Task-Aligned Layered Evaluation for Generative Visualizations

Eunhye Kim*† KAIST
Meng Chen*‡ University of California, Berkeley
Yue Jiang§ University of Utah

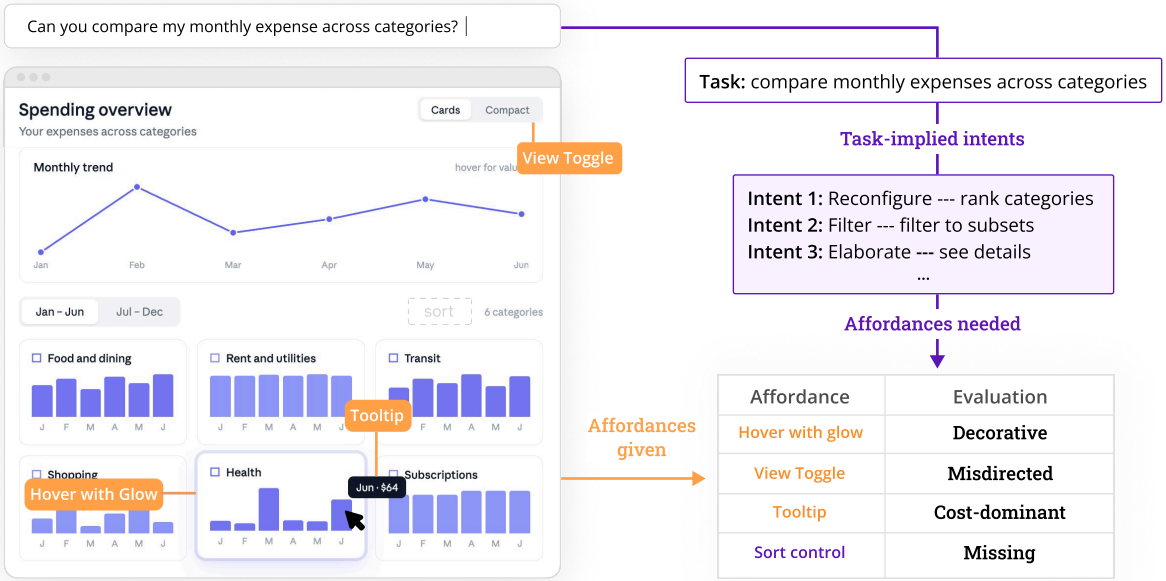


Figure 1: A dashboard generated for the task "compare monthly expenses across categories." All of its controls work, but the hover glow is decorative, the tabs misdirect, the tooltips cost more than they give, and sorting is missing. Mapping generated affordances to task-implied intents surfaces failures that an overall quality judgment can miss.

Abstract

Large language models can now generate interactive visualizations on demand for data analysis, digital marketing, and education. Yet, how should we evaluate their quality? Prior work has evaluated generative visualizations through properties such as visual aesthetics and functional correctness. However, an interaction can work as implemented while still being poorly suited to the user's visualization task. For example, a generated chart may provide filtering controls that function correctly but fail to support the comparisons the user actually needs to make. We argue that to properly evaluate interaction within visualizations, we need to think along the chain of task, intent, affordance, and cost. We characterize four common failures: *decorative*, where interaction adds little task value; *misdirected*, where it supports the wrong analytical intent; *missing*, where an important interaction is absent; and *cost-dominant*, where the effort required outweighs its benefit. These failure modes motivate us to evaluate generative visualizations not only on whether they work, but also on how well they cover the interactions useful for the intended task and at what cost. We discuss how such task-level signals can complement existing evaluation methods and outline a research agenda for fine-grained evaluation and, ultimately, the shaping of interactions in generative visualizations.

Index Terms: Generative Visualization, Interaction, Task Alignment.

*These authors contributed equally.
†e-mail: gracekim027@kaist.ac.kr
‡e-mail: meng.chen@berkeley.edu
§e-mail: yue.jiang@utah.edu

1 Introduction

When asked about monthly spending, modern generative AI systems can do more than answer in prose. They can synthesize an interactive dashboard with charts, summary cards, tabs, and toggles tailored to the request. This capability, often described as generative UI (GenUI), has rapidly moved from research prototypes toward deployed systems [5, 13, 2, 9]. Among the interfaces these systems can produce are dashboards, infographics, and interactive data displays, bringing visualization into the output space of general-purpose generative models [11, 8, 6].

While the interface produced may look polished, there is no guarantee that the interactions within the generated visualization support the user's task. Consider a dashboard in which cards lift on hover and related values are separated across tabs, while the data cannot be sorted or filtered. Each generated interaction may function exactly as intended, and the user may even be able to complete the task, yet the interactions most useful for analytical comparison remain poorly suited to how that task should be carried out. The problem, therefore, is not simply whether an interface is interactive or whether its controls work, but whether the interactions it provides are appropriate for the user's underlying analytical goals.

Existing evaluations of generative visualizations have made substantial progress in assessing properties of the artifact itself, includ-

ing validity, readability, and visual quality. These are important advances, but they primarily tell us about what was generated and whether it can be used or not. However, judging whether the right interaction was generated additionally requires understanding the user's task and the intents within, the affordance generated to support that intent, and the cost of using that affordance. For instance, a tooltip may allow the user to complete a comparison task, but if comparison requires the user to repeatedly reveal and remember values, the interaction may still be poor support. We argue that this relationship among *task*, *intent*, *affordance*, and *cost* remains comparatively under-operationalized in the evaluation of generative visualizations.

Visualization research provides a strong foundation for operationalizing this interaction layer. It offers intent-based theories of interaction [18], taxonomies connecting visualization tasks to analytic operations [15, 1, 3], frameworks for reasoning about interaction cost [10], and examples of visualization design knowledge being formalized and made computationally actionable [14]. Building on these foundations, we ask how knowledge about task, intent, and cost can be turned into an evaluation signal for interactions. We propose a taxonomy of interaction failures and an intent-coverage measure, situate this signal within a broader knowledge-grounded evaluation of generated visualization, and outline a research agenda for bringing visualization knowledge into GenUI evaluation and optimization.

2 What Current Signals Miss About Task-Aligned Interaction

Evaluation of generative visualizations is moving beyond static visual quality. Some systems generate executable interfaces with general-purpose models and evaluate their outputs through human preference [11, 4], while others use signals derived from rendered artifacts and designer judgments to improve UI generation [16]. More recent approaches additionally inspect dynamic behavior and interaction quality [19], and some evaluate whether generated interfaces support successful task execution [12]. Evaluation of generated visualizations has likewise made progress on properties such as validity, readability, and encoding correctness [7]. Together, these approaches broaden evaluation beyond static artifact quality toward functional and interactive properties.

Yet whether an interaction works is different from whether it is appropriate for the visualization task. A screenshot-based evaluator can assess the rendered artifact, structural or interaction-aware evaluation can inspect whether an affordance exists and functions, and task-based evaluation can test whether a specified goal can be completed. These approaches increasingly assess what was generated, but they are less direct at identifying what should have been generated instead. A functioning tab may be unnecessary for the task, while a task-relevant sorting operation that was never generated leaves no affordance to inspect. Evaluating such cases requires an additional account of interaction-to-task alignment.

Visualization research provides a basis for such an account through task taxonomies, intent-based theories of interaction, and interaction-cost frameworks [15, 1, 3, 18, 10]. Together, these perspectives suggest that interaction should be evaluated through the relationship among *task*, *intent*, *affordance*, and *cost*, rather than functionality alone. This gap motivates the interaction failure modes we sketch and our exploration of how visualization knowledge might support task-aligned interaction evaluation.

3 Task-Aligned Layered Evaluation for Generative Visualizations

A task can imply one or more analytic intents, a generative visualization can provide affordances that may support those intents, and the usefulness of those affordances depends partly on the cost of discovering and using them [18, 15, 1, 3, 10]. Visualization research has already shown, through systems such as Draco [17, 14], that design knowledge can be formalized and made computationally actionable. We ask whether a similar move can be made for interaction. Can knowledge about task, intent, affordance, and cost be operationalized into a signal for evaluating task alignment in generative visualizations? We explore this question by sketching a set of task-alignment failures in generated interaction grounded in existing visualization theory and use it to illustrate the shape of an intent-coverage signal. We then briefly situate this interaction signal within broader knowledge-grounded evaluation for generative visualizations, with the longer-term objective of using such signals to guide generation.

Throughout this proposal, we use a running example (Figure 1). Suppose a user asks a generative system to help them *compare monthly expenses across categories, identify the largest categories, and inspect what contributes to each total*. The resulting dashboard contains category cards that glow and lift on hover, divides related values across tabs, reveals category totals only through tooltips, and provides no way to sort the categories.

3.1 Task-Alignment Failures in Generated Interaction

The task–intent–affordance–cost relationship suggests several ways that a generated interaction can fail to align with a user's task. Drawing on prior work on visualization tasks, interaction intents, and interaction costs [18, 15, 1, 3, 10], we sketch out four failure modes. These categories provide a starting point for describing failures that functionality alone fails to capture.

A *decorative* interaction introduces an affordance without actually supporting a clear task-relevant intent. In our example, the hover animation emphasizes values that are already visible without enabling a relevant analytical operation. A *misdirected* interaction supports a recognizable user intent, but that intent may be misaligned with the current task. For example, the tabs support navigation between views, yet separating categories across views makes comparison more difficult. A *missing* interaction occurs when a task-relevant intent exists, but the generated output lacks an adequate corresponding affordance. Here, the user needs to rank categories but cannot sort them. Lastly, a *cost-dominant* interaction supports an appropriate intent, but at a cost that outweighs the value it provides. The tooltip reveals useful detail, yet forcing the user to repeatedly reveal and remember values needed for comparison can be costly.

Although these categories could be equally applied to human-designed visualizations, evaluating them in the context of generation is particularly important because interactions can now be synthesized without a designer explicitly reasoning through the relationship between each affordance and the task it is meant to support. The proposed taxonomy thus illustrates why generated interactions should be evaluated relative to the specific task and context in which they appear.

3.2 From Taxonomy to an Interaction Signal

This taxonomy suggests one possible way to operationalize task-aligned interaction. If generated interactions can be mapped to the intents they are designed to support, and a task is mapped to the intents the user likely will have, two familiar ideas can be useful for using this as an evaluation signal. *Interaction precision* can capture how much of the generated interaction is task-aligned, and *interaction recall* can capture whether task-relevant intents are adequately supported. Decorative and misdirected interactions would lower precision, while missing interactions would lower recall. Cost-dominant interactions suggest an additional weighting by the effort required to discover, execute, and interpret an affordance.

In our running example, the hover effect and tabs add interaction without adequately supporting comparison, while the absence of

sorting leaves a relevant operation unsupported. The tooltip nominally supports elaboration, but its contribution should be discounted when repeatedly revealing hidden values imposes more effort than it saves. The appropriate unit of interaction, how task-relevant intents should be identified, and whether experts, users, or models can annotate them reliably are all open questions. Task–intent coverage allows us to directly evaluate failures such as unnecessary and absent interaction rather than leaving them implicit in an overall quality judgment. Once operationalized, such a measure could serve as an evaluation signal and, eventually, as one component of a reward used to optimize generative models.

3.3 Interaction within Broader Knowledge-Grounded Evaluation

The interaction signal we describe is one component of a broader question—what evidence should be used to evaluate different decisions inside a generated visualization? Different design decisions may require different forms of evidence. For instance, content fidelity can be evaluated against the underlying data and request, while visual encoding can draw on established visualization knowledge on perceptual effectiveness and constraint-based approaches [14]. Interaction adds a different question of task alignment, requiring reasoning about the relationship among task, intent, affordance, and cost.

Preference remains valuable where visualization knowledge does not determine a single choice. When there are multiple encodings, interaction mechanisms, or visual styles that satisfy the relevant constraints, user preference can help distinguish among otherwise acceptable alternatives. We believe that established knowledge can provide guidance where the field has strong evidence, while preference can remain informative where multiple acceptable choices remain. Together, these signals could provide a more structured basis for evaluating, and eventually shaping, generative visualizations.

4 A Research Agenda

Realizing this vision is community-scale work, and we frame its first steps as questions for this workshop.

Formalization: Can interaction knowledge be encoded so that intent coverage is computable without manual human judgment?

Data: Can we build a dataset using this formalization of in-the-wild generated visualizations to ground the failure categories empirically and reveal whether models differ systematically in, for instance, their decorative tendencies?

Evaluation: Where do user preference, usage-based preference, and knowledge-grounded signals disagree, and which best predicts users' actual task performance? Can this provide insight into whether GenUI is being optimized against the right signal at all?

Training: Does finetuning against an intent-coverage reward measurably reduce decorative and missing interactions relative to preference-tuned baselines, as related work on knowledge-informed reward models for UI generation suggests it might [16]?

We believe the visualization community has spent decades learning what interaction is for, and the open question is whether we can leverage that knowledge to enter the training loop. Solving this for visualization would not only repair the field's own generated artifacts, but also demonstrate how generative UI as a whole can be grounded in established design knowledge.

References

[1] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pp. 111–117. IEEE, 2005. 2

[2] Anthropic. Introducing Claude 3.5 Sonnet. https://www.anthropic.com/news/claude-3-5-sonnet, June 2024. Introduces the Artifacts feature. Accessed August 12, 2026. 1

[3] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE transactions on visualization and computer graphics*, 19(12):2376–2385, 2013. 2

[4] Y. Cao, P. Jiang, and H. Xia. Generative and malleable user interfaces with generative and evolving task-driven data model. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–20, 2025. 2

[5] J. Chen, Y. Zhang, Y. Zhang, Y. Shao, and D. Yang. Generative interfaces for language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 1499–1519, 2026. 1

[6] M. Chen and A. Pavel. TaskArtisan: Designing Composable Generative Widgets for LLM-Assisted Analysis, July 2026. arXiv:2607.17394 [cs.HC]. doi: 10.48550/arXiv.2607.17394 1

[7] N. Chen, Y. Zhang, J. Xu, K. Ren, and Y. Yang. Viseval: A benchmark for data visualization in the era of large language models. *IEEE Transactions on Visualization and Computer Graphics*, 31(1):1301–1311, 2024. 2

[8] V. Dibia. Lida: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 113–126, 2023. 1

[9] E. Kim, D. Choi, B. Min, H. Yi, Y. Jiang, and J. Kim. Maru: Information architecture as a shared language for generating aligned and persistent user interfaces. *arXiv preprint arXiv:2608.25565*, 2026. 1

[10] H. Lam. A framework of interaction costs in information visualization. *IEEE transactions on visualization and computer graphics*, 14(6):1149–1156, 2008. 2

[11] Y. Leviathan, D. Valevski, M. Kalman, D. Lumen, E. Segalis, E. Molad, S. Pasternak, V. Natchu, V. Nygaard, J. Manyika, et al. Generative ui: Llms are effective ui generators. *arXiv preprint arXiv:2604.09577*, 2026. 1, 2

[12] K. Q. Lin, S. Hu, L. Li, Z. Yang, L. Wang, P. Torr, and M. Z. Shou. Computer-use agents as judges for generative user interface. *arXiv preprint arXiv:2511.15567*, 2025. 2

[13] Y. Liu, M. Sra, and C. Xiao. Crowdgenui: Aligning llm-based ui generation with crowdsourced user preferences. *arXiv preprint arXiv:2411.03477*, 2024. 1

[14] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, and J. Heer. Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco. *IEEE transactions on visualization and computer graphics*, 25(1):438–448, 2018. 2, 3

[15] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization*, pp. 364–371. Elsevier, 2003. 2

[16] J. Wu, A. Swearngin, A. Krishnavajjala, A. Leung, J. Nichols, and T. Barik. Improving user interface generation models from designer feedback. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2026. 2, 3

[17] J. Yang, P. F. Gyarmati, Z. Zeng, and D. Moritz. Draco 2: An extensible platform to model visualization design. In *2023 IEEE Visualization and Visual Analytics (VIS)*, pp. 166–170. IEEE, 2023. 2

[18] J. S. Yi, Y. ah Kang, J. Stasko, and J. A. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE transactions on visualization and computer graphics*, 13(6):1224–1231, 2007. 2

[19] C. Zhang, Y. Li, C. Xu, J. Liu, A. Liu, C. Zhou, K. Deng, D. Wu, G. Huang, K. Li, et al. Artifactsbench: Bridging the visual-interactive gap in llm code generation evaluation. *arXiv preprint arXiv:2507.04952*, 2025. 2