

VisCopilot: Assisting Data-Driven Article Authoring with Fact-Based and Intent-Aligned Autocompletion

Tzu-Wei Yu¹, Che-Wei Chang¹, Evelyn Yi-Ting Hung², Jia-Yu Pan³, Wen-Chieh Lin¹

¹National Yang Ming Chiao Tung University, Taiwan

²Northwestern University, USA ³Google, USA

twyu.cs11@nycu.edu.tw, composer0824@gmail.com,
evelynhung2025@u.northwestern.edu, jiayu.pan@gmail.com
wclin@cs.nycu.edu.tw

Abstract

Data-driven articles combine narrative text with visualizations to communicate insights but are time-consuming to produce, requiring both narrative composition and accurate text-chart linking. Existing automated linking methods struggle with accuracy, and automated narrative generation often misaligns with user intent and may contain hallucinations. We present VisCopilot, which leverages retrieval-augmented generation and large language models to provide fact-based, intent-aligned suggestions. As users write, the system supports narrative completion and text-chart linking in a single step, with a human-in-the-loop approach ensuring reliability. Technical evaluation on a benchmark dataset of data-driven articles authored by journalists confirmed the accuracy and quality of VisCopilot's links. A user study further showed that VisCopilot outperforms baselines in usability, accuracy, and user confidence, with participants valuing its fact-based autocompletes for aligning suggestions with their writing intent. VisCopilot contributes: (1) an autocomplete-based design for assisting data-driven article authoring that balances efficiency, accuracy, and user autonomy; and (2) a reframing of text-chart link generation as a data-fact recommendation problem, enabling intent-aligned retrieval and systematic evaluation.

1 Introduction

Data-driven articles that combine textual narratives with data visualizations have become a widely used medium for communicating complex information in domains such as journalism (Hao et al., 2024), business reporting (Zhang et al., 2022), and education (Dragicevic et al., 2019; Victor, 2011). As online storytelling has become increasingly popular, the online format of these articles often feature text-chart links that directly connect written content with corresponding visualizations, improving comprehension and engagement (Sultanum et al., 2021;

Conlen et al., 2021; Latif et al., 2021; Sultanum and Srinivasan, 2023). By linking key narrative points to relevant data marks, readers avoid manually searching charts for evidence, making the experience more interactive and accessible.

Despite their value, authoring data-driven articles remains a demanding process. Writers must analyze data, extract insights, structure them into a narrative, and create text-chart links (Sultanum and Srinivasan, 2023; Cao et al., 2023). This is both time-consuming and cognitively taxing. Existing tools only partially address the problem. Manual authoring systems give users full control over linking but require significant effort (Conlen and Heer, 2018; Conlen et al., 2021; Sultanum et al., 2021). Automatic linking (linking-only) systems can misidentify relationships between text and visualizations and still require users to write the text manually (Latif et al., 2021; Lai et al., 2020). Fully automated narrative generation also faces significant challenges, producing content misaligned with user intent or introducing factual errors (Choi and Jo, 2022; Sultanum and Srinivasan, 2023).

These limitations motivate a new approach. We draw inspiration from autocomplete systems such as Gmail Smart Compose (Chen et al., 2019; Zhu et al., 2023) and GitHub Copilot (GitHub, Inc.), which demonstrate the effectiveness of context-aware suggestions for writing and coding, as well as findings that users prefer multiple suggestions in writing assistance systems (Dang et al., 2023). Building on these insights, we propose VisCopilot, which explores a middle ground between manual authoring, linking-only assistance, and fully automated generation. It provides fact-based, intent-aligned autocomplete suggestions that simultaneously establish links, while keeping users in control through explicit acceptance. Due to autocomplete design, text-chart links in this work take the form of *sentence-chart links*, where each sentence conveys a data insight connected to relevant chart elements.

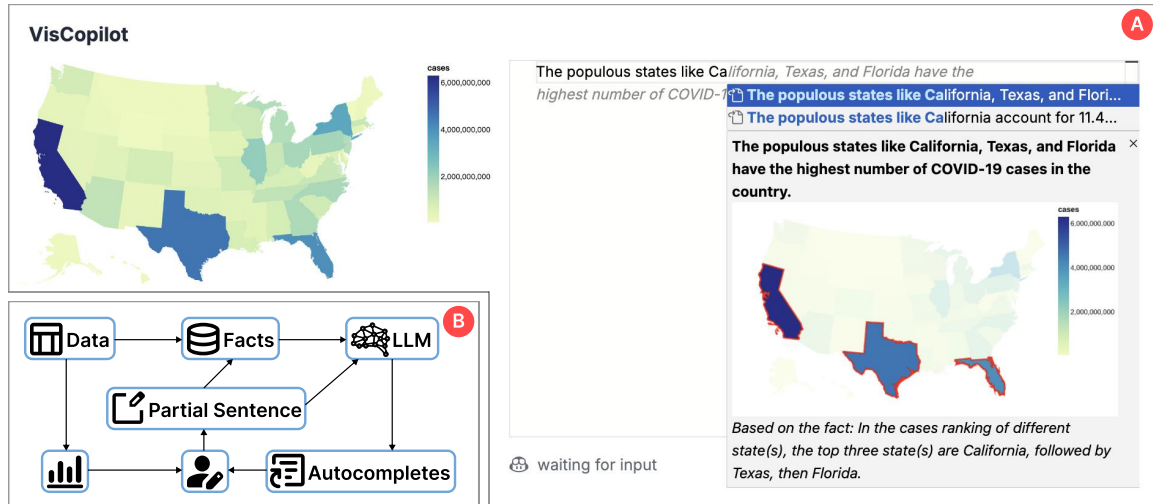


Figure 1: The interface of VisCopilot (A) and the system architecture of VisCopilot (B).

Applying autocomplete to data-driven article authoring, however, introduces unique challenges. Suggestions must be (1) factually accurate and free from hallucinations, and (2) aligned with user intent, since the same dataset may support multiple valid narratives. In our setting, **user intent is represented as a specific data fact** (Section 4.1), which is a finding the user considers noteworthy and wishes to highlight in a chart. To address these challenges, we frame text-chart link autocomplete as a **data-fact recommendation problem**. Given a chart and its data, VisCopilot automatically extracts and ranks data facts, each comprising a narrative unit and corresponding visual highlights. As users begin writing, the system retrieves data facts aligned with their partially expressed intent via retrieval-augmented generation (RAG) and large language models (LLMs), producing suggestions paired with an annotated chart that allows authors to simultaneously draft text and establish accurate sentence-chart links in a single process (Fig. 1B).

We evaluated VisCopilot through a technical evaluation measuring intent-matching accuracy (Hits@K) on a 32-sentence benchmark (Kong et al., 2014; Latif et al., 2021), and a within-subjects user study against a baseline system. VisCopilot showed significantly higher ratings in usability, accuracy, and user confidence, with participants particularly valuing fact-based autocompletes for their alignment with intent.

This work makes the following contributions: (1) An autocomplete-based design for data-driven article authoring assistance that balances efficiency, accuracy, and user autonomy. (2) A reframing of text-chart link generation as a data-fact recommen-

dation problem, enabling intent-aligned retrieval and systematic evaluation. (3) VisCopilot, a RAG- and LLM-based system validated through a technical evaluation and a within-subjects user study, showing improvements in efficiency, accuracy, and user satisfaction.

2 Related Work

2.1 Data-Driven Article Authoring

Data-driven articles combine textual narratives with visualizations through *text-chart links*, which explicitly connect text descriptions to related chart elements (Sultanum et al., 2021; Sultanum and Srinivasan, 2023). These links are often enhanced with interactions such as visual highlights or scroll-driven updates to reduce the cognitive load of navigating between text and charts (Zhi et al., 2019), and have been shown to improve reader comprehension and engagement (Kim et al., 2021).

Given the complexity of authoring data-driven articles and the demonstrated importance of text-chart links for reader comprehension, researchers have proposed various approaches to streamline the process. These approaches can be categorized based on the level of automation and user involvement in narrative authoring and text-chart link creation, as follows:

Manual authoring systems reduce the technical barrier of creating data-driven articles. Idyll (Conlen and Heer, 2018; Conlen et al., 2021) introduced a markup language for authoring without programming, and VizFlow (Sultanum et al., 2021) provided intuitive tools for manually creating text-chart links.

Automatic linking systems generate text-chart

links while leaving narrative writing entirely to the user. Vis-Annotator (Lai et al., 2020) uses NLP to automatically parse charts and establish links, while Kori (Latif et al., 2021) takes a mixed-initiative approach, suggesting data linkages for specific words in fully user-authored sentences. CrossData (Chen and Xia, 2022) focuses on data binding by retrieving relevant elements (rows, columns, values) or performing predefined operations (e.g., sorting, filtering, identifying extremes). Other systems such as EmphasisChecker (Kim et al., 2023) and Data Player (Shen et al., 2024) similarly operate at the phrase or value level. Across these systems, users must still manually compose complete sentences to express their insights, as the systems assist with linking but provide no support for narrative generation itself. In contrast, VisCopilot addresses this gap by leveraging LLMs and RAG to identify user intent from a partial sentence prefix and predict the full intended sentence as an autocomplete suggestion, simultaneously generating the narrative and establishing the corresponding sentence–chart link.

Fully automated systems aim to automate both narrative generation and the creation of text–chart links. Intentable (Choi and Jo, 2022) allows users to select chart elements and a topic as input, guiding neural generation toward user intent. DATATALES (Sultanum and Srinivasan, 2023) generates prompts from user-selected chart marks, passing them to an LLM to produce narratives with automatically created text–chart links. Although efficient, these systems can produce verbose content with factual errors or hallucinations (Choi and Jo, 2022; Sultanum and Srinivasan, 2023), and users seek greater control over the generated output (Sultanum and Srinivasan, 2023).

2.2 Data Narratives and AI-assisted Writing

Early approaches to data narrative generation relied on template-based verbalization (Wang et al., 2020; Srinivasan et al., 2018; Shi et al., 2020; Li et al., 2023), which ensured factual consistency but often produced limited linguistic diversity. More recent methods use language models to generate narratives from charts or tables (Kantharaj et al., 2022; Tang et al., 2023; Liu et al., 2020), offering greater flexibility but frequently suffering from hallucinations and factual inaccuracies (Kantharaj et al., 2022; Sultanum and Srinivasan, 2023; Tang et al., 2023; Li et al., 2024). Recent LLM-based systems introduce interactive workflows for data

storytelling. For example, DataNarrative (Islam et al., 2024) employs a multi-agent pipeline for end-to-end story generation, while DATAWEAVER (Fu et al., 2025) supports bidirectional composition through user-initiated interactions on chart elements. However, both systems require users to explicitly specify their intent (e.g., selecting chart elements or goals) before generation begins. In contrast, VisCopilot continuously infers intent on the fly from the user’s unfolding text, offering real-time completions without requiring prior manual specification.

Outside data narrative authoring, LLM-based writing assistance for general-purpose writing has also become an active research area in HCI and NLP (Lee et al., 2024). Prior work has shown that user directives or partial input can guide generation toward authorial goals (Sun et al., 2021), while systems such as Wordcraft (Yuan et al., 2022) and Sparks (Gero et al., 2022) support ideation and writing through interactive suggestions. Unlike these open-ended writing assistants, VisCopilot addresses the stricter constraints of data-driven authoring: suggestions must dynamically adapt to evolving intent while remaining strictly grounded in verifiable data facts and generating coordinated visual links in real time.

3 Design Requirements

Prior work on data-driven article authoring and text–chart link generation provides valuable guidance and informs our design requirements (Latif et al., 2021; Sultanum et al., 2021; Conlen et al., 2021; Sultanum and Srinivasan, 2023).

DR1: Accurate and Efficient Text–Chart Link Creation. Manual text–chart link creation is time-consuming, while automatic linking systems (Latif et al., 2021; Lai et al., 2020) require complete text beforehand and may produce inaccurate or missed links. A data-driven authoring tool should therefore enable seamless, accurate text–chart link creation with minimal user effort (Ajani et al., 2021; Sultanum et al., 2021; Latif et al., 2021).

DR2: Ensuring Trustworthiness and Mitigating Lengthy Responses. Fully automated systems often generate verbose or factually inaccurate content (Sultanum and Srinivasan, 2023; Kantharaj et al., 2022; Li et al., 2024). The tool should produce concise, trustworthy content grounded in data, minimizing hallucinations and reducing edits.

DR3: Aligning Content with User Intent. Auto-

mated systems often fail to align with user intent due to the multifaceted nature of data (Choi and Jo, 2022; Sultanum and Srinivasan, 2023). The tool should interpret partial user input accurately and generate content that closely reflects the user’s intended narrative.

4 Proposed System

VisCopilot combines a front-end interface and a back-end pipeline (Fig. 2). Users first upload a dataset and configure a chart (type and axes) via the sidebar (Fig. 2A), which is rendered using Vega-Lite (Satyanarayan et al., 2016). They then write in the text editor (Fig. 2B). When a brief typing pause is detected, VisCopilot shows up to three autocomplete suggestions in a dropdown menu (Fig. 2C). Each suggestion includes a completed sentence, highlighted chart, and the underlying data fact (Fig. 2D). Pressing *Enter* accepts a suggestion, inserting the sentence and binding the highlighted chart elements as a sentence-chart link. Users can also toggle to presentation mode (Fig. 2F) to view the interactive article (Fig. 3). Please also refer to the [supplementary video for a demonstration](#).

On the back end, once a chart is configured, the system generates all relevant data facts with corresponding highlighted charts (Fig. 2G–H) and precomputes their sentence embeddings. As the user types, the typed prefix is used to retrieve and rerank the top data facts (Fig. 2I), which are then passed with the prefix to an LLM to produce the final autocomplete suggestions.

4.1 Data Fact and Highlighted Chart Generation

Data Fact Generation: Given a chart and dataset, VisCopilot generates data facts using algorithms from Calliope (Shi et al., 2020) and DataShot (Wang et al., 2020). Following Calliope’s formulation, a data fact is a 5-tuple $\{type, subspace, breakdown, measure, focus\}$; since *breakdown* and *measure* are fixed by the chart specification, we simplify this to a 3-tuple $f_i = \{type, subspace, focus\}$ (e.g., Fig. 2H: *type*=Trend, *subspace*=“Genre=Drama”, *focus*=2006-2010). VisCopilot supports 9 data fact types (Appendix A) across bar, line, pie, choropleth, and scatter charts, selecting applicable types based on axis data types. Facts are articulated in natural language via template-based methods (Shi et al., 2020; Lin et al., 2023), minimizing factual

errors.

Highlighted Chart Generation: Each data fact is paired with a highlighted chart (Fig. 2H) that visually emphasizes the fact’s *focus*. The highlights are generated from Vega-Lite specifications (Satyanarayan et al., 2016) using techniques adapted from prior work on chart highlighting and annotation (Li et al., 2023; Ajani et al., 2021; Latif et al., 2021; Lai et al., 2020), such as color outlines, opacity dimming, and trend arrows. By leveraging these visual encodings, we bring a novel perspective to text-chart linking: rather than inferring visual links post-hoc from written text, we inherently couple each generated data fact with its visual highlight from the outset. This joint formulation serves as the backbone for providing synchronous narrative and visual autocomplete suggestions in the next stage (Section 4.2).

4.2 Autocomplete Suggestions

VisCopilot provides real-time autocomplete suggestions to assist users in writing data-driven articles while creating sentence-chart links in one step.

RAG Framework for Autocompletes: We adopt the RAG framework (Lewis et al., 2020; Gao et al., 2023; Jiang et al., 2023) to ground LLM outputs in retrieved documents to reduce hallucinations (Niu et al., 2024). The generated data facts serve as the knowledge base and the user’s incomplete sentence (*prefix*) acts as the query, retrieving the most relevant data facts and uses them with the prefix to generate intent-aligned completions with fact-based sentence-chart links.

Embedding Calculation: To enable efficient retrieval of relevant data facts, we calculate the text embeddings of the generated data fact sentences using OpenAI’s `text-embedding-3-small` model (OpenAI, Inc.). These embeddings are stored in FAISS (Douze et al., 2026), a highly optimized vector database for similarity search that allows real-time retrieval of relevant data facts based on user input.

Real-Time Autocompletes: As users write, VisCopilot monitors changes in their input, continuously waiting for a brief pause in typing. We empirically set a 1.2-second pause threshold through pilot testing to prevent premature triggers while typing. When detecting a pause, VisCopilot uses OpenAI’s embedding model to compute the embedding of the user’s *prefix* and performs a cosine similarity search in FAISS to retrieve the top-10 most relevant data facts.



Figure 2: System overview of VisCopilot. Top: Front end interface; Bottom: Back end pipeline.

Although retrieving the top-10 similar data facts offers a solid foundation, the relevance and precision of these results can vary (Barnett et al., 2024). To refine these suggestions, we introduce a reranking step that scores and reorders candidates against the user’s *prefix*, serving as both a filter and an enhancer (Zhuang et al., 2023; Hou et al., 2024). Following recommendations from the survey (Gao et al., 2023), we adopt Cohere’s reranking model (Cohere, Inc.), `rerank-english-v3.0` with `top_n=3`, to narrow the candidate list down to the top-3 most relevant data facts.

Prompt Generation and LLM Completion: Once the top-3 data facts are selected, VisCopilot generates a prompt for each fact using a predefined template (Listing 1 in Appendix B). This prompt combines the user’s *prefix* with the retrieved data fact, the information of the dataset, and the **prior sentences in the article as context** for the LLM (gpt-4o-mini with `temperature=0.1`). The prompts are sent to the LLM, which responds with a complete sentence that integrates the data fact into the user’s narrative. The LLM is instructed to return `NO_ANSWER` if it cannot confidently generate a valid completion, allowing the system to filter out unhelpful responses.

Delivering the Autocomplete Suggestions: After filtering out any `NO_ANSWER` responses, the system displays up to three autocomplete suggestions in a dropdown menu (Fig. 2C). Each suggestion includes a completed sentence, a highlighted chart emphasizing the *focus* data points, and the underlying data fact (Fig. 2D), simultaneously linking the text to the corresponding visual elements and ensuring the narrative remains grounded in data.

4.3 Sentence-Chart Linking and Presentation

VisCopilot not only assists in writing narratives and linking them to relevant visualizations but also offers interactive ways to present the completed articles. In presentation mode, the sentence-chart links established during the writing process become interactive elements of the article (Fig. 3). When a reader hovers over a linked sentence, the chart aside dynamically updates to show the highlighted chart corresponding to the sentence-chart link. Previous studies have demonstrated that interactive text-visual linking aids narrative comprehension (Lai et al., 2020; Sultanum et al., 2021); our presentation mode illustrates how links created during authoring can directly support such interactive reading experiences.



Figure 3: VisCopilot’s interactive presentation mode: hovering over a linked sentence dynamically updates the chart and highlights relevant data points (red arrow, red boxes).

5 Technical Evaluation

We conducted a technical evaluation to assess VisCopilot’s intent-matching accuracy.

5.1 Evaluation Method and Measures

Dataset: We adopted a dataset of 40 paragraph-chart pairs from Kong et al. (2014), which was previously used in Kori (Latif et al., 2021) to evaluate text-chart links. Each pair contains a paragraph with sentences referring to a corresponding chart. Since VisCopilot operates at the sentence level, we refined the dataset by removing sentences containing unsupported data facts (e.g., comparisons involving more than two data points), resulting in 32 sentence-chart pairs. We then manually labeled each sentence-chart pair to establish the ground-truth data fact, identifying the data fact ($f_i = \{type, subspace, focus\}$) described in each sentence-chart pair.

Method: We evaluated the intent-matching accuracy of sentence-chart link generation by providing different prefix lengths of each sentence in the dataset, including one-third, two-thirds, and the full sentence. Each prefix was used as input to retrieve and rerank candidate data facts using sentence embeddings, and the retrieved results were compared against the ground-truth data fact associated with the sentence-chart pair.

Measure: We measured intent-matching accuracy using the **Hits@K** metric (Cremonesi et al., 2010), a standard measure for recommendation systems. In our setting, Hits@K evaluates whether the ground-truth data fact appears within the top- K retrieved candidates: $Hits@K = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\text{rank}(f_i) \leq K)$, where N is the total number of sentence-chart pairs and f_i denotes the ground-truth data fact for the i -th test case.

We report four metrics to evaluate both retrieval and reranking: Retrieved Hits@10 and Retrieved Hits@3 measure whether the ground-truth data fact appears in the top-10 and top-3 retrieved candidates, respectively, while Reranked Hits@3 and Reranked Hits@1 evaluate the results after rerank-

Prefix Length	Retrieved Hits@10 (%)	Retrieved Hits@3 (%)	Reranked Hits@3 (%)	Reranked Hits@1 (%)
Whole Sentence	81.3	56.3	71.9	46.9
2/3 Sentence	59.4	37.5	37.5	25.0
1/3 Sentence	34.4	15.6	25.0	21.9

Table 1: VisCopilot’s retrieval and reranking accuracy (Hits@K) across different prefix lengths.

ing. Comparing these metrics quantifies the contribution of each stage on intent-matching accuracy.

5.2 Results

Table 1 shows VisCopilot’s intent-matching accuracy across prefix lengths. For whole sentences, Retrieved Hits@10 reached 81.3%, with performance declining as prefixes shorten (59.4% and 34.4% for two-thirds and one-third, respectively). The reranking stage improved results in most settings, including both the one-third-prefix and whole-sentence conditions: for the one-third prefix, Reranked Hits@3 rose to 25.0% from a Retrieved Hits@3 of 15.6%, while for whole sentences, it rose to 71.9% from 56.3%.

VisCopilot achieved Reranked Hits@1 of 46.9% for whole sentences, compared to Kori’s (Latif et al., 2021) 41.6%. Although the two systems differ in linking granularity (sentences vs. phrases), this confirms that our baseline (Section 6.1) is robust and comparable to existing systems.

6 User Study

We conducted a within-subjects user study to evaluate the effectiveness of VisCopilot’s autocomplete functionality and overall system performance by comparing it with a baseline system.

6.1 Study Design

Baseline: The baseline system used the same back-end algorithms and RAG-based data fact pipeline as VisCopilot, but differed in interaction timing: suggestions were triggered only after the user completed a full sentence, offering a single post-hoc link recommendation. We chose this controlled baseline over existing systems to isolate the effect of interaction design, avoiding confounds from algorithmic or task differences. The baseline provides a strong reference condition with fully specified user intent, allowing us to evaluate whether VisCopilot can infer intent from partial input while maintaining accuracy and improving usability.

Datasets used in the study were sourced from the Pew Research Center, widely used in previous studies (Kong et al., 2014; Sultanum et al., 2021;

Latif et al., 2021; Chen and Xia, 2022). The first one, *Social Media* dataset (Pew Research Center, 2023) featured a grouped bar chart illustrating the usage of five different social media apps, and the second one, *News Platform* dataset (Pew Research Center, 2011), was a multi-series line chart depicting the sources where people get their news over a decade. The different chart types were used to avoid the participants’ learning effect. Additionally, the movies dataset from the Vega dataset (2024) was used for the tutorial and exercise.

Participants: We recruited 12 participants (6 male, 6 female, ages 20–27) with regular experience using data processing and document software. All were comfortable using English in professional contexts. Participants were publicly recruited and received approximately NT\$400 as a token of appreciation. The study was approved by our university’s Institutional Review Board (IRB), and all participants provided informed consent before participation.

Tasks: Participants were instructed to view a chart from the prepared dataset and verbally articulate their findings using a think-aloud protocol before writing them down. When using VisCopilot, participants could continue writing to receive additional suggestions, revise their text as needed, or write complete sentences without assistance – much like their regular writing process. In contrast, when using the baseline system, participants wrote a complete sentence independently, after which a suggestion would appear; participants could then accept, skip, or revise their sentence.

Procedure: The on-site study lasted 60-90 minutes. Participants completed two counterbalanced tasks (one per system), each preceded by a tutorial and followed by system usability scale (SUS), a post-experiment questionnaire, and a semi-structured interview.

Measures: We employed four types of measures: (1) **interaction logs** to track sentence-chart link coverage and autocomplete acceptance rate; (2) the **SUS** for perceived usability; (3) a **post-experiment questionnaire** (7-point Likert scale) covering accuracy, efficiency, trustworthiness, and intent alignment (Q1-Q10, Fig. 5), with additional autocomplete-specific questions (Q11-Q19, Fig. 6 in Appendix C) for VisCopilot; and (4) **semi-structured interviews** for qualitative feedback.

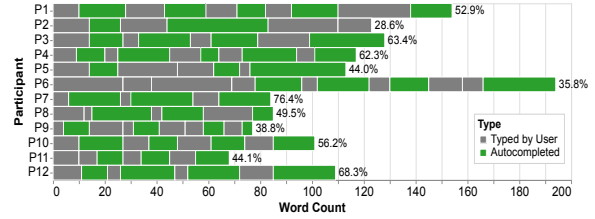


Figure 4: Proportion of text typed by the user (grey) versus autocompleted by VisCopilot (green) per participant. A grey bar followed by a green bar indicates the autocomplete completed the user-typed prefix.

6.2 Quantitative Results

Paired measures (SUS and Q1–Q10) were analyzed using Wilcoxon signed-rank tests given the within-subjects design. We did not apply multiple-comparison correction because the questionnaire items assessed distinct constructs.

Sentence-Chart Link Creation and Autocomplete Usage: We collected 24 articles from 12 participants. VisCopilot achieved 80.0% sentence-chart link coverage (44 of 55 sentences), compared to 57.7% for the baseline (30 of 52 sentences), indicating that autocomplete promotes more connections between narrative and data. Participants received 99 autocomplete suggestions with an average acceptance rate of 57.0%, and autocomplete contributed to 49.8% of words per sentence on average (Fig. 4), effectively assisting 9 out of 12 participants with 44% or more of their text.

System Usability Scale (SUS): Both VisCopilot and the baseline were generally well-received by participants, with both systems achieving high usability ratings. The baseline achieved a mean SUS score of 75.0, while VisCopilot got a significantly higher score of 84.6 ($p = 0.0246$), suggesting that participants found VisCopilot easier to use and overall more satisfying.

User Experience and Satisfaction: As shown in Fig. 5, VisCopilot was rated higher than the baseline in all aspects, with significant improvements in accuracy (Q1, $p = 0.01$) and user confidence (Q6, $p = 0.026$), suggesting that autocomplete assistance improves both precision and trust even when users retain full narrative control.

Satisfaction with VisCopilot’s autocomplete feature (Fig. 6 in Appendix C) was highly positive: participants found the suggestions easy to understand, factually correct, and concise (Q11-Q15), appreciated chart integration (Q16, Q17), needed only minor adjustments after accepting suggestions (Q18, $M = 5.3$), and 9 of 12 strongly agreed autocomplete improved writing efficiency (Q19).

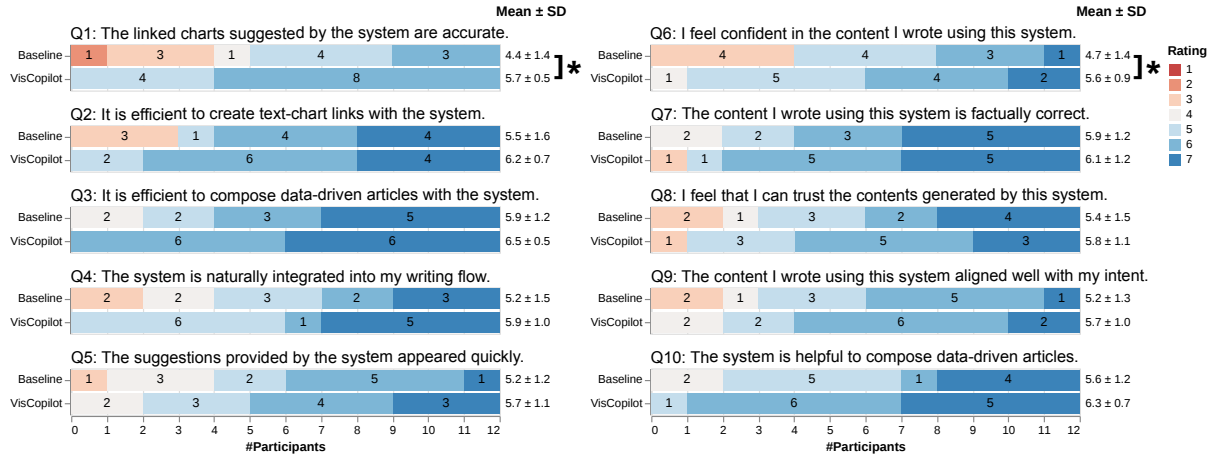


Figure 5: Participant ratings for VisCopilot and the baseline. VisCopilot was generally rated higher for all questions. The symbol * indicates a statistically significant difference with $p < 0.05$.

6.3 Qualitative Feedback from Interview

We summarize the interview findings below; the interview questions and additional details are provided in Appendix D.

System Preference: 11 of 12 participants preferred VisCopilot over the baseline, praising its efficiency and reduced writing effort (P9: “VisCopilot is more efficient and easier than writing from scratch”; P1: “...it often says it better than I would have”). P10 mentioned that “VisCopilot can complete sentences and link to the appropriate charts based on what you’ve written, making it more effective.”. On the other hand, P7 said “With VisLinker (baseline), you have to type out the whole sentence for it to understand what you mean,” and P9 noted it “takes more time because you have to think through the structure of each sentence.” While the majority favored VisCopilot, P11 preferred the baseline and explained that “I prefer VisLinker because its sentence-chart link suggestions are more aligned with what I want to describe.”

Although most participants leaned toward VisCopilot, three participants (P3, P4, P5) highlighted scenarios where the baseline could be useful. P4 noted that “Baseline is better if I’ve already finished writing and then want to generate charts.” P5 mentioned that “I’m a bit more confident with VisLinker since the sentences are my own.” Thus, the baseline may be preferred when users already have a clear direction or want greater control over their writing.

Autocomplete Quality was rated highly - most participants (P1, P4–P6, P8–P9, P11–P12) emphasized the high quality of VisCopilot’s autocompletes. P8 noted that “The quality is pretty good, even better than what I’d write.” Six participants

(P2, P4–P6, P9, P12) also appreciated how autocomplete enhanced their arguments by incorporating relevant data, as it “clearly shows comparisons, which makes it more persuasive” (P6). P2 shared, “I find it convenient that it directly states the percentage increase, which I might not calculate myself just from looking at the chart.”

In terms of trustworthiness, 9 of 12 participants encountered no factual errors and trusted the suggestions (P12: “I trust the outputs as they’re based on facts”). Although P4, P5, and P9 each encountered one factual error, they still preferred the system because errors were minor and rare. P9 reported that “It initially suggested ‘increase,’ but it should have been ‘decrease,’ so I corrected it. I still trust the system because I expect that I have to review it, just like when using ChatGPT.”

User Behavior: All participants reported that receiving autocompletes mid-writing did not interrupt their flow, as they typically “pause when being unsure how to phrase something” (P1). When suggestions were not entirely relevant, most participants (P1, P4–P11) naturally continued writing to guide the system.

When evaluating autocomplete suggestions, the majority of participants (P1, P2, P4, P6–P12) reported looking at the highlighted chart first. P4 shared, “I usually check the chart first to see if it’s what I want because it is faster.” This behavior illustrates how visualization served as a perceptual shortcut, helping participants reduce the cognitive effort required to review.

7 Discussion

We discuss whether the design requirements were fulfilled and offer design implications.

7.1 Effectiveness of Autocomplete-Assistance

VisCopilot addresses **DR1** by integrating narrative generation and sentence-chart linking into a single step. Participants produced significantly more sentence-chart links (80.0% vs. 57.7% coverage), and autocomplete contributed to nearly half of each sentence (49.8%), reducing manual writing effort. These results highlight VisCopilot’s balance between user control and system assistance — a middle ground between fully manual and fully automated approaches.

Beyond efficiency, VisCopilot meets **DR2** by grounding suggestions in verified data facts derived directly from their dataset, resulting in higher trustworthiness (Q6), correctness (Q7), and user confidence (Q8) compared to the baseline. Limiting autocompletes to single sentences also prevents verbosity and lowers cognitive load (Q12–Q16). More importantly, this mixed-initiative design creates a built-in safety check: in the rare instances where suggestions contained inaccuracies (such as P9’s inverted trend direction), participants easily spotted and corrected them by cross-referencing the synchronized chart highlights. Because the author actively inspects and approves every suggested fragment and link before insertion, this human-in-the-loop process provides an additional safeguard for the quality and correctness of the final article.

Regarding intent alignment (**DR3**), VisCopilot offers multiple suggestions, allowing users to select the option that best fits their intent. VisCopilot scored higher than the baseline in intent alignment (Q9), even though participants had full narrative control in the baseline. Our user study suggests that this collaborative setup strikes a natural balance between automated assistance and user agency. While participants benefited from autocomplete, they retained full authorship: when suggestions occasionally deviated from user intent, most participants (P1, P4–P11) naturally continued typing to guide the system (Sec. 6.3), demonstrating VisCopilot’s collaborative nature in keeping users in control of the final output.

7.2 Design Implications

A human-in-the-loop autocomplete design proves more effective than fully manual (Conlen et al., 2021; Sultanum et al., 2021) or fully automated (Choi and Jo, 2022; Sultanum and Srinivasan, 2023) approaches. Pairing each suggestion with its corresponding chart and highlighted data

facts mitigated cognitive load (Vaithilingam et al., 2022) (Q17), acting as a perceptual shortcut that reduces review effort. Participants rated autocompletes highly on correctness, clarity, and reliability (Q12–Q15), with qualitative feedback suggesting that intelligent assistance can improve narrative quality while preserving user autonomy.

Another insight relates to the gap between the technical evaluation and the user study. The *Reranked Hits@3* accuracy of 37.5% for two-thirds of a sentence reflects the challenge of pre-authored test data, where key data points often appear later in the sentence, making intent prediction more difficult before they are revealed. In practice, users naturally continue typing when suggestions are misaligned, providing richer context that steers the system toward their intent.

8 Conclusion

VisCopilot demonstrates a significant advancement in the field of data-driven article authoring by providing an effective balance between user control and intelligent assistance. The system integrates autocomplete functionality with accurate sentence-chart linking, offering fact-based suggestions that enhance both the accuracy and efficiency of data-article authoring. Our evaluation results indicate that VisCopilot outperforms a baseline in writing accuracy, efficiency, and user satisfaction, with higher sentence-chart link coverage and improved user trust. By leveraging RAG and LLMs, VisCopilot ensures content relevance to user intent while reducing factual errors and preventing overly verbose responses. Overall, VisCopilot’s design provides valuable insights for future intelligent authoring tools that enhance productivity without sacrificing user control or content quality.

9 Limitations

While VisCopilot achieved promising results, there are several limitations that point toward areas for future research. First, the system’s ability to handle more complex, multi-fact, or causal narratives remains limited, as it currently depends on a predefined set of fact types. However, VisCopilot’s architecture is fundamentally modular: the underlying fact generation and text-chart binding components can be readily extended or replaced with more advanced insight extraction engines. Similarly, the system can be expanded to support broader visualization types. Our prototype currently supports

bar, line, pie, choropleth map, and scatter charts; expanding beyond these standard chart types will make the system more versatile.

In addition, our current prototype focuses primarily on narrative writing assistance rather than the layout of visual representations. Although our supplementary video demonstrates the feasibility of linking across two charts, scaling to richer storytelling formats, such as multi-view dashboards, scrollytelling articles, animations, and transitions across visual idioms, introduces open challenges in cross-view grounding, visual layout synthesis, and discourse-level narrative coherence.

Furthermore, considerations around the ethical use of LLMs, such as potential biases in generated content (Lucy and Bamman, 2021; Li et al., 2024) and the potential to influence or constrain users’ writing styles or ideas (Padmakumar and He, 2023), require careful attention. Currently, VisCopilot generates candidates strictly by semantic relevance to align with users; however, this may restrict exploratory flexibility. Ensuring that future versions foster creativity without over-constraining user expression is crucial, which could be addressed by incorporating diversity-aware retrieval or alternative ranking strategies to balance intent alignment with conceptual divergence.

Finally, our study focused on a relatively small and homogenous participant group of 12 participants, primarily composed of students. A broader user base, including professional journalists and domain experts with different expertise levels and backgrounds, would provide a more comprehensive evaluation of the system’s effectiveness. Future research should address these limitations to better generalize the system’s benefits across a wider audience and more varied use cases.

Acknowledgments

This work was supported in part by the Taiwan National Science and Technology Council under Grants 112-2221-E-A49-057-MY3 and 114-2221-E-A49-117-MY3. ChatGPT was used solely for proofreading and language refinement; all scientific content and core writing are the authors’ own.

References

Kiran Ajani, Elsie Lee, Cindy Xiong, Cole Nussbaumer Knaflig, William Kemper, and Steven Franconeri. 2021. Declutter and focus: Empirically evaluating design guidelines for effective data communication.

IEEE Transactions on Visualization and Computer Graphics (TVCG), 28(10):3351–3364.

Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. 2024. Seven failure points when engineering a retrieval augmented generation system. *Preprint*, arXiv:2401.05856.

Yining Cao, Jane L E, Zhutian Chen, and Haijun Xia. 2023. Dataparticles: Block-based and language-oriented authoring of animated unit visualizations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–15.

Mia Xu Chen, Benjamin N. Lee, Gagan Bansal, Yuan Cao, Shuyuan Zhang, Justin Lu, Jackie Tsay, Yinan Wang, Andrew M. Dai, Zhifeng Chen, Timothy Sohn, and Yonghui Wu. 2019. Gmail smart compose: Real-time assisted writing. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2287–2295.

Zhutian Chen and Haijun Xia. 2022. CrossData: Leveraging text-data connections for authoring data documents. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*.

Jiwon Choi and Jaemin Jo. 2022. Intentable: A mixed-initiative system for intent-based chart captioning. In *2022 IEEE Visualization and Visual Analytics (VIS)*, pages 40–44. IEEE.

Cohere, Inc. Rerank-English-v3.0. <https://cohere.com/rerank>.

Matthew Conlen and Jeffrey Heer. 2018. Idyll: A markup language for authoring and publishing interactive articles on the web. In *The 31st Annual ACM Symposium on User Interface Software and Technology*, pages 977–989.

Matthew Conlen, Megan Vo, Alan Tan, and Jeffrey Heer. 2021. Idyll studio: A structured editor for authoring interactive & data-driven articles. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 1–12.

Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46.

Hai Dang, Sven Goller, Florian Lehmann, and Daniel Buschek. 2023. Choice over control: How users write with large language models using diegetic and non-diegetic prompting. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–17.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2026. The faiss library. *IEEE Transactions on Big Data*, 12(2):346–361.

- Pierre Dragicevic, Yvonne Jansen, Abhraneel Sarma, Matthew Kay, and Fanny Chevalier. 2019. Increasing the transparency of research papers with explorable multiverse analyses. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–15.
- Yu Fu, Dennis Bromley, and Vidya Setlur. 2025. [Dataweaver: Authoring data-driven narratives through the integrated composition of visualization and text](#). *Computer Graphics Forum*, 44.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.
- Katy Ilonka Gero, Vivian Liu, and Lydia Chilton. 2022. Sparks: Inspiration for science writing using language models. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference*, pages 1002–1019.
- GitHub, Inc. Github copilot. <https://github.com/features/copilot>. Accessed: 2026-05-20.
- Shan Hao, Zezhong Wang, Benjamin Bach, and Larissa Pschetz. 2024. Design patterns for data-driven news articles. In *CHI’24: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*.
- Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*, pages 364–381. Springer.
- Mohammed Saidul Islam, Md Tahmid Rahman Laskar, Md Rizwan Parvez, Enamul Hoque, and Shafiq Joty. 2024. DataNarrative: Automated data-driven storytelling with visualizations and texts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19253–19286.
- Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.
- Shankar Kantharaj, Rixie Tiffany Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. 2022. Chart-to-text: A large-scale benchmark for chart summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4005–4023.
- Dae Hyun Kim, Seulgi Choi, Juho Kim, Vidya Setlur, and Maneesh Agrawala. 2023. Emphasischecker: A tool for guiding chart and caption emphasis. *IEEE Transactions on Visualization and Computer Graphics*.
- Dae Hyun Kim, Vidya Setlur, and Maneesh Agrawala. 2021. Towards understanding how readers integrate charts and captions: A case study with line charts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–11.
- Nicholas Kong, Marti A Hearst, and Maneesh Agrawala. 2014. Extracting references between text and charts via crowdsourcing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 31–40.
- Chufan Lai, Zhixian Lin, Ruike Jiang, Yun Han, Can Liu, and Xiaoru Yuan. 2020. Automatic annotation synchronizing with textual description for visualization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13.
- Shahid Latif, Zheng Zhou, Yoon Kim, Fabian Beck, and Nam Wook Kim. 2021. Kori: Interactive synthesis of text and charts in data documents. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 28(1):184–194.
- Mina Lee, Katy Ilonka Gero, John Joon Young Chung, Simon Buckingham Shum, Vipul Raheja, Hua Shen, Subhashini Venugopalan, Thiemo Wambsganss, David Zhou, Emad A Alghamdi, Tal August, Avinash Bhat, Madiha Zahrah Choksi, Senjuti Dutta, Jin L C Guo, Md Naimul Hoque, Yewon Kim, Simon Knight, Seyed Parsa Neshaei, and 17 others. 2024. [A design space for intelligent and interactive writing assistants](#). In *Proceedings of the CHI Conference on Human Factors in Computing Systems*.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-Tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Haotian Li, Yun Wang, and Huamin Qu. 2024. Where are we so far? Understanding data storytelling tools from the perspective of human-AI collaboration. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–19.
- Haotian Li, Lu Ying, Haidong Zhang, Yingcai Wu, Huamin Qu, and Yun Wang. 2023. Notable: On-the-fly assistant for data storytelling in computational notebooks. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–16.
- Yanna Lin, Haotian Li, Leni Yang, Aoyu Wu, and Huamin Qu. 2023. InkSight: Leveraging sketch interaction for documenting chart findings in computational notebooks. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):944–954.
- Can Liu, Liwenhan Xie, Yun Han, Datong Wei, and Xiaoru Yuan. 2020. AutoCaption: An approach to generate natural language description from visualization automatically. In *2020 IEEE Pacific visualization symposium (PacificVis)*, pages 191–195.

- Li Lucy and David Bamman. 2021. Gender and representation bias in GPT-3 generated stories. In *Proceedings of the third workshop on narrative understanding*, pages 48–55.
- Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878.
- OpenAI, Inc. Embedding model. <https://platform.openai.com/docs/guides/embeddings>.
- Vishakh Padmakumar and He He. 2023. Does writing with language models reduce content diversity? *arXiv preprint arXiv:2309.05196*.
- Pew Research Center. 2011. Internet gains on television as public’s main news source. <https://www.pewresearch.org/politics/2011/01/04/internet-gains-on-television-as-publics-main-news-source/>. Accessed: 2026-05-20.
- Pew Research Center. 2023. Teens, social media and technology 2023. <https://www.pewresearch.org/internet/2023/12/11/teens-social-media-and-technology-2023/>. Accessed: 2026-05-20.
- Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2016. Vega-lite: A grammar of interactive graphics. *IEEE TVCG*, 23(1):341–350.
- Leixian Shen, Yizhi Zhang, Haidong Zhang, and Yun Wang. 2024. Data Player: Automatic generation of data videos with narration-animation interplay. *IEEE TVCG*, 30(1):109–119.
- Danqing Shi, Xinyue Xu, Fuling Sun, Yang Shi, and Nan Cao. 2020. Calliope: Automatic visual data story generation from a spreadsheet. *IEEE TVCG*, 27(2):453–463.
- Arjun Srinivasan, Steven M Drucker, Alex Endert, and John Stasko. 2018. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):672–681.
- Nicole Sultanum, Fanny Chevalier, Zoya Bylinskii, and Zhicheng Liu. 2021. Leveraging text-chart links to support authoring of data-driven articles with VizFlow. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–17.
- Nicole Sultanum and Arjun Srinivasan. 2023. DATATALES: Investigating the use of large language models for authoring data-driven articles. In *2023 IEEE Visualization and Visual Analytics (VIS)*, pages 231–235. IEEE.
- Simeng Sun, Wenlong Zhao, Varun Manjunatha, Rajiv Jain, Vlad Morariu, Franck Dernoncourt, Balaji Vasanth Srinivasan, and Mohit Iyyer. 2021. IGA: An intent-guided authoring assistant. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5972–5985.
- Benny Tang, Angie Boggust, and Arvind Satyanarayan. 2023. VisText: A benchmark for semantically rich chart captioning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7268–7298.
- Priyan Vaithilingam, Tianyi Zhang, and Elena L. Glassman. 2022. Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–7.
- Vega. 2024. Vega dataset. <https://vega.github.io/vega-datasets/>.
- Bret Victor. 2011. Explorable explanations. <http://worrydream.com/ExplorableExplanations>.
- Yun Wang, Zhida Sun, Haidong Zhang, Weiwei Cui, Ke Xu, Xiaojuan Ma, and Dongmei Zhang. 2020. Datashot: Automatic generation of fact sheets from tabular data. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):895–905.
- Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story writing with large language models. In *Proceedings of the 27th International Conference on Intelligent User Interfaces*, pages 841–852.
- Zhuohao Zhang, Sana Malik, Shunan Guo, Jane Hoffswell, Ryan Rossi, Fan Du, and Eunye Koh. 2022. Understanding business analysts’ needs for data report authoring. *Proceedings of the EuroVA*.
- Qiyu Zhi, Alvitta Ottley, and Ronald Metoyer. 2019. Linking and layout: Exploring the integration of text and visualization in storytelling. *Computer Graphics Forum*, 38(3):675–685.
- Yun Zhu, Xiayu Chen, Lei Shu, Bowen Tan, Xinying Song, Lijuan Liu, Maria Wang, Jindong Chen, and Ning Ruan. 2023. An efficient conversational smart compose system. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 456–462.
- Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. 2023. Open-source large language models are strong zero-shot query likelihood models for document ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8807–8817.

A Data Fact Types

Table 2 lists the nine data fact types supported by VisCopilot.

B Prompt Template

Listing 1 shows the prompt template that VisCopilot uses to inform the LLM to synthesize the prefix and data fact.

C User Satisfaction for Autocompletes

In the post-experiment questionnaire of the user study (Section 6), we also asked the participants to rate their satisfaction with VisCopilot’s autocompletes. Fig. 6 shows the questions and rating results. The participants found the suggestions easy to understand, factually correct, and concise (Q11-Q15), appreciated chart integration (Q16, Q17), needed only minor adjustments after accepting suggestions (Q18, $M = 5.3$), and 9 of 12 strongly agreed autocomplete improved writing efficiency (Q19).

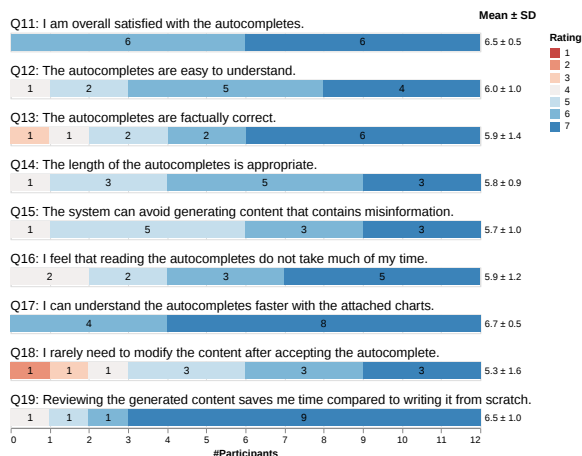


Figure 6: User satisfaction ratings with VisCopilot’s autocompletes.

D Interview Questions and Results

D.1 Interview Questions

System Preference

- Which system do you prefer between the two, and why?
- What specific features or functions of VisCopilot did you like or dislike?
- How did you feel when using VisCopilot compared to the baseline?
- What is the key differentiator between VisCopilot and the baseline for you?

- In what scenarios would you prefer the baseline over VisCopilot?

Autocomplete Quality

- How do you perceive the quality of the content generated by autocomplete?
- How much do you trust the autocomplete suggestions?
- Did the autocomplete feature improve your writing efficiency?
- Can you give examples where autocomplete aligned or did not align with your intent?

User Behavior

- When evaluating autocomplete suggestions, do you look at the text or the chart first?
- How did the highlighted charts influence your understanding of the suggestions?
- Did the autocomplete feature interrupt your writing flow?
- Did the suggestions inspire further content or ideas for your article?
- When suggestions were unrelated or incorrect, did you find it intuitive to continue writing?

D.2 Interview Results

This section provides extended excerpts complementing the summary in Section 6.3.

System Preference: 11 out of 12 participants preferred VisCopilot over the baseline. They praised VisCopilot for its efficiency and ability to reduce writing effort. P9 noted that “*VisCopilot is more efficient and easier than writing from scratch.*” P10 mentioned that “*VisCopilot can complete sentences and link to the appropriate charts based on what you’ve written, making it more effective,*” echoed by P12, who appreciated that “*What I like most is that it can automatically complete my sentences and at the same time generate charts.*” While the majority favored VisCopilot, P11 preferred the baseline and explained that “*I prefer VisLinker (baseline) because its sentence-chart link suggestions are more aligned with what I want to describe.*”

Nine participants (P1-P3, P5-P8, P10, P12) found that VisCopilot significantly reduced the effort needed to compose data-driven articles. P1 noted that “*The autocomplete feature is quite helpful for me because sometimes I know what I want to say, but I can’t express it well. When it suggests something, it often says it better than I would have.*”

Fact Type	Breakdown (b_i)	Measure (m_i)	Focus (x_i)	Derived Value (V_d)	Template
Value	C/T	N	1	derived value	The $\{m_i\}$ of x_i is $\{V_d\}$ when $\{s_i\}$.
Difference	C/T	N	2	difference value	The $\{m_i\}$ of $\{x_i[1]\}$ is $\{V_d\}$ higher/lower than that of $\{x_i[2]\}$ when $\{s_i\}$.
Proportion	C/T	N	1	percentage	The $\{x_i\}$ accounts for $\{V_d\}$ of the $\{m_i\}$ when $\{s_i\}$.
Rank	C/T	N	3	top 3 / last 3	In the $\{m_i\}$ ranking of different $\{b_i\}$, the $\{V_d\}$ are $\{x_i[1]\}$, followed by $\{x_i[2]\}$, then $\{x_i[3]\}$, when $\{s_i\}$.
Extreme	C/T	N	1	lowest / highest	The $\{V_d\}$ value of the $\{m_i\}$ is $\{x_i\}$ when $\{s_i\}$.
Outlier	C/T	N	1	-	The $\{m_i\}$ of $\{x_i\}$ is an outlier when compare with that of other $\{b_i\}$ when $\{s_i\}$.
Association	C/T	$N \times N$	≥ 2	Pearson's r	There is a strong/moderate/weak relationship between $\{m_i[1]\}$ and $\{m_i[2]\}$, with the Pearson correlation coefficient of $\{V_d\}$, when $\{s_i\}$.
Outlier (scatter)	C/T	$N \times N$	1	derived value	$\{x_i\}$ is an outlier with $\{m_i[1]\}$ is $\{V_d[1]\}$ and $\{m_i[2]\}$ is $\{V_d[2]\}$ when $\{s_i\}$.
Trend	T	N	≥ 2	increasing / waver- ing / decreasing	The $\{V_d\}$ trend of $\{m_i\}$ over $\{b_i\}$ from $\{x_i[1]\}$ to $\{x_i[n]\}$ when $\{s_i\}$.

Table 2: Definitions of the nine data fact types supported by VisCopilot. In the *Breakdown* and *Measure* column, N , C , T indicate the numerical, categorical, and temporal data types, respectively. In the *Template*, $\{s_i\}$ is the subspace, where it could indicate the entire dataset, or a subset defined by a set of data filters (Shi et al., 2020).

P6 mentioned that “it makes the sentence better while keeping the same meaning.” P10 further emphasized this: “VisCopilot helps save writing effort by generating sentences that are close to the intended meaning.” On the other hand, P7 said “With VisLinker, you have to type out the whole sentence for it to understand what you mean,” and P9 noted it “takes more time because you have to think through the structure of each sentence.”

Although most participants leaned toward VisCopilot, three participants (P3, P4, P5) highlighted scenarios where the baseline could be useful. P4 noted that “Baseline is better if I’ve already finished writing and then want to generate charts.” P5 mentioned that “I’m a bit more confident with VisLinker since the sentences are my own.” Thus, the baseline may be preferred when users already have a clear direction or want greater control over their writing.

Autocomplete Quality: Most participants (P1, P4-P6, P8-P9, P11-P12) emphasized the high quality of VisCopilot’s autocompletes. P8 noted that “The quality is pretty good, even better than what I’d write.” Six participants (P2, P4-P6, P9, P12) also appreciated how autocomplete enhanced their arguments by incorporating relevant data, as it “clearly shows comparisons, like how many times something is more than another, which makes it more persuasive” (P6), and “makes the content feel more concrete than just having descriptive text” (P9). P2 shared, “I find it convenient that it directly states

the percentage increase, which I might not calculate myself just from looking at the chart.”

In terms of trustworthiness, 9 out of 12 participants reported no factual errors and were inclined to trust the autocomplete suggestions, though they still reviewed them carefully. P12 stated “I trust the outputs as they’re based on facts, all I need is to confirm they align with my intended meaning.” Although P4, P5, and P9 each encountered one factual error, they still preferred the system because errors were minor and rare. P9 reported that “It initially suggested ‘increase,’ but it should have been ‘decrease,’ so I corrected it. I still trust the system because I expect that I have to review it, just like when using ChatGPT.”

User Behavior: All participants reported that receiving autocompletes mid-writing did not interrupt their flow, as they typically “pause when being unsure how to phrase something” (P1). When suggestions were not entirely relevant, most participants (P1, P4-P11) naturally continued writing to guide the system, or “give it more keywords until it adjusts to what I’m trying to describe” (P6).

When evaluating autocomplete suggestions, the majority of participants (P1, P2, P4, P6-P12) reported looking at the highlighted chart first. P4 shared, “I usually check the chart first to see if it’s what I want because it is faster.” This behavior illustrates how visualization served as a perceptual shortcut, helping participants reduce the cognitive effort required to review multiple textual sugges-

Using the provided data information, FACT, CONTEXT, and PROMPT (an incomplete sentence), complete the PROMPT by responding a full sentence that incorporates the FACT in a coherent manner that fits the CONTEXT. If no suitable completion is possible for the FACT, respond with NO_ANSWER.

Name of the Dataset: game_market.csv
Columns in the Dataset: Online gaming device,% market share
FACT: The Mobile accounts for 25.24% of the % market share.
CONTEXT: The gaming industry has seen significant shifts in market share over the years.
PROMPT: mobile games account

ANSWER: Mobile games account for about a quarter (25.24%) of the market share.
=====

Name of the Dataset: [dataset]
Columns in the Dataset: [columns]
FACT: [fact]
CONTEXT: [context]
PROMPT: [prompt]

ANSWER:

Listing 1: The prompt template used in VisCopilot for the LLM to synthesize the *prefix* and the data fact with context. Brackets indicate placeholder variables filled during autocompletion.

tions and highlighting visualization's efficiency in quickly conveying key points.