS x GenAI Workshop's Mini-Challenge). In this context, validatory user interfaces may enable users to quickly validate AI-generated partial or end-to-end data analysis results, along with the underlying analytical process.

Such interfaces represent a paradigm shift in many applications: the users' role moves from performing the task to guiding AI and validating how well AI completes the task. Inspired by Shneiderman's visual information-seeking mantra [21], "Overview first, zoom and filter, then details-on-demand", one may consider a mantra for human-GenAI co-creation: *Steering first, generate and validate, then agency on demand*. For tasks where GenAI excels, if the goal is to ensure efficiency, the user should be first presented with AI-generated results for validation. If necessary, the user may step in and resume a high level of agency in performing the task.

We suggest that validatory visualizations, as a key component of the validatory interaction paradigm, deserve more attention in visualization research. Validatory visualizations not only have the potential to expand the scope of visualization by reaching a wide audience of GenAI application users, but also benefit the existing focus of visualization on data analysis.

2 Background: Validation of AI Output

AI models are prone to errors, making human validation essential in quality-sensitive scenarios. We review conceptual models of human-AI interaction for validating AI outputs, which inform our model of validation steps and associated costs (Sec. 5).

In data-centric AI, validation is critical in data labeling systems to ensure the quality of AI-generated labels. Zhang et al. model the costs of such quality assurance processes, focusing on human-AI interactions in labeling, validation, and correction [29, 30].

In GenAI, validation also plays a central role in ensuring the quality of AI-generated content. Glassman characterizes the validation as a human-AI communication loop with steps such as expressing intent and recognizing alignment [10]. Gordon et al. simplify Glassman's model into a prompt-response-audit loop and highlight the importance of tools that allow users to "co-audit" AI-generated content [12]. Similarly, Karpathy argues that successful AI applications rely on an efficient generation-verification loop [14].

Across these models, quality assurance of AI output can be abstracted as a three-step loop: steering (user communicates intent), generation (AI generates output), and validation (user evaluates output). We refer to it as the *validatory interaction paradigm*.

While both steering and validation involve the user and demand effective interfaces, we focus on validation. Steering has received wide attention, covering topics such as prompt engineering [23] and intent disambiguation [28]. By contrast, the potential of visualization to ease validation across various tasks, particularly those encountered by ordinary users, has yet to be fully unlocked.

3 Define Validatory Visualization

To clarify our argument, we first define the notions of validatory user interface and validatory visualization, and then contrast them with related concepts.

*e-mail: yuzhang94@outlook.com

- **Validatory user interface**[1] refers to user interfaces for users to inspect and decide whether machine-generated content[2] is fit for their task, intent, and standards.
- **Validatory visualization** refers to visualizations for validating machine-generated output. They serve as components of validatory user interfaces, providing a visual overview of the machine-generated content and visual evidence about aspects such as provenance and uncertainty of the content.

Two concepts related to *validatory visualization* are *communicative visualization* and *confirmatory visualization*. All these three concepts stand in contrast to *exploratory visualization*.

- *Communicative visualization* refers to visualizations for communicating information, such as a finding in a dataset [20], rather than for supporting open-ended exploration. Validatory visualization also involves communicating information, specifically of machine-generated output, but the two differ in the goal and epistemic authority. Validatory visualization emphasizes efficiency and effectiveness in enabling user validation, whereas communicative visualization prioritizes persuasiveness and knowledge transfer. Validatory visualization places authority with the user, who decides whether the output is acceptable, whereas communicative visualization places authority with the producer, whose expertise shapes the message.
- *Confirmatory visualization* (or *confirmative visualization*) refers to visualizations used in confirmatory analysis, where the goal is to confirm or reject hypotheses formed by the user. Validatory visualization similarly involves confirmation, but the entity being validated is machine-generated content rather than the hypotheses. Compared with confirmatory visualization, the content that validatory visualization may validate is more diverse and is sourced from the machine instead of from the user.

Validatory visualization is also related to but distinct from visualization for explainable AI (XAI). *Visualization for XAI* is process-centric and focuses on facilitating a mechanistic understanding of models. It concerns why a model produces a certain output. In contrast, *validatory visualization* is output-centric and focuses on enabling users to validate model output. It concerns whether a model output is acceptable. At the same time, validatory visualization can benefit from techniques developed for XAI, as understanding how the model arrives at the output may help users validate it.

4 Opportunities

GenAI presents great opportunities for validatory visualization in both productivity tools and data analysis.

4.1 Validatory Visualization for Productivity Tools

GenAI's integration into productivity tools allows instant generation of large amounts of content. It creates a growing burden on users to validate model outputs. For example, a coding agent may edit thousands of lines of code in a short time, requiring substantial effort from a programmer to validate. Such scenarios highlight the need for techniques that amplify human information processing capabilities to keep pace with GenAI.

Visualization, long recognized as a cognitive amplifier [6], is particularly relevant in this context. Moreover, validation tasks present a unique opportunity for visualization to reach a broad audience of GenAI-infused productivity tool users.

In fact, visualization is already used for validation in tools such as Cursor [2], which uses color encoding to highlight AI-generated code, helping users quickly recognize changes. Yet questions remain: is the current visual design for showing AI-generated code diffs optimal? When multiple files are edited, can we go beyond the textual summaries of changes that current tools provide? Prior research on software visualization [22] offers a rich foundation for addressing these questions.

This example of AI-assisted programming also highlights a recurring user task in human-GenAI co-creation: understanding what changes the AI has made to an artifact, which is a prerequisite for validation. Efficiently conveying such information can be challenging for complex or extensive changes, and may benefit from visual summarization and comparison techniques [11].

Another challenge is designing workflows for validating AI-completed multi-step tasks, such as deep research or UI automation. Dividing the task into multiple checkpoints for user validation may reduce error propagation and enhance user agency. However, excessive user involvement increases cognitive and time costs. The visual analytics community has long studied workflows for AI-assisted data analysis, and the insights gained may be transferable to the design of validation workflows.

4.2 Validatory Visualization for Data Analysis

Efforts to shift exploratory data analysis from a primarily manual process to a mixed-initiative process predate the recent surge of GenAI. An early example is the Voyager system [27], which recommends visualizations according to statistical and perceptual measures. GenAI, particularly agentic AI, shows great promise in advancing this paradigm shift. UI automation agents [31] and data analysis agents [32] can highlight potentially interesting data facets or directly suggest tentative analysis results.

This paradigm shift introduces the need for analysts to validate AI-generated content, including the final outcomes and possibly the intermediate steps. In this case, visualization serves not only as the outcomes of AI-initiated data analysis, but also as a means to inspect and manipulate provenance, i.e., the AI's reasoning process and operations in the visualization pipeline [15, 33, 24].

5 Costs in the Validatory Interaction Paradigm

To better understand the premise of these opportunities, we use a simple interaction cost model to reason about when the validatory interaction paradigm reduces effort.

5.1 Setup

We consider the interaction cost for a user to complete a content generation task, where the goal is to generate output with acceptable quality (**A1** in Sec. 5.4). The cost may correspond to the task completion time or some other one-dimensional metric (**A2** in Sec. 5.4).

5.2 Model

We consider a simplified model of multi-round human-GenAI co-creation informed by the workflows discussed in Sec. 2, and particularly Gordon et al.'s prompt-response-audit cycle [12]. We define cost ($c_{\text{action}}^{\text{agent}}$)[3] and probability ($p_{\text{event}}$) parameters associated with steps in the model, as listed in Tab. 1. The model is not intended to fully capture all factors in real-world scenarios. Rather, it serves as

[1] We considered two naming alternatives: "validate" and "verify". We prefer "validate" as it emphasizes *assessing appropriateness* (e.g., meeting user needs), while "verify" emphasizes assessing correctness [5].

[2] We use "content" to refer to any artifact that can be externalized. *Such artifacts include insights in data analysis.* "Insight" has been defined as utterances, data facts, hypotheses, or knowledge links [3]. These definitions share the notion that insights can be externalized and represented in a data structure [4]. Thus, it is valid to say that "the AI presents tentative insights".

[3] The superscript "agent" refers to the party (human or machine) primarily responsible for an action. The other party may also participate in the action. For example, for the validation step, the machine may conduct a self-validation before user validation takes place.

Table 1: Cost ($c^{\text{agent}}_{\text{action}}$) and probability ($p_{\text{event}}$) parameters in human-GenAI co-creation.

Parameter	Description
$c^{\text{user}}_{\text{steer}}$	Cost for the user to steer the machine (e.g., provide a prompt or parameters).
$c^{\text{machine}}_{\text{generate}}$	Cost for the machine to generate or edit content, comprising the cost of computation (e.g., API usage) and user wait time.
$c^{\text{user}}_{\text{validate}}$	Cost for the user to validate if machine-generated content is acceptable.
$c^{\text{user}}_{\text{refine}}$	Cost for the user to refine unacceptable machine-generated content to make it acceptable.
$c^{\text{user}}_{\text{generate}}$	Cost for the user to generate content of acceptable quality from scratch without machine assistance.
p_{complete}	Probability that machine-generated content is acceptable.
p_{refine}	Probability that machine-generated content is unacceptable and the user decides to refine it to an acceptable form.

a tool to examine the key factors affecting the cost. In this model, the steps of human-GenAI co-creation[4] are as follows:

1. **Steering:** The user requests the machine to generate or edit content. Conveying the request requires providing input, such as a prompt or a set of parameters, to steer the machine through conversational or direct manipulation interfaces. (cost: $c^{\text{user}}_{\text{steer}}$)
2. **Generation:** The machine generates a piece of content based on the user's request. (cost: $c^{\text{machine}}_{\text{generate}}$)
3. **Validation:** The user validates whether the machine-generated content is acceptable in a validatory interface. (cost: $c^{\text{user}}_{\text{validate}}$)
   - **Branch 1:** With probability p_{complete}, the content is acceptable. The task is completed.
   - **Branch 2:** With probability p_{refine}, the content is unacceptable and the user refines it to an acceptable form (e.g., when the model output is nearly acceptable or when the user recognizes that the model is not capable of generating an acceptable result directly). The task is completed. (cost: $c^{\text{user}}_{\text{refine}}$)
   - **Branch 3:** With probability $1-p_{\text{complete}}-p_{\text{refine}}$, the content is unacceptable. Go to the steering step.

Assume the values of parameters listed above are constant across rounds of interaction (**A4** in Sec. 5.4). Let $p = p_{\text{complete}} + p_{\text{refine}}$. The probability that the user needs n rounds of interaction to obtain acceptable content without refinement is $(1-p)^{n-1}p_{\text{complete}}$. The probability that the user needs n rounds of interaction to obtain content that is unacceptable but can be refined to an acceptable form is $(1-p)^{n-1}p_{\text{refine}}$. Let $c_{\text{loop}} = c^{\text{user}}_{\text{steer}} + c^{\text{machine}}_{\text{generate}} + c^{\text{user}}_{\text{validate}}$ be the cost of one round of steering, generation, and validation. The expected total cost to complete the task with human-GenAI co-creation is:

$$\begin{aligned} C_{\text{co}} &= \sum_{n=1}^{\infty}(1-p)^{n-1}(pnc_{\text{loop}} + p_{\text{refine}}c^{\text{user}}_{\text{refine}}) \\ &= \frac{1}{p}c_{\text{loop}} + \frac{1}{p}p_{\text{refine}}c^{\text{user}}_{\text{refine}} \end{aligned}$$

5.3 Analysis

We introduce $C_{\text{manual}} = c^{\text{user}}_{\text{generate}}$ to denote the cost of completing the task manually without machine assistance.

When is machine assistance beneficial? $C_{\text{co}} < C_{\text{manual}}$ holds when $c_{\text{loop}} + p_{\text{refine}}c^{\text{user}}_{\text{refine}} < pc^{\text{user}}_{\text{generate}}$. In other words, machine assistance is beneficial for tasks where the machine is reasonably skilled at (i.e., with large p) and the cost to manually generate the content is much higher than the cost to steer, validate, and refine. Note that a necessary condition for machine assistance to be beneficial is $c^{\text{user}}_{\text{validate}} < c^{\text{user}}_{\text{generate}}$.

What bounds the benefit of machine assistance? Let $r \in (-\infty, 1]$ denote the rate at which machine assistance reduces the total cost relative to manual content generation.

$$r = 1 - C_{\text{co}}/C_{\text{manual}} = 1 - \frac{c_{\text{loop}} + p_{\text{refine}}c^{\text{user}}_{\text{refine}}}{pc^{\text{user}}_{\text{generate}}}$$

- For fixed costs ($c^{\text{agent}}_{\text{action}}$) and varying probabilities (p_{event}), the supremum of r is $1 - c_{\text{loop}}/c^{\text{user}}_{\text{generate}}$, approached when $p \to 1$ and $p_{\text{refine}} = 0$.
- For fixed probabilities (p_{event}) and varying costs ($c^{\text{agent}}_{\text{action}}$), the supremum of r is 1, approached when $c_{\text{loop}} \to 0$ and $c^{\text{user}}_{\text{refine}} \to 0$.

These bounds suggest that the benefit of improving the model performance (i.e., increasing p) is limited by the validatory interface design, which influences the costs. Conversely, the benefit of improving the validatory interface design (i.e., reducing c_{loop} and $c^{\text{user}}_{\text{refine}}$) is not limited by the model performance. Even with a mediocre model, it is possible to achieve a significant reduction in the total cost by improving the interface to empower the user. It highlights the opportunity for visualization research to design cognitive amplifiers that maximize the benefits of machine assistance.

Special parameter setups. The model can represent specific application scenarios by setting the parameter values, such as:

- **Vibe coding:** Setting $p_{\text{refine}} = 0$ models the vibe coding setup where the user only validates the functionalities of machine-generated code and never edits it directly.
- **Data labeling:** Setting $p \to 1$ models the setup of data labeling with machine-generated default labels where the user either accepts or refines the machine-generated label and never asks the machine to regenerate it. In this case, multi-round interaction degenerates to a single round.

Compare $c^{\text{user}}_{\text{validate}}$ and $c^{\text{user}}_{\text{generate}}$. We consider two perspectives:

- **Complexity theory perspective:** $P \subseteq NP$ formalizes that for some problems, verifying a solution is computationally easier than generating a solution. Thus, validating machine-generated candidate solutions may require less effort from the user than solving the problem (i.e., $c^{\text{user}}_{\text{validate}} < c^{\text{user}}_{\text{generate}}$). However, counterexamples exist. For instance, generating a program that halts (e.g., "print(1)") is easy, but validating whether an arbitrary program halts is undecidable. Thus, converting a generation task to a validation task does not guarantee to reduce user effort[5].
- **Machine learning perspective:** Reinforcement learning from human feedback (RLHF) [9] aligns generative models with human preferences. With RLHF, human annotators guide a model to generate desired content by providing feedback on the quality

[4] We adopt a broad definition of "co-creation" that *considers steering as a form of co-creation*. Even if the user does not directly edit the artifact, the intent conveyed through steering shapes the artifact.

[5] *Even for problems where generating a solution is easier than verifying one, the validatory interaction paradigm may still bring benefit.* For example, instead of requiring the user to type "print(1)" manually, the machine may generate this string for the user to validate, which may save time. The problem arises when the machine is not competent in generating easy-to-validate content.

of model outputs, without having to generate the desired content themselves. This approach assumes evaluation is easier than generation, referred to as the asymmetry of verification [25]. It can be seen as assuming $c^{\text{user}}_{\text{validate}} < c^{\text{user}}_{\text{generate}}$.

5.4 Simplifying Assumptions

The model relies on the following simplifying assumptions, which restrict the focused scenarios and indicate where the model may diverge from real-world situations:

- **A1: Output-centric and cost-sensitive task.** The model focuses on content generation tasks. It does not consider scenarios where the outcome cannot be externalized (e.g., learning a skill or entertainment). Such scenarios may not produce outputs, or directing users to AI-generated outputs may undermine the goal. Another presumption is that the task is cost-sensitive, making it meaningful to model the cost and preferable to reduce it.
- **A2: One-dimensional cost metric.** We assume the interaction cost can be represented by a single metric. In reality, the cost may be multi-dimensional (e.g., time, monetary cost, and cognitive effort). It may be infeasible to collapse them into one metric.
- **A3: No interleaving of refine and steer.** We assume that once the user decides to refine the machine-generated content, the user will no longer steer the machine to update the content or generate new content. In practice, the user may alternate between refining and steering. For example, in AI-assisted programming, the user may refactor code (i.e., refine) and then prompt the machine to generate code for a new feature (i.e., steer).
- **A4: Constant parameter values.** We assume the parameter values remain constant across rounds of interaction. In practice, learning effects or user fatigue may change the costs of user operations. The probability for the machine to generate acceptable content may increase as it gains more context about the user intent, or it may decrease as repeated failures may indicate that the machine is not capable of producing acceptable content due to task difficulty.

6 Design Considerations

Existing guidelines on mixed-initiative user interfaces and human-AI interaction [13, 1, 26] remain relevant to validatory user interfaces. Gordon et al. [12] discuss principles for designing co-audit systems that are directly applicable. We refer readers to these existing guidelines and highlight a few additional considerations.

Designing for ordinary users: GenAI empowers users to perform tasks that previously required specialized expertise or significant effort. As GenAI is increasingly used in consumer applications, validatory visualizations must account for users with limited visualization literacy.

Asynchronous interaction: Latencies in GenAI models can make real-time interaction impractical. Validatory interfaces should support asynchronous interaction to avoid blocking users during model processing. Strategies include and are not limited to:

- **Parallelize tasks:** Allow users to perform parallel tasks while waiting for model output.
- **Anticipate user actions:** Predict user intent and pre-generate results proactively [7, 31].
- **Interact with streaming output:** Render partial or streaming model output to allow interaction before the result is complete.
- **Merge human and AI edits:** Enable simultaneous edits by the user and the machine, with conflict resolution mechanisms.
- **Provide contextual cues:** Help users recall their prior requests and intent to validate AI outputs effectively.

Mitigating automation bias: Automation introduces biases in human decision-making [19]. Users tend to accept default options provided by the system [18]. Validatory interfaces should mitigate these biases and encourage critical evaluation of machine-generated content. Strategies include and are not limited to:

- **Show uncertainty:** Communicate model output uncertainty to avoid over-reliance [16].
- **Offer alternatives:** Present multiple options for users to choose from, rather than a single default choice.

7 Discussion

While the validatory interaction paradigm can empower users, it also introduces a tension between efficiency and deeper forms of human engagement, along with their associated benefits. This tension manifests in the following potential losses.

Loss of context: The validatory interaction paradigm assumes that if a user can complete a task without AI, the user can validate AI outputs for the task. One may question whether this premise holds for exploratory tasks. In exploratory tasks, the user gains experience and context through exploration. Without such context, the user may not be fully capable of validating AI outputs. For example, in data analysis, the user may be unable to determine what data facts are valuable without first exploring the dataset.

Loss of serendipity: Exploration sometimes yields unexpected discoveries. A validation-focused workflow prioritizes efficiency, which can reduce opportunities for serendipitous discoveries.

Loss of creativity: Immersion in AI-generated content may hinder creativity. Chiang argues that art arises from making countless deliberate choices that reflect the creator's intent and experience [8]. Offloading tasks to AI reduces opportunities for such deliberate choices, thereby constraining the space for innovation.

Loss of frictions: Litt argues that increasing automation introduces a loss of frictions [17]. Frictions and inconvenience create opportunities for meaning and human connection in scenarios such as reading and discussion. The value sometimes lies not only in the outcome but also in the journey.

These potential losses suggest that the validatory interaction paradigm may not be preferable for every task, at every stage of a task, or for every user. Systems should consider allowing flexible configurations of user agency so that the user can step when needed. Some tasks involve both output-centric and non-output-centric goals, which require careful consideration of user agency. For example, AI-assisted note-taking may save time, but it can also negatively impact the learning experience involved in taking notes. Additionally, validatory user interfaces for exploratory tasks may provide provenance information as context, showing not only the tentative results but also how those results were derived.

8 Conclusion

In this paper, we position validatory visualization as an essential component of human-GenAI co-creation. While generative AI can accelerate content creation across applications such as productivity tools and data analysis, it also introduces the need for validation. We argue that visualization can play a pivotal role in this process by amplifying human information processing capabilities and enabling scalable review of AI outputs. The interaction cost modeling highlights the importance of designing effective interfaces to maximize the benefits of machine assistance. While validatory visualization offers opportunities, it also raises new design challenges and potential issues. It deserves systematic attention in future research, both to safeguard the reliability of GenAI-assisted workflows and to broaden the reach of visualization as a discipline in everyday human-AI interaction.

References

[1] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen, J. Teevan, R. Kikin-Gil, and E. Horvitz. Guidelines for human-AI interaction. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, May 2019. doi: 10.1145/3290605.3300233 4

[2] Anysphere. Cursor. https://cursor.com/, 2023. Accessed: 2025-09-02. 2

[3] L. Battle and A. Ottley. What exactly is an insight? a literature review. In *Proceedings of the IEEE Visualization and Visual Analytics Conference*, pp. 91–95. IEEE, Oct. 2023. doi: 10.1109/VIS54172.2023.00027 2

[4] L. Battle and A. Ottley. What do we mean when we say "insight"? a formal synthesis of existing theory. *IEEE Transactions on Visualization and Computer Graphics*, 30(9):6075–6088, Sept. 2024. doi: 10.1109/TVCG.2023.3326698 2

[5] B. W. Boehm. Verifying and validating software requirements and design specifications. *IEEE Software*, 1(1):75–88, Jan. 1984. doi: 10.1109/MS.1984.233702 2

[6] S. K. Card, J. D. Mackinlay, and B. Shneiderman. Information visualization. In S. K. Card, J. D. Mackinlay, and B. Shneiderman, eds., *Readings in Information Visualization: Using Vision to Think*, pp. 1–34. Morgan Kaufmann Publishers, 1999. 2

[7] V. Chen, A. Zhu, S. Zhao, H. Mozannar, D. Sontag, and A. Talwalkar. Need help? designing proactive AI assistants for programming. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, 2025. doi: 10.1145/3706598.3714002 4

[8] T. Chiang. Why A.I. isn't going to make art. *The New Yorker*, Aug. 2024. 4

[9] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. In *Proceedings of the International Conference on Neural Information Processing Systems*, pp. 4302–4310. Curran Associates Inc., 2017. 3

[10] E. L. Glassman. Designing interfaces for human-computer communication: An on-going collection of considerations. 2023. doi: 10.48550/ARXIV.2309.02257 1

[11] M. Gleicher. Considerations for visualizing comparison. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):413–423, Jan. 2018. doi: 10.1109/TVCG.2017.2744199 2

[12] A. D. Gordon, C. Negreanu, J. Cambronero, R. Chakravarthy, I. Drosos, H. Fang, B. Mitra, H. Richardson, A. Sarkar, S. Simmons, J. Williams, and B. Zorn. Co-audit: tools to help humans double-check AI-generated content. 2023. doi: 10.48550/ARXIV.2310.01297 1, 2, 4

[13] E. Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 159–166. ACM, 1999. doi: 10.1145/302979.303030 4

[14] A. Karpathy. Software is changing (again). https://www.youtube.com/watch?v=LCEmiRjPEtQ, 2025. Accessed: 2025-09-02. 1

[15] M. Kazemitabaar, J. Williams, I. Drosos, T. Grossman, A. Z. Henley, C. Negreanu, and A. Sarkar. Improving steering and verification in AI-assisted data analysis with interactive task decomposition. In *Proceedings of the ACM Symposium on User Interface Software and Technology*. ACM, 2024. doi: 10.1145/3654777.3676345 2

[16] Q. V. Liao and J. Wortman Vaughan. AI transparency in the age of LLMs: A human-centered research roadmap. *Harvard Data Science Review*, (Special Issue 5), Feb. 2024. doi: 10.1162/99608f92.8036d03b 4

[17] G. Litt. AI as teleportation. https://www.geoffreylitt.com/2025/09/10/ai-as-teleportation, Sept. 2025. Accessed: 2025-10-01. 4

[18] B. C. Madrian and D. F. Shea. The power of suggestion: Inertia in 401(k) participation and savings behavior. *The Quarterly Journal of Economics*, 116(4):1149–1187, Nov. 2001. doi: 10.1162/003355301753265543 4

[19] R. Parasuraman and V. Riley. Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39(2):230–253, June 1997. doi: 10.1518/001872097778543886 4

[20] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, Nov. 2010. doi: 10.1109/TVCG.2010.179 2

[21] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pp. 336–343. IEEE, 1996. doi: 10.1109/VL.1996.545307 1

[22] J. T. Stasko, J. B. Domingue, M. H. Brown, and B. A. Price, eds. *Software Visualization: Programming As a Multimedia Experience*. MIT Press, Cambridge, MA, USA, 1998. 2

[23] H. Strobelt, A. Webson, V. Sanh, B. Hoover, J. Beyer, H. Pfister, and A. M. Rush. Interactive and visual prompt engineering for ad-hoc task adaptation with large language models. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1146–1156, Jan. 2023. doi: 10.1109/TVCG.2022.3209479 1

[24] C. Wang, B. Lee, S. M. Drucker, D. Marshall, and J. Gao. Data Formulator 2: Iterative creation of data visualizations, with AI transforming data along the way. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, 2025. doi: 10.1145/3706598.3713296 2

[25] J. Wei. Asymmetry of verification and verifier's rule. https://www.jasonwei.net/blog/asymmetry-of-verification-and-verifiers-law, July 2025. Accessed: 2025-09-02. 4

[26] J. D. Weisz, J. He, M. Muller, G. Hoefer, R. Miles, and W. Geyer. Design principles for generative AI applications. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, 2024. doi: 10.1145/3613904.3642466 4

[27] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, Jan. 2016. doi: 10.1109/TVCG.2015.2467191 2

[28] M. J. Q. Zhang and E. Choi. Clarify when necessary: Resolving ambiguity through interaction with LMs. In L. Chiruzzo, A. Ritter, and L. Wang, eds., *Findings of the Association for Computational Linguistics: NAACL*, pp. 5526–5543. Association for Computational Linguistics, Apr. 2025. doi: 10.18653/v1/2025.findings-naacl.306 1

[29] Y. Zhang, B. Coecke, and M. Chen. On the cost of interactions in interactive visual machine learning. In *Proceedings of the IEEE VIS Workshop on Evaluation of Interactive Visual Machine Learning Systems*. IEEE, 2019. 1

[30] Y. Zhang, M. Tennekes, T. De Jong, L. Curier, B. Coecke, and M. Chen. Simulation-based optimization of user interfaces for quality-assuring machine learning model predictions. *ACM Transactions on Interactive Intelligent Systems*, 14(1), Jan. 2024. doi: 10.1145/3594552 1

[31] Y. Zhao, X. Shu, L. Fan, L. Gao, Y. Zhang, and S. Chen. ProactiveVA: Proactive visual analytics with LLM-based UI agent. *IEEE Transactions on Visualization and Computer Graphics*, 2025. 1, 2, 4

[32] Y. Zhao, J. Wang, L. Xiang, X. Zhang, Z. Guo, C. Turkay, Y. Zhang, and S. Chen. LightVA: Lightweight visual analytics with LLM agent-based task planning and execution. *IEEE Transactions on Visualization and Computer Graphics*, 31(9):6162–6177, 2025. doi: 10.1109/TVCG.2024.3496112 2

[33] Y. Zhao, Y. Zhang, Y. Zhang, X. Zhao, J. Wang, Z. Shao, C. Turkay, and S. Chen. LEVA: Using large language models to enhance visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 31(3):1830–1847, Mar. 2025. doi: 10.1109/TVCG.2024.3368060 2