

DEMONSTRATING VIVIDOC: GENERATING INTERACTIVE DOCUMENTS THROUGH HUMAN-AGENT COLLABORATION

Yinghao Tang¹, Yupeng Xie², Yingchaojie Feng³, Tingfeng Lan⁴, Wei Chen^{1*}

¹State Key Lab of CAD&CG, Zhejiang University ²HKUST(GZ)

³National University of Singapore ⁴University of Virginia

Abstract

Interactive articles help readers engage with complex ideas through exploration, yet creating them remains costly, requiring both domain expertise and web development skills. Recent LLM-based agents can automate content creation, but naively applying them yields uncontrollable and unverifiable outputs. We present VIVIDOC, a human-agent collaborative system that generates interactive educational documents from a single topic input. VIVIDOC introduces a multi-agent pipeline (Planner, Executor, Evaluator) and the Document Specification (DocSpec), a human-readable intermediate representation that decomposes each interactive visualization into State, Render, Transition, and Constraint components. The DocSpec enables educators to review and refine generation plans before code is produced, bridging the gap between pedagogical intent and executable output. Expert evaluation and a user study show that VIVIDOC substantially outperforms naive agentic generation and provides an intuitive editing experience. Our project homepage is available at <https://vividoc-homepage.vercel.app/>.

1 Introduction

Interactive articles are an emerging communication medium that leverages the dynamic capabilities of the web to help readers engage with complex ideas (Victor, 2011). Through interactive elements such as sliders, dropdowns, and direct manipulation controls, readers can actively explore concepts, observe cause-and-effect relationships, and build intuition through experimentation. This form of communication has been adopted across diverse domains: explorable explanations in education (Chi and Wylie, 2014), data-driven storytelling in journalism, and interactive tutorials in scientific publishing (Hohman et al., 2020). Studies show that interactive articles improve reader engagement and

*Corresponding author.

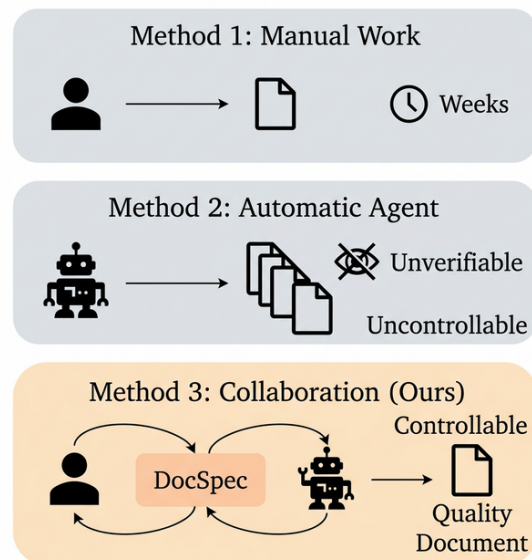


Figure 1: Three approaches to creating interactive documents: manual authoring, fully automatic generation, and human-agent collaboration through DocSpec.

learning compared to static alternatives (Hohman et al., 2020).

Despite their promise, creating interactive articles remains prohibitively costly. Producing a single piece, such as those on Explorable Explanations or Distill.pub (Hohman et al., 2020), requires expertise in both domain knowledge and web development, often demanding days or weeks of effort per article. This bottleneck severely limits the availability of high-quality interactive content across domains.

Recent advances in LLM-based agents have shown promise in automating complex content creation tasks such as video generation (Chen et al., 2025a), slide design (Zheng et al., 2025; Liang et al., 2025), and visualization generation (Tang et al., 2026; Xie et al., 2025; Pan et al., 2025). These systems combine the text and code generation capabilities of large language models (LLMs) with agentic planning and tool use. However, apply-

ing such approaches to interactive educational documents remains largely unexplored, and doing so naively faces three key limitations. First, the generation process is **largely uncontrollable**: there is a fundamental gap between pedagogical intent (what the educator wants the learner to experience) and the executable code that realizes that experience, and the agent resolves this gap based on its own implicit preferences. Second, humans cannot effectively participate in the process: the agent operates as a **black box**, leaving educators no meaningful way to review or adjust the intermediate decisions that shape the final output. Third, datasets for this task are **scarce**, making it difficult to systematically evaluate approaches.

To address these limitations, we present VIVIDOC, a human-agent collaborative system for generating interactive educational documents. VIVIDOC is built on two key ideas. First, we design a multi-agent pipeline where a Planner agent decomposes a topic into a structured plan, an Executor agent synthesizes text and code from the plan, and an Evaluator agent checks the output for correctness. Second, we introduce the Document Specification (DocSpec), a structured intermediate representation that organizes a document into knowledge units, each containing a text description for content generation and an Interaction Specification for visualization generation. The Interaction Specification, developed from interactive visualization theory (Munzner, 2014), decomposes each visualization into four components: State, Render, Transition, and Constraint (SRTC). We extend this theoretical framework to serve as a controllable intermediate layer for LLM-based generation. Because the DocSpec is structured and human-readable, educators can review, edit, and refine it before code is produced, enabling meaningful human control over the generation process. The DocSpec also acts as a contract between agents, reducing ambiguity at the most error-prone step: translating intent into code.

We summarize our contributions as follows:

- We design a multi-agent system that generates interactive educational documents from a single topic input through structured planning, code synthesis, and automated evaluation.
- We propose the Document Specification (DocSpec), a structured intermediate representation consisting of text descriptions and Interaction Specifications (SRTC), that makes the generation process controllable and allows human in-

tervention before code is produced.

- We collect a dataset of 101 real-world interactive documents from over 60 websites across 11 domains. Expert blind evaluation on 10 randomly sampled topics shows that VIVIDOC substantially outperforms naive agentic generation in content richness, interaction quality, and visual quality, and a user study confirms that the DocSpec editing interface is easy to learn and effective for guiding the generation process.

2 Related Work

2.1 Interactive Document

Interactive documents, often conceptualized as explorable explanations (Victor, 2011), are a transformative medium that leverages dynamic web capabilities to foster deep engagement with complex ideas. The theoretical foundation for such active reading environments traces back to early human-computer interaction paradigms, including Engelbart’s framework for augmenting human intellect (Engelbart, 2023) and foundational hypertext structures (Nelson, 1965), which were operationalized in early systems like PLATO (Bitzer et al., 2007). Today, empirical studies confirm that interactive articles significantly improve learner engagement and comprehension compared to static alternatives (Chi and Wylie, 2014; Hohman et al., 2020). The medium has been successfully adopted across diverse fields, from immersive data journalism (e.g., The New York Times’ Snow Fall (Branch, 2012) and Bloomberg’s climate visualizations (Ros-ton and Migliozi, 2015)) to scholarly communications exploring machine learning fairness (Wattenberg et al., 2016).

Despite their efficacy, creating these artifacts remains prohibitively costly, demanding a rare intersection of deep domain knowledge and web development expertise. This bottleneck severely limits the availability of high-quality interactive content. To overcome these authoring barriers, we introduce VIVIDOC, a human-agent collaborative system designed to structurally automate and control the generation of interactive educational documents.

2.2 LLM Agents for Content Creation

Recent advancements increasingly rely on LLMs and multi-agent systems to automate complex content creation by decomposing tasks across specialized agents (Lin et al., 2025; Li et al., 2026; Xie et al., 2026b; Bian et al., 2025). These frame-

works have achieved success across various domains, including data visualization systems like LIDA (Dibia, 2023), Infogen (Ghosh et al., 2025), HAICart(Xie et al., 2024), and PlotGen (Goswami et al., 2025) that synthesize complex infographics from unstructured text (Tang et al., 2026). Similarly, presentation agents like PPTAgent (Zheng et al., 2025), and SlideGen (Liang et al., 2025) use structural schemas for coherent slide decks, while systems like LAVES (Yan et al., 2026) extend multi-agent orchestration to generate synchronized educational videos.

However, applying such agentic approaches to interactive educational documents remains largely unexplored. Naive implementation faces critical limitations regarding uncontrollable generation lacking pedagogical alignment, the black-box nature of agents that preclude meaningful human intervention, and a scarcity of specialized datasets for systematic evaluation. Addressing these fundamental bottlenecks, VIVIDOC establishes a transparent, human-in-the-loop generation paradigm by structuring the collaborative process around an interpretable intermediate representation.

3 System Overview

Figure 2 illustrates the VIVIDOC pipeline. Given a topic as input, VIVIDOC generates a complete interactive educational document through three agents coordinated by a structured intermediate representation, the Document Specification (DocSpec). A key design principle is that the DocSpec is exposed to the user between planning and execution, enabling human review and editing before any code is generated.

3.1 Pipeline

Planner. The Planner agent takes a topic (e.g., “What is the rank of a matrix?”) and produces a DocSpec. It decomposes the topic into a sequence of knowledge units, each containing a text description that guides content generation and an Interaction Specification (SRTC) that defines the interactive visualization. The Planner uses an LLM with structured output to ensure the DocSpec conforms to a predefined schema. We describe the DocSpec structure in detail in Section 4.

Executor. The Executor agent takes the (possibly edited) DocSpec and generates the final HTML document. It processes each knowledge unit in two stages. In Stage 1, it generates the text content as

an HTML fragment, guided by the text description and the context of previously generated sections to maintain stylistic consistency. In Stage 2, it generates the interactive visualization as HTML, CSS, and JavaScript, guided by the SRTC Interaction Specification. The Executor includes a retry mechanism: if a generated fragment fails validation, it regenerates with up to k attempts.

Evaluator. The Evaluator agent checks the generated document for quality. It assesses overall text coherence and logical flow using an LLM, and verifies that all knowledge units have been successfully generated and pass HTML validation. If issues are found, the Evaluator provides feedback that can trigger re-execution of specific components.

Human Review. Human review is not limited to a single stage but spans the entire pipeline. After the Planner produces the DocSpec, the user can reorder knowledge units, modify text descriptions, adjust interaction parameters (e.g., variable ranges, control types), or add and remove units entirely. Because the DocSpec is structured rather than free-form, edits are targeted and predictable: changing a slider range in the Interaction Specification directly affects the generated visualization without requiring the user to write code. After the Executor produces the final document, the user can further review the output and request revisions. In both stages, the user may either edit directly through the interface or converse with an LLM-powered chat assistant to describe desired changes in natural language. This step is optional but provides human control at the points where it is most effective.

The DocSpec serves as a contract between pipeline stages: the Planner expresses pedagogical intent in a structured form, the Executor implements that intent as code, and the Evaluator verifies the result. This decomposition isolates the most error-prone step, the translation from intent to code, and constrains it with a structured specification rather than ambiguous natural language. Placing human review at both the planning and output stages ensures that editorial decisions can be made where they have the highest leverage, before code is generated, and where they are most visible, after the final document is produced.

4 Document Specification

The Document Specification (DocSpec) is the structured intermediate representation at the core of

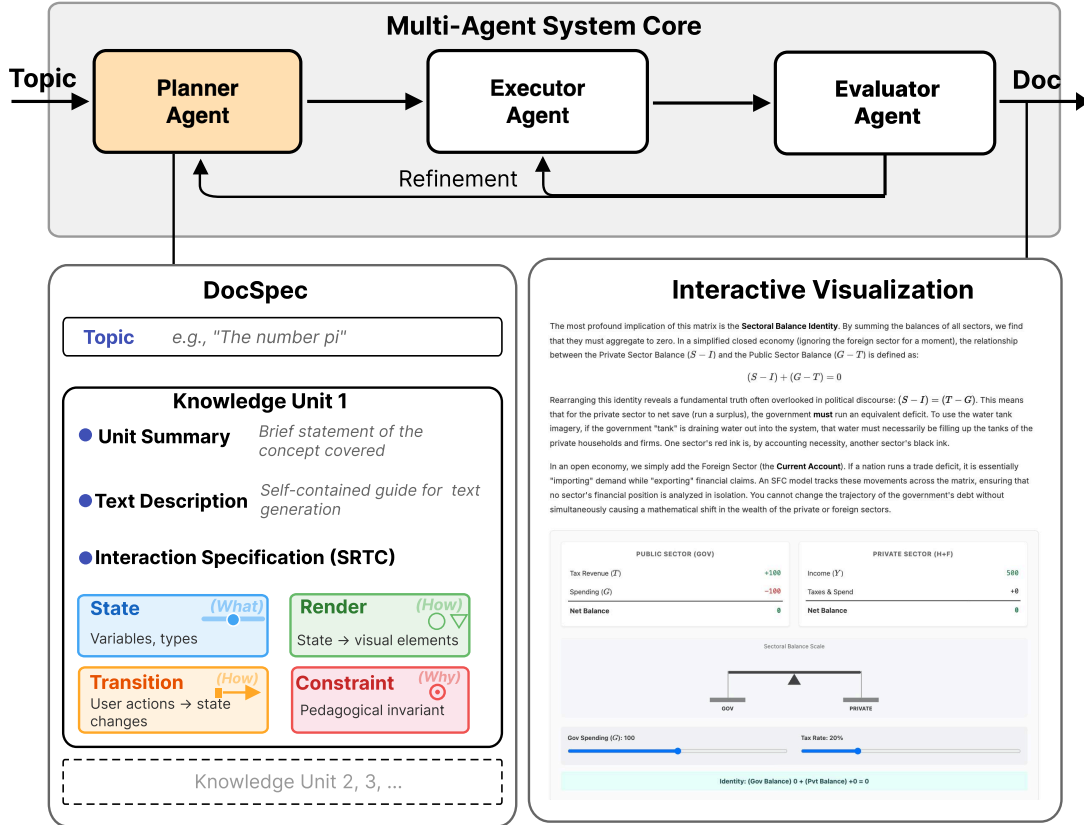


Figure 2: The VIVIDOC pipeline. The Planner generates a Document Specification (DocSpec) from a topic. The user can review and edit the DocSpec before the Executor generates the final document. The Evaluator checks the output and provides feedback.

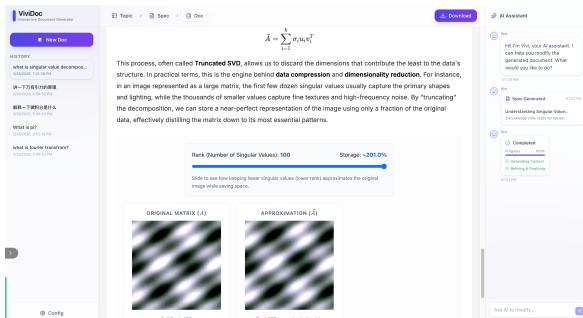


Figure 3: The VIVIDOC user interface. Users follow a Topic → Spec → Doc workflow to generate interactive educational documents.

VIVIDOC. It bridges the gap between a high-level topic and the generated interactive document by decomposing the content into a sequence of knowledge units, each with explicit instructions for both text and interaction generation (Xie et al., 2026a,b; Chen et al., 2025b).

4.1 Structure

A DocSpec consists of a topic and an ordered list of knowledge units. Each knowledge unit contains three components:

- A **unit summary** that briefly states the concept covered.
- A **text description** that provides a self-contained guide for generating the explanatory text of the section. It specifies what the reader should understand after reading, without prescribing exact wording.
- An **Interaction Specification** that defines the interactive visualization using the SRTC decomposition described below.

The text description and Interaction Specification serve different stages of the Executor: the former guides Stage 1 (text generation) and the latter guides Stage 2 (visualization generation). This separation allows each stage to operate independently with minimal ambiguity. The Interaction Specification supports common interaction patterns

such as parameter exploration (sliders), selection (dropdowns), direct manipulation (drag), and mode switching (toggles), while static visualizations are handled as a degenerate case where no transitions are defined.

4.2 Interaction Specification: SRTC

Munzner’s What-Why-How framework (Munzner, 2014) provides a foundational decomposition for visualization design: What data is being visualized, How it is visually encoded and interacted with, and Why the user engages with it. We adapt this framework to the setting of LLM-based generation by introducing the SRTC Interaction Specification, which decomposes each interactive visualization into four components:

- **State (S):** The variables underlying the visualization, including their types, domains, default values, and derivation rules. Variables are either user-controllable (e.g., a slider with a specified range) or derived from other variables (e.g., a formula). This corresponds to Munzner’s What.
- **Render (R):** A description of how the state maps to visual elements on screen, such as geometric shapes, labels, or charts. This corresponds to the visual encoding aspect of How.
- **Transition (T):** A description of how user actions modify the state, specifying the cause-and-effect relationship between input events and state changes. This corresponds to the interaction idiom aspect of How.
- **Constraint (C):** The pedagogical invariant that the visualization is designed to demonstrate. This is the key insight the learner should discover through interaction. This corresponds to Munzner’s Why, adapted from “what task is the user performing” to “what should the learner observe.”

This structured format benefits all three agents: it constrains the Planner to produce well-formed specifications, provides the Executor with an unambiguous contract for code synthesis, and gives the Evaluator an objective criterion (the Constraint C) for verifying correctness.

Example. Table 1 contrasts a natural language interaction description with its corresponding SRTC specification for a visualization about π . The natural language version leaves implicit what variables exist, how they relate, what appears on screen, and what invariant the learner should notice. The SRTC

Natural language description

“The reader can adjust the radius of a circle using a slider, and as the radius changes, the visualization updates the circle while recalculating and displaying the circumference, diameter, and ratio, allowing the reader to see that the ratio stays roughly the same.”

SRTC Interaction Specification

S	r : slider $[0.5, 5]$, default 1; C: derived $2\pi r$; D: derived $2r$; ratio: derived C/D
R	A circle whose size reflects r ; labels showing C , D , and ratio
T	Dragging the slider changes r ; C , D , ratio update automatically
C	ratio ≈ 3.14159 regardless of r

Table 1: Comparison of a natural language interaction description and its SRTC Interaction Specification for a visualization about π .

version makes each of these explicit. The State defines a user-controllable variable r (a slider over $[0.5, 5]$) and three derived variables: circumference $C = 2\pi r$, diameter $D = 2r$, and their ratio C/D . The Render specifies that the visualization displays a circle whose size reflects r , along with labels for C , D , and the ratio. The Transition links the slider interaction to the state: dragging the slider changes r , and all derived variables update automatically. Finally, the Constraint encodes the pedagogical invariant that the learner should discover: $C/D \approx 3.14159$ regardless of r .

5 Experiments

5.1 Output Quality Evaluation

We randomly sampled 10 topics from our benchmark dataset and generated interactive educational documents using two methods, producing 20 documents in total. VIVIDOC follows the multi-agent pipeline: the Planner generates a DocSpec, the Executor generates the document, and the Evaluator checks the output. To ensure a fair comparison, we disable the Human Review step so that no human feedback is involved in either method. Naive Agent uses the same topic as input and prompts the LLM to generate a complete HTML document in a single call, without DocSpec planning. Both methods use the same underlying model (Gemini 3.0 Flash) to control for differences in model capability.

We recruited three domain experts with backgrounds in interactive educational content design or visualization research to conduct a blind evaluation of the 20 documents. Experts were not informed of which method produced each document. They

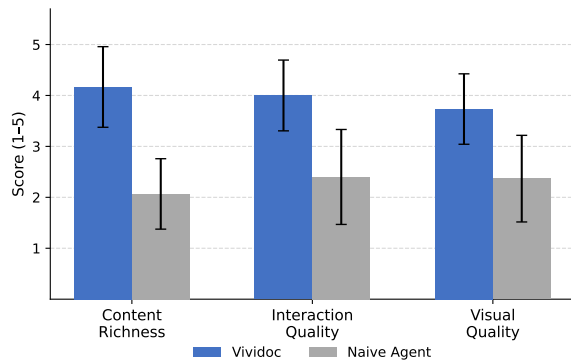


Figure 4: Expert blind evaluation scores for VIVIDOC and Naive Agent across three dimensions.

rated each document on three dimensions using a 5-point Likert scale: *Content Richness* (whether the document covers the topic with sufficient depth and breadth), *Interaction Quality* (whether the interactive design helps readers explore and understand the core concept), and *Visual Quality* (the layout and readability of the document).

As shown in Figure 4, VIVIDOC outperforms Naive Agent on all three dimensions. The gap is largest for Content Richness (4.17 vs. 2.07), followed by Interaction Quality (4.00 vs. 2.40) and Visual Quality (3.73 vs. 2.37). These results indicate that without structured planning through DocSpec, a single-call LLM struggles to produce documents with well-organized content, coherent interactive design, and clear visual layout. The most pronounced drop in Content Richness suggests that without structured planning, LLM-generated documents tend to be shallow and narrowly focused, while the drop in Interaction Quality confirms that the SRTC Interaction Specification plays a central role in constraining the generation of meaningful interactions.

5.2 User Study

We conducted a user study to evaluate the usability of VIVIDOC. The study focused on whether participants could effectively use the DocSpec editing interface to guide the generation process toward their intended output.

Participants. We recruited three participants with backgrounds in visualization or educational technology research. All participants had prior experience in interactive content creation or educational tool design.

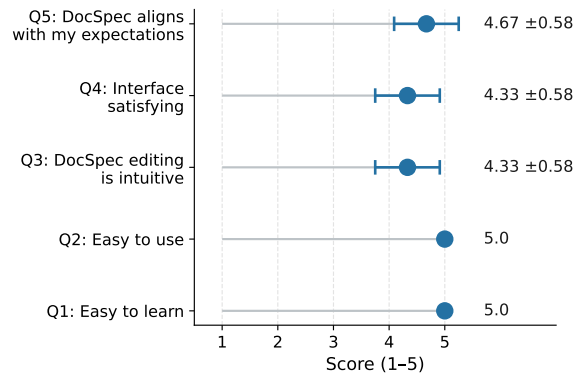


Figure 5: Participant ratings of VIVIDOC on five usability dimensions. Values show mean and standard deviation.

Procedure. Each participant completed three document generation tasks covering topics in mathematics, algorithms, and physics, presented in a random order to reduce ordering effects. A 10-minute introduction and practice session was provided before the tasks began. Tasks had no time limit. Participants were encouraged to think aloud throughout the session, and a 20-minute semi-structured interview was conducted afterward to collect qualitative feedback.

Usability Ratings. After completing the tasks, participants rated the system on five items using a 5-point Likert scale. As shown in Figure 5, all items received positive ratings: easy to learn (5.0), easy to use (5.0), DocSpec editing is intuitive (4.33 ± 0.58), interface satisfying (4.33 ± 0.58), and DocSpec aligns with my expectations (4.67 ± 0.58), indicating that the DocSpec interface effectively supports the document generation process.

In the interviews, participants noted that DocSpec was the first time they could “actually decide how the interaction works without writing any code” (P1), and that the separation between the text description and the interaction specification made editing more straightforward (P2).

6 Conclusion

We presented VIVIDOC, a human-agent collaborative system for generating interactive educational documents. By introducing the Document Specification (DocSpec) as a structured intermediate representation, VIVIDOC enables meaningful human control over the generation process while leveraging multi-agent automation. Experiments show that structured planning through DocSpec leads to

substantial quality improvements over naive generation, and a user study confirms that the editing interface is intuitive and effective. We hope VIVIDOC lowers the barrier to creating interactive content and inspires further research on human-AI collaboration in document authoring.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62421003) and Zhejiang Provincial Natural Science Foundation of China under Grant No. LD24F020011.

References

- Yutong Bian, Xianhao Lin, Yupeng Xie, Tianyang Liu, Mingchen Zhuge, Siyuan Lu, Haoming Tang, Jinlin Wang, Jiayi Zhang, Jiaqi Chen, and 1 others. 2025. You don't know until you click: Automated gui testing for production-ready software evaluation. *arXiv preprint arXiv:2508.14104*.
- Donald Bitzer, Peter Braunfeld, and Wayne Lichtenberger. 2007. Plato: An automatic teaching device. *IRE Transactions on Education*, 4(4):157–161.
- John Branch. 2012. Snow fall: The avalanche at tunnel creek. The New York Times. <https://www.nytimes.com/projects/2012/snow-fall/index.html>.
- Yanzhe Chen, Kevin Qinghong Lin, and Mike Zheng Shou. 2025a. Code2video: A code-centric paradigm for educational video generation. *arXiv preprint arXiv:2510.01174*.
- Yiyu Chen, Yifan Wu, Shuyu Shen, Yupeng Xie, Leixian Shen, Hui Xiong, and Yuyu Luo. 2025b. Chartmark: A structured grammar for chart annotation. In *2025 IEEE Visualization and Visual Analytics (VIS)*, pages 311–315. IEEE.
- Micheline T. H. Chi and Ruth Wylie. 2014. The ICAP framework: Linking cognitive engagement to active learning outcomes. *Educational Psychologist*, 49(4):219–243.
- Victor Dibia. 2023. 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)*, pages 113–126.
- Douglas C Engelbart. 2023. Augmenting human intellect: A conceptual framework. In *Augmented education in the global age*, pages 13–29. Routledge.
- Akash Ghosh, Aparna Garimella, Pritika Ramu, Sambaran Bandyopadhyay, and Sriparna Saha. 2025. Infogen: Generating complex statistical infographics from documents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20552–20570.
- Kanika Goswami, Puneet Mathur, Ryan Rossi, and Franck Dernoncourt. 2025. Plotgen: multi-agent llm-based scientific data visualization via multimodal retrieval feedback. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1672–1676.
- Fred Hohman, Matthew Conlen, Jeffrey Heer, and Duen Horng Chau. 2020. Communicating with interactive articles. *Distill*.
- Boyan Li, Yiran Peng, Yupeng Xie, Sirong Lu, Yizhang Zhu, Xing Mu, Xinyu Liu, and Yuyu Luo. 2026. Deepeye: A steerable self-driving data agent system. *arXiv preprint arXiv:2603.28889*.
- Xin Liang, Xiang Zhang, Yiwei Xu, Siqi Sun, and Chenyu You. 2025. Slidegen: Collaborative multimodal agents for scientific slide generation. *arXiv preprint arXiv:2512.04529*.
- Yi-Cheng Lin, Kang-Chieh Chen, Zhe-Yan Li, Tzu-Heng Wu, Tzu-Hsuan Wu, Kuan-Yu Chen, Hung-yi Lee, and Yun-Nung Chen. 2025. Creativity in llm-based multi-agent systems: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27572–27595.
- Tamara Munzner. 2014. *Visualization Analysis and Design*. A K Peters Visualization Series. A K Peters/CRC Press.
- Theodor Holm Nelson. 1965. Complex information processing: a file structure for the complex, the changing and the indeterminate. In *Proceedings of the 1965 20th national conference*, pages 84–100.
- Bo Pan, Yixiao Fu, Ke Wang, Junyu Lu, Lunke Pan, Ziyang Qian, Yuhan Chen, Guoliang Wang, Yitao Zhou, Li Zheng, and 1 others. 2025. Vis-shepherd: Constructing critic for llm-based data visualization generation. *arXiv preprint arXiv:2506.13326*.
- Eric Roston and Blacki Migliozi. 2015. What's really warming the world. Bloomberg Business. <https://www.bloomberg.com/graphics/2015-whats-warming-the-world/>.
- Yinghao Tang, Xueding Liu, Boyuan Zhang, Tingfeng Lan, Yupeng Xie, Jiale Lao, Yiyao Wang, Haoxuan Li, Tingting Gao, Bo Pan, and 1 others. 2026. Igenbench: Benchmarking the reliability of text-to-infographic generation. *arXiv preprint arXiv:2601.04498*.
- Bret Victor. 2011. Explorable explanations. <http://worrydream.com/ExplorableExplanations/>. Accessed: 2026-02-27.
- Martin Wattenberg, Fernanda Viégas, and Moritz Hardt. 2016. Attacking discrimination with smarter machine learning. Google Research.

<https://research.google.com/bigpicture/attacking-discrimination-in-ml/>.

Yupeng Xie, Yuyu Luo, Guoliang Li, and Nan Tang. 2024. Haichart: Human and ai paired visualization system. *arXiv preprint arXiv:2406.11033*.

Yupeng Xie, Chen Ma, Zhenyang Wang, Liangwei Wang, Jiayi Zhu, Chuxuan Zeng, Zhouan Shen, Boyan Li, and Yuyu Luo. 2026a. Datamagic: Transforming tabular data into data insight video. *arXiv preprint*.

Yupeng Xie, Zhenyang Wang, Liangwei Wang, Jiayi Zhu, Zhouan Shen, and Yuyu Luo. 2026b. Datamagic: Authoring data videos through declarative multi-agent orchestration. *arXiv preprint*.

Yupeng Xie, Zhiyang Zhang, Yifan Wu, Sirong Lu, Jiayi Zhang, Zhaoyang Yu, Jinlin Wang, Sirui Hong, Bang Liu, Chenglin Wu, and 1 others. 2025. Visjudge-bench: Aesthetics and quality assessment of visualizations. *arXiv preprint arXiv:2510.22373*.

Lingyong Yan, Jiulong Wu, Dong Xie, Weixian Shi, Deguo Xia, and Jizhou Huang. 2026. Beyond end-to-end video models: An llm-based multi-agent system for educational video generation. *arXiv preprint arXiv:2602.11790*.

Hao Zheng, Xinyan Guan, Hao Kong, Wenkai Zhang, Jia Zheng, Weixiang Zhou, Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2025. Pptagent: Generating and evaluating presentations beyond text-to-slides. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14413–14429.

7 Ethical Considerations

The dataset of 101 interactive documents was collected and utilized in strict compliance with applicable copyright regulations. For our user study, we obtained explicit informed consent from all participants and rigorously anonymized all interview records to protect personal privacy. Although VIVI-DOC employs LLMs to assist in generation, we mitigate potential risks of AI hallucinations and uncontrollable outputs through a human-in-the-loop paradigm. By allowing users to review and edit the Document Specification (DocSpec) prior to code synthesis, the system ensures that educators retain full control over the pedagogical intent. As this work focuses on collaborative human-agent authoring rather than deploying a fully autonomous system, we do not foresee any significant ethical or societal risks.