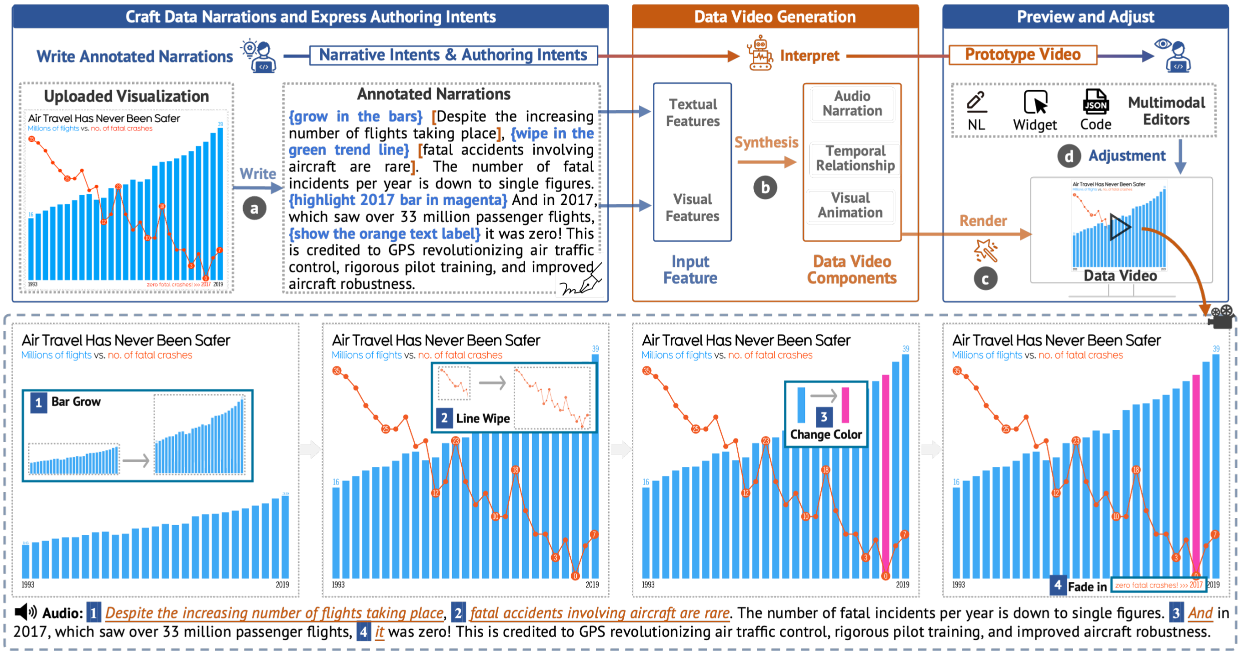
Data Playwright: Authoring Data Videos with Annotated Narration

Leixian Shen, Haotian Li, Yun Wang, Tianqi Luo, Yuyu Luo, and Huamin Qu



{grow in the bars} [Despite the increasing number of flights taking place], {wipe in the green trend line} [fatal accidents involving aircraft are rare]. The number of fatal incidents per year is down to single figures. {highlight 2017 bar in magenta} And in 2017, which saw over 33 million passenger flights, {show the orange text label} it was zero! This is credited to GPS revolutionizing air traffic control, rigorous pilot training, and improved aircraft robustness.

Audio: 1 Despite the increasing number of flights taking place, 2 fatal accidents involving aircraft are rare. The number of fatal incidents per year is down to single figures. 3 And in 2017, which saw over 33 million passenger flights, 4 it was zero! This is credited to GPS revolutionizing air traffic control, rigorous pilot training, and improved aircraft robustness.

Fig. 1: Creating data videos with Data Playwright: (a) Users craft text narrations based on visualizations and incorporate natural language commands for data video authoring as inline annotations, known as *annotated narration*. (b) The automatic interpreter processes the syntactically annotated narration and visualizations to synthesize audio narration, visual animations, and their temporal relationship. (c) The synthesized components are rendered into a high-quality data video. (d) Users can further preview and fine-tune the prototype video with multimodal editors. The bottom portion showcases the final data video with narration-animation interplay.

Abstract— Creating data videos that effectively narrate stories with animated visuals requires substantial effort and expertise. A promising research trend is leveraging the easy-to-use natural language (NL) interaction to automatically synthesize data video components from narrative content like text narrations, or NL commands that specify user-required designs. Nevertheless, previous research has overlooked the integration of narrative content and specific design authoring commands, leading to generated results that lack customization or fail to seamlessly fit into the narrative context. To address these issues, we introduce a novel paradigm for creating data videos, which seamlessly integrates users' authoring and narrative intents in a unified format called *annotated narration*, allowing users to incorporate NL commands for design authoring as inline annotations within the narration text. Informed by a formative study on users' preference for annotated narration, we develop a prototype system named Data Playwright that embodies this paradigm for effective creation of data videos. Within Data Playwright, users can write annotated narration based on uploaded visualizations. The system's interpreter automatically understands users' inputs and synthesizes data videos with narration-animation interplay, powered by large language models. Finally, users can preview and fine-tune the video. A user study demonstrated that participants can effectively create data videos with Data Playwright by effortlessly articulating their desired outcomes through annotated narration.

Index Terms—Data Video, Intent, Natural Language, Annotated Narration, Large Language Model

1 Introduction

Data videos, a popular medium of data storytelling, can effectively convey data stories by combining animated data visualizations with audio narration [6, 11, 37]. However, creating such videos requires significant effort and expertise in various tasks, including crafting narration, designing animations, recording audio, and aligning the audio narration and visual animations. These tasks often pose challenges, especially for novices [49]. As a result, there has been a growing number of authoring tools designed to facilitate the process (*e.g.,* Data Player [49], DataParticles [8], and AutoClips [51]).

One promising research thread is leveraging the convenience of natural language (NL) interaction to automatically synthesize data video components. Specifically, two primary types of authoring paradigms have emerged. The first type is command-driven natural language interfaces [28,44]. These interfaces interpret users' NL commands (*e.g.,* "*Highlight top 2 models in pink*", "*Grow in the nine bars*") for specific authoring tasks and generate corresponding visual outputs, such as

- *L. Shen, H. Li, and H. Qu are with The Hong Kong University of Science and Technology. E-mail: {lshenaj, haotian.li}@connect.ust.hk; huamin@cse.ust.hk*
- *Y. Wang is with Microsoft. E-mail: wangyun@microsoft.com.*
- *T. Luo and Y. Luo are with The Hong Kong University of Science and Technology (Guangzhou). E-mail: yuyuluo@hkust-gz.edu.cn.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

updated charts [64] and motion graphic animations [61]. However, they overlook the narration context, leading to inconsistencies between the generated visuals and the overall video. As a result, users need to invest significant manual effort in precisely aligning the visual design with the narratives. The second type takes text narrations as input and analyzes the relations between narrative patterns and visuals to recommend suitable visual presentations [8, 26, 49, 65]. For instance, most recently, Data Player [49] leverages large language models (LLMs) to establish connections between narration segments and visual elements, and then employs constraint programming to recommend animation sequences for the text-visual links. However, the end-to-end automatic workflow fails to incorporate users' diverse authoring preferences into the creation process, falling into the "one-design-fits-all" pitfall. Overall, previous research has paid little attention to the integration of data narrations and specific authoring commands. As a result, the generated results may lack customization or struggle to seamlessly align with the narration context. Furthermore, this separation hampers adequate support for the iterative workflow of creating data videos, where both narrations and visual designs are frequently modified in tandem.

To address these issues, we explore a novel NL-based paradigm for data video creation, which seamlessly integrates users' authoring and narrative intents in a unified format called *annotated narration*. In this paradigm, users can naturally incorporate their NL commands for data video authoring by writing them as inline annotations within the narration text, as shown in Fig. 1-a, effectively merging the processes of crafting text narrations and authoring data videos. The user's input, *i.e.,* annotated narration, serves as a new shared intermediate medium between users and AI systems [18]. Users can express their diverse intents by writing annotated narration, while AI systems parse the annotated narration for subsequent generations. This paradigm creates a user-friendly experience where users can effortlessly articulate their desired outcomes and directly achieve the final product with seamless translations from their intents.

We further develop a prototype system, Data Playwright, that realizes this paradigm for effective creation of data videos. With Data Playwright , users can upload their visualizations, write annotated narrations, and automatically generate customized data videos based on their inputs. To gain a deeper understanding of users' actions in data video authoring and their preferences for writing annotated narration, we conduct a formative study and literature review. Based on the findings, we formulate a user-friendly syntax for *annotated narration*. Following the syntax, we implement an automatic interpreter to understand users' diverse inputs and synthesize data video components. The interpreter parses the syntax and utilizes text-to-speech services to generate audio narration while establishing temporal and semantic relations between NL commands, narration segments, and potential animations. It further leverages LLMs to extract target visual elements, animation effects, and properties for each animation. The synthesized components are then rendered into a coherent data video. Finally, Data Playwright allows users to preview data videos and iteratively fine-tune the generated outputs through multimodal editors.

We conduct a user study to evaluate the utility and expressiveness of Data Playwright, where participants complete data video creation tasks by writing annotated narration. The produced data videos also form our example gallery. Finally, we discuss insights learned from the user study and potential future directions.

The main contributions of this paper are as follows:

- An innovative paradigm for data video creation that seamlessly merges users' effortlessly expressed narrative intents and authoring intents into a unified format called *annotated narration*.
- A formative study to understand users' data video authoring actions and gain insights into their preferences and patterns when writing *annotated narration*.
- A prototype system, Data Playwright, that embodies the paradigm and features an automatic interpreter to translate users' visualizations and annotated narration into data videos.
- A user study with an example gallery to assess the effectiveness and expressiveness of Data Playwright.

2 Related Work

Data Playwright draws upon prior works in data video creation, natural language interaction, and user intent expression for storytelling.

2.1 Data Video Creation

Data video is one of the popular data storytelling genres [42]. Prior studies have demonstrated that compared to static presentations, data videos, with their visual animations and audio narration, offer additional channels of communication, resulting in improved information transformation and audience engagement [6, 11, 13, 37]. Empirical research has also investigated different aspects of data videos, such as visual narrative [5], animated data-driven graphics [19,52,59], narrative transitions [58], and interplay of narrations and animations [11], etc.

On top of these findings, various manual authoring and programming tools have been developed to facilitate the creation of data videos [9,59]. These tools employ various authoring paradigms, such as programming languages [16, 21, 75], keyframe-based animation generation [15, 60], and presets and templates [7, 24]. Additionally, general video creation tools like Adobe After Effects provide extensive control over visual elements through animation keyframing and presets, but they often require significant manual effort. Furthermore, DataClips [7] simplifies the process by enabling users to effortlessly compose a sequence of suitable clips for visualizing different data facts. Recently, WonderFlow [65] introduced a narration-driven design pipeline, which allows users to interactively specify text-visual connections and select a suitable animation preset for the established connections, streamlining the manual effort involved in the creation process. Despite the diverse interaction design present in these manual tools, they still require significant manual effort and involve a learning curve for tool-specific operations.

To eliminate manual effort, researchers have made significant progress in developing automatic approaches. For instance, InfoMotion [63] empowers the automatic generation of animated infographics based on motion graphical properties and structures. Gemini2 [22] enhances Gemini [21] by offering suggestions for keyframe transitions. AutoClips [51] constructs a fact-driven clip library and automatically generates videos from a sequence of data facts. Roslingifier [53] automatically generates visual highlights and playback narratives for animated scatterplots. Live Charts [73] revive static visualizations by decomposing the chart information and explaining it with animations and audio narration. Data Player [49] utilizes LLMs to link narration segments and visual elements semantically, recommends animations for text-visual links using constraint programming, and renders the animation sequence with automatically generated audio narration into a data video. While these end-to-end workflows can efficiently generate data videos from users' static materials, they cannot encode users' diverse and evolving authoring preferences into the creation process.

Overall, manual tools tend to be tedious and require expertise, while automatic methods often overlook users' diverse authoring intents. This paper presents a new data video creation tool, Data Playwright, where users can effortlessly express their narrative and authoring intents by writing annotated narration. Data Playwright then dispatches its automatic interpreter to translate users' inputs into data videos, ensuring a seamless transfer of users' creative vision into the final output.

2.2 Natural Language Interaction

Natural language enables users to freely and intuitively express diverse intents, without requiring explicit tool-specific knowledge. The inherent convenience nature has nourished a series of natural language interfaces across various domains [44]. These interfaces take the user's NL queries as input, extract relevant information from the input text as an abstraction layer, and finally translate it into appropriate actions or representations. In the field of visualization research, prior work has successfully mapped users' diverse NL commands or queries into the intent space of visualization authoring [64], spreadsheet formula generation [55], web style customization [20], color refinement [50], question answering [66], infographics design [14], sports video augmentation [10], etc. However, these interfaces tend to focus on specific authoring tasks while overlooking the overall data narration context when it comes to creating complex storytelling forms like data videos.

Another line of work adopts a narrative-driven paradigm to facilitate data story creation, including creating data documents [25,57], authoring animated unit visualizations [8], augmenting audio podcasts [68], and generating digital content [67]. For example, Kori [25] and VizFlow [57] build text-chart references to enhance the reading experience of data-driven articles. Charagraph [34] dynamically generates real-time charts and annotations for data-rich paragraphs to improve the reading experience. DataParticles [8] utilizes the connection between text, data, and visualizations to facilitate the exploration of story narratives and visualizations. Data Player [49] automates narration-centric data video creation with LLMs. Crosscast [68] automatically enhances audio travel podcasts with visuals retrieved online. However, these end-to-end automatic workflows from narrations to visuals usually fail to encode users' diverse authoring intents for storytelling forms.

In this paper, we align with the NL-based approach and focus on data video authoring. In addition, we introduce the concept of *annotated narration*, integrating NL commands for data video authoring into text narration as inline annotations, seamlessly combining data narration crafting and data video authoring.

2.3 Expressing User Intent for Storytelling

User intents play a crucial role in storytelling tasks, and various interaction modalities have been used to facilitate the expression of users' diverse intents. For example, WIMP interface-based tools, such as MEDLEY [39], TaskVis [45,47], and PyGWalker [74], provide users with a familiar graphical user interface, allowing them to communicate their tasks through various actions. Natural language-based tools, such as Sporthesia [10] and Text-to-Viz [14], enable users to express their intents intuitively through written or spoken language, utilizing NLP techniques to interpret their input into actions or representations. Sketch-based tools allow users to communicate their intents through freehand sketches or annotations [27,32]. For instance, InkSight [32] can document chart findings by allowing users to sketch atop visualizations. Example-based tools, such as Wakey-Wakey [69], allow users to provide references to convey their desired style. Ivy [36] and GALVIS [46] also enable users to browse examples for inspiration and adopt one to start their design. In addition, behavior-based tools infer users' intents by analyzing their usage preferences or patterns [23].

Inspired by the convenience of natural language interaction [28], the important narrative role of NL (as text narration) in data videos [11], and the impressive NL understanding abilities of existing AI models [72], in this paper, we embrace NL as the primary interaction modality, and integrate other modalities for users to fine-tune the generated results.

3 Formative Study

The study aims to understand: (1) How users author data videos, including their actions and intents in the authoring process; (2) How users express their authoring intents through annotated narration, *i.e.,* using a combination of natural language and common notations as annotations within the text narration.

3.1 Participants and Procedure

The study involved six experts (donated as E1 to E6), each with expertise in creating different types of animated data stories. The experts included motion graphic designers, visualization researchers, journalists, and film editors. They have all used professional software such as Adobe After Effects or simplified tools like iMovie and Microsoft PowerPoint to produce data videos, accumulating at least five years of experience ($M = 5.83$, $SD = 0.75$).

The study procedure consists of retrospective analysis [40] and semi-structured interviews. Firstly, each expert was asked to showcase their previously created data videos or animated data stories, explaining the components and corresponding creation workflow. They were specifically prompted to offer detailed explanations of the authoring intents associated with particular software or coding actions, as well as the consequent effects. Furthermore, they were encouraged to reflect on the difficulties encountered throughout the process. Next, participants were asked to use NL to describe their specific authoring actions assuming an intelligent agent could execute these tasks for them. In the semi-structured interviews, we collected four well-designed data videos from real-world storytelling practices, along with their accompanying text narration. We prompted the participants to imagine a data video authoring tool that could accurately understand their commands. Then, to understand users' natural thought processes, they were asked to encode their authoring intents for reproducing the given data videos as inline annotations (*e.g.,* NL commands and notations) in the text narration, clearly describing their authoring actions. Participants were encouraged to think aloud during the study.

3.2 Findings

The study provided valuable insights into users' natural ways of thinking and authoring, including common data video creation practices (Sec. 3.2.1) and observations of users' annotation patterns (Sec. 3.2.2).

3.2.1 Data Video Creation Practices

General Workflow: Creating data videos is a tedious and skill-intensive process, involving diverse user actions and intents across various stages, as illustrated in Tab. 1. Based on the visual data analysis results, users need to write text narration, record audio narration, organize visual elements, design visual animations, time-align audio narration and visual animations, and render the data video. The workflow is not strictly linear, as users may need to revisit and modify tasks iteratively, as noted by E3.

User Preferences for Workflow Streamlining: Based on the understanding of the diverse user actions and intents in the data video authoring workflow, we gathered insights into users' preferences for human and AI roles to satisfy their authoring intents, as shown in Tab. 1 (User Preference). Specifically, participants commonly craft text narration based on insights from visualizations (S1). Most participants (5/6) expressed a desire for a complete lead in this stage, while also expecting automatic conversion of the text into audio (S2). Participants (6/6) universally found it cumbersome to semantically link textual and visual elements (S3), design animations (S4), and align narrations and animations (S5) due to the numerous tool-specific operations across multiple tools, aligning with prior studies [8,49,65]. Consequently, they expect to efficiently express their ideas in these stages and have the system rapidly implement them. Additionally, we observed that participants readily incorporated NL commands alongside semantic-related narration segments, denoting them with commonly used notations. Furthermore, participants (6/6) unanimously expected to complete all tasks in one platform that supports iterative preview and adjustment (S6).

3.2.2 Observations

All participants expressed interest in the authoring experience of annotating narrations with NL commands and notations, considering it especially useful when they need to navigate unfamiliar user interfaces with complex workflows. Here we summarize the common design practices and observations for stages (S3-S5) where users anticipated the use of a mixed-initiative approach based on annotated narration:

Visual element reference: When referring to elements in the visualizations (S3), participants mainly used two expression types: data-driven and visual-driven. Data-driven expressions frequently include data labels (*e.g.,* "*wipe in the USA line*", "*highlight 1995 and 1998 bars*"), which involve specifying the data column and value information. Visual-driven expressions often encompass information related to the visualization appearance (*e.g.,* color, shape, and position), such as "*shine the second bar from the left*" and "*fade out the blue lines*". Additionally, some participants (E2, E4, E5) frequently omitted information in their NL commands that had already been mentioned in the text narration, such as, "*...{fade lines of other countries} Japanese outlays almost half across the two lost decades,...*".

Animation design: Participants commonly incorporated three prevalent animation behaviors in their NL commands: entrance, emphasis, and exit. When conveying animation effects (T4.2), users may directly specify the animation effect (*e.g.,* "fade in", "change color"), or they sometimes simply described the animation behavior (*e.g.,* "show", "enter"). Furthermore, when it comes to the opening animation (T4.1), E5 said, "*sometimes describing the opening effect in words is tough for me.*

Table 1: General workflow of crafting data videos, corresponding actions and intents in each stage, and user preferences for workflow streamlining.

Stage	Actions and User Intents	User Preference
S1. Write text narration	T1.1. Describe the data insights in the visualizations	Human lead
	T1.2. Design the narrative structure	Human lead
S2. Record audio narration	T2.1. Record audio based on the text narration	AI lead
S3. Organize visual elements	T3.1. Group the semantic-related elements	Mixed-initiative
	T3.2. Build semantic references between narration segments and visual elements	Mixed-initiative
S4. Design visual animations	T4.1. Design the opening animation of background elements	Mixed-initiative
	T4.2. Specify behavior animation types and effects	Mixed-initiative
	T4.3. Adjust animation properties	Mixed-initiative
S5. Time-align narration and animations	T5.1. Listen to audio repeatedly to identify timestamps for triggering animations	Mixed-initiative
	T5.2. Arrange animations on the timeline and adjust the duration	Mixed-initiative
S6. Render data video	T6.1. Synthesize the coordinated animations and audio into a data video	AI lead

There are so many visual elements involved, and honestly, I don't really fuss over these details. I just want to grab audience's attention right from the start." Moreover, NL commands can encompass animation properties, such as color, direction, order, etc (T4.3). Additionally, users may also provide additional explanations, such as the purpose of the animation (*e.g.,* "*hide all lines here to highlight axes*") or the status of other visual elements (*e.g.,* "*..., lines still faded out*").

Time alignment of narration and animations: Participants (6/6) consistently viewed the text narration as the timeline of the data video. They strategically placed NL commands near semantically relevant narration segments to indicate animation triggering times (T5.1). Regarding the duration of animations (T5.2), they sometimes lacked clarity and may overlook specifying this information in their NL commands. As expressed by E1, "*sometimes I just want specific visual elements to appear, with a short default duration.*" However, participants still expressed a desire for more precise control over the animation duration beyond default configurations. In particular, E2-E6 preferred using notations to mark semantically relevant narration segments, with the segment length corresponding to the desired animation duration. And E1 favored continuing to use NL to describe the animation duration.

Furthermore, participants may use commands like "*same as above*" to indicate their intention to reuse animations. When dealing with parallel content, users preferred a single NL command over separate commands for each parallel segment.

4 Data Playwright

In this section, we first describe the design considerations and an overview of Data Playwright. Then, we introduce the process of writing annotated narration. Next, we discuss the underlying data video specification and the automatic interpretation process from annotated narration to video specifications. Finally, we present the user interface.

4.1 Design Consideration

We identify a set of design considerations from the formative study and literature review:

C1. Seamlessly integrate the process of crafting text narration with authoring data videos. Data video creation usually involves separate stages of crafting narrations and authoring videos [49, 51, 73]. The system should further reduce the gap by integrating the two stages. So we propose a new NL-based paradigm that combines narrative and authoring intents in a unified format called *annotated narration*.

C2. Unleash human preference and harness AI to reduce manual labor. Data video creation involves significant creativity and labor. To optimize the process, it is important to divide and allocate tasks between human users and AI systems, leveraging the strengths of each while adapting to human preferences (Tab. 1) [18, 30].

C3. Enable user-friendly expression of authoring intents with a minimal learning curve. The system should empower users to flexibly express their intents by writing annotated narration. To ensure ease and accuracy of parsing, users should be encouraged to follow a syntax that strikes a balance between simplicity and alignment with their everyday practices (*e.g.,* using markdown notation for note-taking) [56].

C4. Empower users with an automatic interpreter for transforming diverse expressions into data videos. Upon users' flexible expression of diverse authoring intents [44], the integration of an automatic interpreter is imperative to comprehend the narration context, analyze embedded commands, and subsequently translate them into data video component representations.

C5. Support real-time preview and iterative fine-tuning. Fragmentation across multiple tools in data video creation hampers real-time preview and adjustment [65]. It is necessary to integrate all components in a unified platform, thereby addressing these issues effectively.

4.2 Overview

The workflow of Data Playwright is shown in Fig. 1. It embodies a new data video authoring paradigm that allows users to write annotated narrations based on uploaded visualizations (Fig. 1-a), organically combining narrative and authoring intents (**C1**). Our formative study found participants generally showed interest for this new paradigm. According to the findings of user preference (Sec. 3.2.1), humans prefer leading creativity-intensive tasks such as text narration crafting, while AI systems can lead labor-intensive tasks like audio generation and video synthesis. For tasks requiring both creativity and labor, such as organizing visual elements, creating animations, and aligning timelines, the collaboration between humans and AI through annotated narration is recommended (**C2**). To facilitate these mixed-initiative tasks, we devise an annotation syntax (Tab. 2) with a minimal learning curve, based on the observations (Sec. 3.2.2) in our formative study (**C3**). We further develop a JSON-formatted data video specification (Fig. 3) to enhance human-AI collaboration. The automatic interpreter in Data Playwright plays a crucial role in translating users' diverse inputs, containing various observed design practices (Sec. 3.2.2), into the data video specification. (**C4**). As depicted in Fig. 1-b, textual and visual features are extracted from users' inputs and utilized to synthesize data video components, *i.e.,* audio narration, visual animations, and their temporal relations. The synthesized components are finally rendered into a data video (Fig. 1-c). An example video, derived from a real-world story [1], is shown in the bottom portion of Fig. 1, with the keyframe index numbers corresponding to the related NL command index numbers (likewise in subsequent cases). Additionally, Data Playwright supports preview and iterative adjustments of the system-generated prototype (**C5**). To accommodate users with varying skill levels, Data Playwright provides multimodal editors that combine natural language, interactive widgets, and a code panel (Fig. 1-d).

4.3 Annotated Narration Writing

To enable friendly intent expression, we introduce an innovative NL-based paradigm with *annotated narration*, allowing users to freely and seamlessly merge their data storytelling and video authoring intents.

4.3.1 Syntax

To ensure ease and accuracy of parsing, it is essential to establish a syntax for writing annotated narration. Inspired by the formative study and existing research [56], our primary principle is to replicate the familiar experience of taking notes using tools like Markdown in users' daily work, enabling users to express their data video authoring intents freely, with minimal learning curves. Based on the observations in the formative study, we developed the initial syntax version,

Table 2: Annotated narration syntax, where *Seg* is text narration segment, {NL} denotes inserted NL commands, and [] indicates specified temporal relations between narrations and animations.

Syntax	Explanation
Seg{NL}Seg	The animation specified by the {NL command} will be triggered when the audio reaches the corresponding narration segment.
{NL}[Seg]	The animation specified by the {NL command} has a matching duration with the corresponding audio of the [narration segment].
{NL}[Seg][Seg][Seg]	The {NL command} corresponds to each of the following parallel [narration segment].
{NL}[Seg{NL}[Seg]]	Nested specification of animations and timings.

with consideration of users' annotation habits and adaptability with the subsequent parsing model. As we progressed through the design process, we continuously refined the syntax based on feedback from domain experts and early test users. The resulting syntax, as displayed in Table 2, empowers users to articulate their authoring intents using NL while effectively indicating the temporal and semantic connections between animations and narration segments. Furthermore, the syntax accommodates parallel content and nested structures, enhancing the flexibility and expressiveness of the authoring experience.

4.3.2 Example Walkthrough

Let's explore the process of writing annotated narration with an example in Fig. 2. Meet Jessie, a data analyst who has visualized the Iris dataset using a parallel coordinates plot [4]. Jessie wants to create a data video to explain the dataset and begins scripting annotated narration.

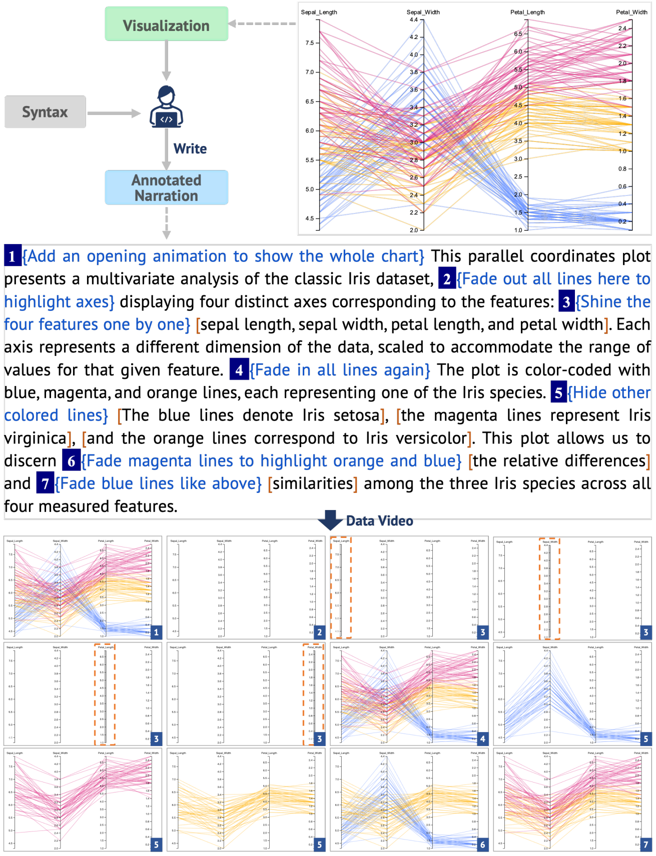
Initially, Jessie wants to showcase the entire visualization and adds an opening animation to capture the audience's attention. Since she doesn't have a clear idea yet, she directly inserts a command with curly brackets at the beginning: "*Add an opening animation to show the whole chart*" (Fig. 2-1). The text narration serves as the timeline for the data video, with animation triggered at the timestamps indicated by the placement of NL commands.

When Jessie mentions "*four distinct axes*", she wants all the line marks to fade out to highlight the axes. Therefore, she inserts the corresponding NL command (Fig. 2-2) before the word "*displaying*", indicating that all the lines will disappear when the audio reaches the word "*displaying*". Next, Jessie wants the four features to flash individually when mentioned in the audio. She inserts the NL command (Fig. 2-3) and encloses "*sepal length, sepal width, petal length, and petal width*" in square brackets to specify the start time and duration of the animation, aligned with the beginning and ending of this audio narration segment. After introducing the axes, Jessie brings back all the lines (Fig. 2-4). Later, Jessie writes a paragraph with parallel segments to explain the meanings of the three colors of lines separately. Jessie desires similar animation effects for the three segments, with only the target visual elements differing. Hence, she encloses the three parallel segments in square brackets and inserts the command "*Hide other colored lines*" at the beginning (Fig. 2-5), indicating that when the audio reaches a specific parallel segment, all lines except the ones mentioned in that segment will be hidden. Finally, Jessie describes that this plot allows discerning the relative differences and similarities among the three Iris species. When Jessie mentions "*differences*" and "*similarities*", she intends to showcase some corresponding data examples. Therefore, she writes NL commands to fade one color of lines to highlight the relationship between the other two (Fig. 2-6 and Fig. 2-7).

Jessie proceeds to input the written annotated narration into the system and obtains the corresponding results in the bottom portion of Fig. 2. She then previews and refines the video, expressing great satisfaction with the seamless NL-based experience.

4.4 Data Video Specification

To bridge human expressions and AI systems, it is essential to have a user-friendly and model-friendly specification for video representation [30]. Inspired by existing visualization-specific languages [35], we design a declarative specification for data videos that incorporates hierarchical structures. As shown in Fig. 3, users provide the static



1{Add an opening animation to show the whole chart} This parallel coordinates plot presents a multivariate analysis of the classic Iris dataset, 2{Fade out all lines to highlight axes} displaying four distinct axes corresponding to the features: 3{Shine the four features one by one} [sepal length, sepal width, petal length, and petal width]. Each axis represents a different dimension of the data, scaled to accommodate the range of values for that given feature. 4{Fade in all lines again} The plot is color-coded with blue, magenta, and orange lines, each representing one of the Iris species. 5{Hide other colored lines} [The blue lines denote Iris setosa], [the magenta lines represent Iris virginica], [and the orange lines correspond to Iris versicolor]. This plot allows us to discern 6{Fade magenta lines to highlight orange and blue} [the relative differences] and 7{Fade blue lines like above} [similarities] among the three Iris species across all four measured features.

Fig. 2: An annotated narration example. Users can incorporate authoring commands ({enclosed in blue curly brackets}) while crafting their text narration. They can also specify the desired duration of animations with orange square brackets ([ ]). The bottom portion shows the output video.

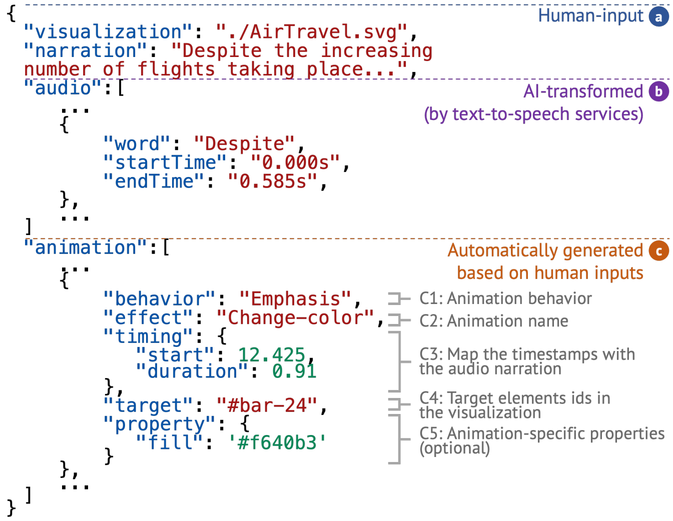
visualization and *narration* text (a). Building upon prior research, we enhance the visualizations in SVG format by incorporating encoded data information [16, 49] and the visualization structure [31, 54]. The text narration is further automatically transformed into *audio* (b), with each word assigned a specific timestamp using text-to-speech services. This audio serves as the timeline for the data video. The animation sequence specification is automatically generated based on human inputs (c). It comprises a series of animation units, each containing an animation effect, behavior, start time and duration aligned with the audio, target visual elements, and optional properties.

4.5 Automatic Annotated Narration Interpretation

The automatic interpreter is designed to understand users' diverse inputs and translate them into data video specifications (Sec. 4.4), with a focus on animation units. The advent of LLMs with advanced NL understanding and zero-shot learning capabilities has opened up new opportunities for tackling this challenging task [72]. The interpreter (Fig. 4) decouples annotation narrations and visualizations into distinct features, then uses them to synthesize data video components. Text narrations are transformed into audio, while temporal relationships are initially established by parsing notation annotations within the text narration. The LLM is leveraged to infer target visual elements, animation effects, and properties based on the extracted textual and visual features, thereby completing the animation unit specification (Fig. 3-c). Finally, the synthesized audio narration, visual animations, and their temporal relationships are rendered into a data video. The specific steps are as follows.

4.5.1 Feature Extraction

We first extract and organize input features, following previous research [25, 49, 54]. These features come from two sources, *i.e.,* visual-



```
{
  "visualization": "./AirTravel.svg",
  "narration": "Despite the increasing
  number of flights taking place...",
  "audio":[
     ...
     {
       "word": "Despite",
       "startTime": "0.000s",
       "endTime": "0.585s",
     },
     ...
  ]
  "animation":[
     ...
     {
       "behavior": "Emphasis",
       "effect": "Change-color",
       "timing": {
          "start": 12.425,
          "duration": 0.91
       },
       "target": "#bar-24",
       "property": {
          "fill": '#f640b3'
       }
     },
     ...
  ]
}
```

Human-input a
AI-transformed b
(by text-to-speech services)
Automatically generated c
based on human inputs
C1: Animation behavior
C2: Animation name
C3: Map the timestamps with the audio narration
C4: Target elements ids in the visualization
C5: Animation-specific properties (optional)

Fig. 3: A data video specification depicting the example in Fig. 1.

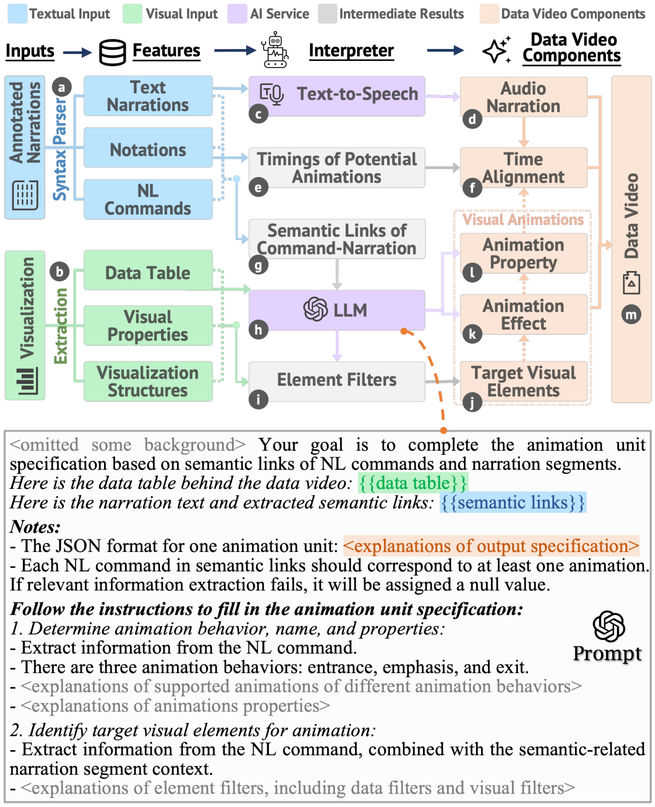
ization and annotated narration. Annotated narrations are decoupled by a syntax parser, resulting in text narrations, NL commands, and notations that specify the timings and semantic relationships between narration segments and NL commands (Fig. 4-a). As for visualizations (Fig. 4-b), each SVG element contains encoded data information, visual properties (*e.g.,* color, shape, and position) extracted from SVG tags and attributes, as well as the inferred visualization structure (*e.g.,* marks, axes, legends, labels, etc.).

4.5.2 Time Alignment

The next step is to construct the timeline of the data video to organize potential animations. As illustrated in Fig. 4-c, the text narration within users' annotated narration is automatically transformed into audio (Fig. 4-d) using text-to-speech services, with timestamps assigned to each word (Fig. 3-b). Thus, the text narration also serves as a timeline that ensures synchronization between the visual and auditory elements [12, 49, 65]. The subsequent task is to determine the trigger timestamp and duration of potential animations on the timeline. The notations in annotated narration syntax (Tab. 2) specify the temporal relationships between narration segments (with timestamps in the audio) and potential animations (determined by NL commands), as shown in Fig. 4-e and Fig. 3-C3, establishing the initial time alignment of the entire data video (Fig. 4-f). Specifically, the inserted positions of NL commands indicate the trigger timestamps for users' desired animations. For the animation duration, if the user has accurately defined the correspondence (using notations "[ ]") between specific animations and a narration segment, the duration of that segment in the audio will align with the animation duration. In cases where such correspondence is not explicitly defined, a default duration will be set based on the inferred animation effects. Following this step, we obtain the data video timeline, which includes anchor points that signify the desired timestamps for animation insertion, as well as the links of NL commands and narration segments that are semantically associated with these animations (Fig. 4-g). The subsequent tasks are to specify target visual elements, animation effects, and properties of the animations based on the semantic links.

4.5.3 Target Visual Elements Extraction

With features of each visual element at hand, we consider the entire collection of elements in the visualization as a visual "database". Our objective in this task is to filter and retrieve users' target visual elements, which are determined by both NL commands and semantic-related narration segments (Fig. 4-g), as presented in the formative study. Although LLMs excel at NL understanding, feeding the complete SVG specification into a language model would require a substantial number of tokens and a high time cost, and they may struggle with accurately comprehending visualizations with intricate SVG structures. To address these challenges, inspired by prior research on task abstraction for language models [20, 43, 49] and SVG abstraction techniques [31, 54], we propose *element filters* to bridge LLMs and SVGs, incorporating data filters and visual filters based on the formative study's findings.



<omitted some background> Your goal is to complete the animation unit specification based on semantic links of NL commands and narration segments.
Here is the data table behind the data video: {{data table}}
Here is the narration text and extracted semantic links: {{semantic links}}
Notes:
- The JSON format for one animation unit: <explanations of output specification>
- Each NL command in semantic links should correspond to at least one animation. If relevant information extraction fails, it will be assigned a null value.
Follow the instructions to fill in the animation unit specification:
1. Determine animation behavior, name, and properties:
- Extract information from the NL command.
- There are three animation behaviors: entrance, emphasis, and exit.
- <explanations of supported animations of different animation behaviors>
- <explanations of animations properties>
2. Identify target visual elements for animation:
- Extract information from the NL command, combined with the semantic-related narration segment context.
- <explanations of element filters, including data filters and visual filters>

Fig. 4: Automatic interpreter to synthesize data video components from users' annotated narration and visualizations.

As illustrated in Tab. 3, data filters consist of predicates such as "equal" and "range", where "range" includes bounded intervals (*e.g.,* "from 2010 to 2020") and unbounded intervals (*e.g.,* "larger than" and "less than"). Visual filters encompass features like shape, color, and position (determined by direction and order). The GPT-4 model [2] (Fig. 4-h) is then utilized to map the user's NL command to *element filters* (Fig. 4-i) based on text narration context (Fig. 4-g). These filtering conditions are further intersected to extract the final target visual elements (Fig. 4-j and Fig. 3-C4) based on visual features (Fig. 4-b). In addition, we refine the extracted visual elements by considering their grouping relations and removing unnecessary animated elements. The prompt structure is shown in Fig. 4, and for a complete reference, please refer to the supplementary material.

The supported visualization richness mainly depends on the shape and structure information in the visual filters. The currently supported visualization types include common ones like line, pie, bar, scatter, area, boxplot, tick, radar, etc. For elements lacking explicit visualization structure semantics (particularly in infographics), users can specify them based on the element shape and text semantics (if available). In principle, the approach can handle any SVG file. But to ensure accurate parsing, we will need to further expand the extracted visual features.

Compared to traditional command-based natural language interfaces, our approach fully integrates narration context and users' authoring commands. With LLMs, text narrations provide contextual information for the entire story, while NL commands act as the "eyes" of LLMs to perceive visualizations. This approach can alleviate ambiguity and underspecification issues in extracting target visual elements.

4.5.4 Animation Effect and Property Inference

In the formative study, we observed that users' NL commands frequently imply animation behaviors (entrance, emphasis, and exit), ef-

Table 3: Definition of *elements filters*, which bridge LLMs and visualization for target visual elements extraction.

Type	Feature	Example Utterance	Parsing Results
Data Filters	Equal	"wipe in the USA line"	{"column": "Country", "values": "USA"}
	Range	"shine 2005-2010 bars"	{"column": "Year", "values": [2005, 2010]}
Visual Filters	Color	"hide the blue lines"	{"color": "blue"}
	Shape	"hide the blue lines"	{"shape": "line"}
	Position	"highlight the bar 2rd from the left"	{"direction": "left", "order": 2}

fects, and properties. To meet users' animation needs, we utilize an animation library [3] to develop a set of commonly used animation presets as a proof-of-concept. For example, entrance animations include fade-in, float-in, fly-in, grow-in, etc., while emphasis animations encompass effects like change-color, shine, bounce, etc. Exit animations comprise actions such as zoom out, fade out, and hide.

The task of inferring animation effects and properties operates concurrently with the extraction of target visual elements within the same prompt (Fig. 4). Users' utterances vary in expression and specificity. Typically, animation behaviors (Fig. 3-C1) can be accurately extracted, while animation effects (Fig. 3-C2 and Fig. 4-k) are more ambiguous and may involve property information (Fig. 3-C5 and Fig. 4-i, *e.g.,* color, staggering, direction, etc.). For instance, user utterances like "*show the bars*", "*grow in the bars*", and "*grow in the bars from the bottom*" all convey the user's intention for the bars to appear. To address this, we go beyond LLM-based parsing and incorporate heuristics to populate the animation unit specification when relevant information cannot be inferred. On one hand, each animation has predefined properties and values, and on the other hand, we leverage common practices to establish default animation preferences for different animation behaviors (*e.g.,* "fade in" for entrance, "keep one and fade others" for emphasis, and "fade out" for exit) and for different visual elements (*e.g.,* "grow" for bars and "wipe" for lines). In addition, we have also predefined specific animation combinations based on visualization structures [49, 65]. This strategy serves to address the vague animation needs spanning the entire visualization, such as opening animations.

4.6 Interface

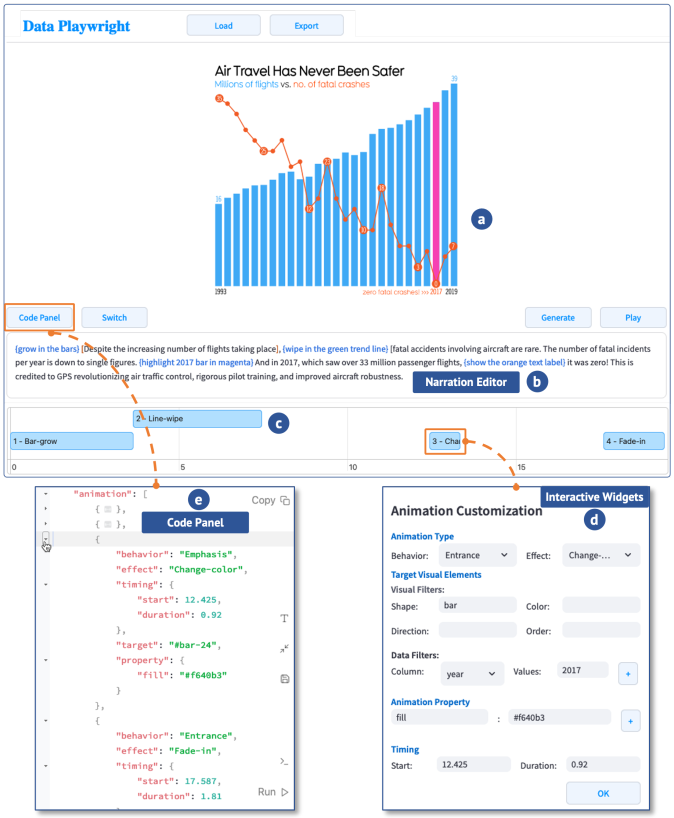
The user interface of Data Playwright is shown in Fig. 5. Users begin by uploading their visualization on the visual canvas (a) and proceed to write annotated narration in the narration editor (b). NL commands and notations are visually highlighted in distinct colors to differentiate them from the text narration. By clicking the "Switch View" button, the narration annotations will be hidden, allowing for a focused view solely on the text narration. Upon clicking the "Generate" button, the interpreter is invoked to automatically transform the input into a data video, with animation units displayed on the timeline (c). The "Play" button allows users to preview the data video directly on the canvas. Users have three main ways to make adjustments within the interface. They can directly edit the annotated narration in the editor (b) and rerun the process, utilize the interactive widgets (d) triggered by clicking on the timeline animation units (c) to modify attributes, or edit the data video specification directly in the "Code Panel" (e). This user-friendly interface caters to individuals with varying skill levels.

5 User Study

To evaluate the effectiveness and expressiveness of Data Playwright, we conducted a user study and created an example gallery based on participants' output during the study.

5.1 Study Design

Participants. We recruited 10 participants (P1-P10; 6 females and 4 males; aged 20-35) from various fields, including data analysts (P1, P4), science researchers (P2), software engineers (P3), AI researchers (P5), graphic designers (P6, P10), HCI/VIS researchers (P7, P9), and journalist (P8). Despite their diverse backgrounds, all participants



{grow in the bars} [Despite the increasing number of flights taking place], {wipe in the green trend line} [fatal accidents involving aircraft are rare. The number of fatal incidents per year is down to single figures. {highlight 2017 bar in magenta} And in 2017, which saw over 33 million passenger flights, {show the orange text label} it was zero! This is credited to GPS revolutionizing air traffic control, rigorous pilot training, and improved aircraft robustness.

```
"animation": [
    { ▭ },
    { ▭ },
    {
        "behavior": "Emphasis",
        "effect": "Change-color",
        "timing": {
            "start": 12.425,
            "duration": 0.92
        },
        "target": "#bar-24",
        "property": {
            "fill": "#f640b3"
        }
    },
    {
        "behavior": "Entrance",
        "effect": "Fade-in",
        "timing": {
            "start": 17.587,
            "duration": 1.81
```

Animation Customization

Animation Type

Behavior: Entrance Effect: Change-...

Target Visual Elements

Visual Filters:

Shape: bar Color:

Direction: Order:

Data Filters:

Column: year Values: 2017

Animation Property

fill : #f640b3

Timing

Start: 12.425 Duration: 0.92

Fig. 5: Data Playwright Interface: Users can preview and fine-tune the data video using NL (b), interactive widgets (d), and the code panel (e).

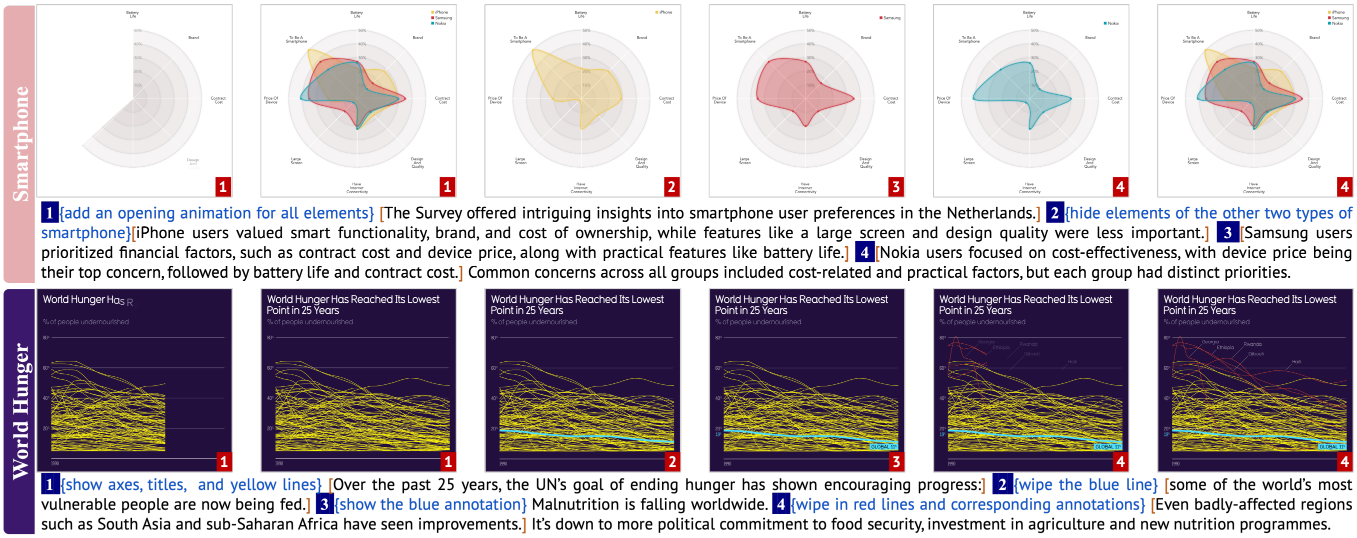
showed interest in watching data videos and expressed a desire to conveniently create their own. They all have basic knowledge of data visualization and animations on slides, with some participants having prior experience in using video editing tools (*e.g.,* iMovie and Adobe Premiere), with self-reported familiarity with data visualization (M = 3.50, SD = 1.18, 5 = Expert), animation (M = 3.00, SD = 1.33), and video editing (M = 2.80, SD = 1.4). They received a $30 gift card for their time (about 75 mins).

Procedure. The study began by collecting demographic information and introducing the study's goals and procedures. Participants were then guided through the following steps with a think-aloud approach. First, they received a tutorial on using Data Playwright and the supported animation effects and were familiarized with the easy syntax of *annotated narration*. They were encouraged to explore the system through a warm-up exercise with an example. Subsequently, participants were tasked with reproducing the air travel story (Fig. 1) to ensure that they had mastered the system. The visualization and text narrations were pre-loaded. Next, participants were given nine sets of visual designs from real-world storytelling practices, along with corresponding narrative hints (*e.g.,* topic and key insights). Their task was to select two sets they were familiar with or interested in and create their own data videos. They were expected to write annotated narration based on the visualizations to articulate their desired outcomes. The provided narrative hints served as references, and participants were free to explore the internet for relevant information or writing assistance. They were encouraged to align the text narration with their storytelling intentions and make necessary edits after the initial generation, ensuring the final output quality reflected their design skills. Finally, participants completed a questionnaire and engaged in a semi-structured interview.

5.2 Example Gallery

We collected a corpus of 20 videos created by participants, which also served as our example gallery[1]. Partial examples are shown in

[1] https://datavideos.github.io/Data-Playwright/



1{add an opening animation for all elements} [The Survey offered intriguing insights into smartphone user preferences in the Netherlands.] 2{hide elements of the other two types of smartphone}[iPhone users valued smart functionality, brand, and cost of ownership, while features like a large screen and design quality were less important.] 3[Samsung users prioritized financial factors, such as contract cost and device price, along with practical features like battery life.] 4[Nokia users focused on cost-effectiveness, with device price being their top concern, followed by battery life and contract cost.] Common concerns across all groups included cost-related and practical factors, but each group had distinct priorities.

1{show axes, titles, and yellow lines} [Over the past 25 years, the UN's goal of ending hunger has shown encouraging progress:] 2{wipe the blue line} [some of the world's most vulnerable people are now being fed.] 3{show the blue annotation} Malnutrition is falling worldwide. 4{wipe in red lines and corresponding annotations} [Even badly-affected regions such as South Asia and sub-Saharan Africa have seen improvements.] It's down to more political commitment to food security, investment in agriculture and new nutrition programmes.

Fig. 6: Data video examples created in the user study from real-world storytelling practices.

Fig. 1, Fig. 2, and Fig. 6 (the complete gallery can be found in the supplementary materials). The static materials used in the gallery were sourced from real-world storytelling practices. The gallery encompasses a wide range of topics (*e.g.,* social media usage, smartphone preference, COVID-19, world hunger, Thames state, tourism, GDP, etc.) and includes various visualization types (bar, line, radar, parallel coordinates, pie, etc.). Each video was accompanied by annotated narration (both narration text and NL commands). On average, these annotated narration consisted of 80.50 words (word count determined by spaces), ranging from 45 to 113.

5.3 Quantitative Results

Questionnaire. All participants completed the tasks. Fig. 7 illustrates the user ratings obtained from the questionnaire, utilizing a 5-point Likert scale where 5 represents the most positive response. Overall, participants were satisfied with the system (M = 4.20, SD = 0.42) and generally expressed their interest in utilizing the system in the future. The user experience was highly rated in terms of friendliness (M = 4.60, SD = 0.52) and enjoyment (M = 4.60, SD = 0.52). Furthermore, participants generally agreed that the system was easy to learn (M = 4.60, SD = 0.52) and easy to use (M = 4.40, SD = 0.52). They also perceived the system as effective in meeting their needs and expectations (M = 4.10, SD = 0.57). The quality of the videos consistently received positive ratings (M = 4.20, SD = 0.63). However, participants rated the system relatively low on powerfulness, suggesting opportunities to integrate more powerful functionalities and interactions, or to explore merging our paradigm with other tools.

Written annotated narrations analysis. In order to assess the complexity of users' written annotated narrations, we analyzed the number of NL commands in each annotated narration as well as the length of the utterances. The maximum number of NL commands observed in a single annotated narration was 17, while the minimum recorded was 2. On average, users only needed approximately 5.73 NL commands to create a data video. In the example gallery, a total of 126 NL commands were collected from participants. The shortest command consisted of only one word (*e.g.,* "same"), while the longest spanned 11 words (*e.g.,* "grow in 1990-today annotations and change the river line to blue"). The average length of utterances was 4.84 words. The results indicated that users can create data videos without the need for extensive command writing, and individual commands do not have to be overly complex.

5.4 Qualitative Results

Feedback of the authoring experience with annotated narration. The participants generally appreciated this new data video creation paradigm by writing annotated narration. They were able to focus more on storytelling, using casual language expressions instead of complex and time-consuming tool-specific operations. P8 remarked, "*This approach is amazing! Combining narration and commands makes the whole process simpler and smoother. It should be integrated into mainstream video tools.*" Users often input casual utterances that may contain errors. However, thanks to the fault tolerance of LLMs, the interpreter can still effectively parse these inputs to the best of its ability. P3 mentioned, "*I can freely input natural language expressions, and the system can intelligently understand my intentions, even if there are grammar or spelling errors.*" For straightforward authoring tasks, users can easily accomplish them using simple NL commands and notations. And more complex or fine-tuning tasks can be efficiently iterated through the process of writing annotated narration. P9 also noted that the syntax for parallel content "*is very interesting and useful*", as demonstrated in the smartphone example (Fig. 6). Moreover, 82/126 NL commands were accompanied by the "[ ]" notations, indicating a user's desire for precise time control. P10 said, "*Using notations to specify time relationship is very easy and helpful, avoiding the troublesome alignment work.*" However, as mentioned by P5, while the system's syntax highlighting helps users comprehend the annotated narration, excessive commands can increase complexity and potentially hinder the intuitive observation of the narrative structure. Additionally, P3 hoped to fully integrate existing syntax (e.g. Markdown) to enrich the current annotation syntax.

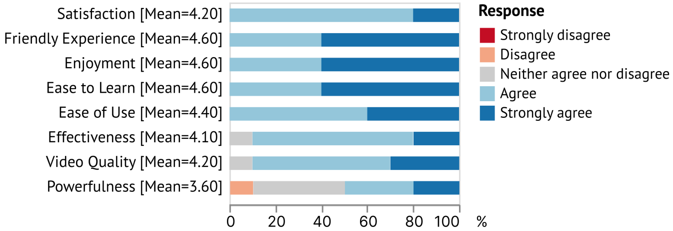


Fig. 7: User ratings of Data Playwright with a 5-point Likert scale.

Feedback of usability of Data Playwright. Participants found the system easy to learn and use, including the interaction and syntax of annotated narration, with almost no learning curve. Even those with limited experience in visualization and video authoring expressed their satisfaction. P3 remarked, "*The pure NL interaction is very convenient, especially for beginners like me. When I wanted to create data videos for my presentations before, I felt that the process required a lot of time and effort, so I would usually give up.*" Designer P6 stated, "*Compared to constantly navigating and switching between different tools on traditional interfaces, I prefer the natural language interaction entirely within a single text box.*" However, participants (P7, P9) also reported inconvenience in performing fine-grained selection of target elements during post-generation editing. They desired the ability to directly click (with a mouse) or sketch (with a pen) on the visualization to select target elements, rather than limited to NL, widgets, and code.

Feedback of data video quality. Most participants were highly satisfied with the final generated data videos. They recognized the intuitiveness and engaging nature of data videos by combining animated visuals and narration audio, and appreciated that this can significantly enhance their presentation engagement. P4 stated, "*I can't believe that I can create such videos with little experience.*" P2 said, "*This is crazy useful for enhancing my boring presentations.*" Some participants (P7, P9, P10) viewed this approach as a means to quickly prototype videos or even directly use them as their final output, particularly when working with visualization-related content. However, the limited diversity of the animation library, as well as the lack of support for multi-chart visualizations (and their transitions), hampers its ability to meet a wider range of design requirements.

Analysis of failure cases. Out of the 126 NL commands collected, 35/126 underwent further modifications by users. Among these modifications, 12/35 were made due to the system's failure in accurately interpreting user intentions, while 23/35 were adjustments made by users through NL interactions after previewing the results (*e.g.,* changing the animation effect from "fade in" to "fly in"). We further analyzed the failure cases and identified the following reasons: (1) Participants expected to navigate back to a previous moment (*e.g.,* "*back to normal*"), but the system currently lacks support for frame rollback; (2) The desired animation effects requested by users exceeded the system's animation library scope (*e.g.,* "*shake the bar*"); (3) Participants used vague descriptive adjectives (*e.g.,* "*magically show elements*"), causing the system to recommend a default animation that did not always produce the desired effects; (4) Users' intended commands contradicted the visual facts. For example, in the smartphone example (Fig. 6), a user wrote "*fade in the yellow radar*" to showcase the iPhone's mark, but the radar's outer ring had an additional linear border, causing the yellow border not to be identified as the target element. These failure cases highlight the need to enhance the annotated narration grammar, expand the animation library, strengthen the robustness of fuzzy semantic understanding, and increase the recommendation effectiveness.

6 Discussion

In this section, we discuss how to design more powerful annotated narration, build mixed-initiative interfaces for human-AI collaboration, and support more application scenarios, as well as limitations.

Design more powerful annotated narration. While annotated narration can effectively help create data videos, our current version only supports explicit authoring commands, lacking creative ideas. Participants in the user study also expressed the desire for a more powerful annotated narration. Addressing these issues necessitates enhancing both the syntax and video representation from two main aspects: functionality and design. In terms of functionality, our current focus is single visual design, there is potential to develop block-based multi-scene stories [8, 12]. Furthermore, reusability can be improved by supporting animation indexing, caching, retrieval, and enabling specific keyframe rollback. Regarding design, future work could explore finer-grained features like emotional audio and animations [69], element transitions [58], narrative structures [71], and cinematic effects [70]. Additionally, the current approach integrates inline annotations with the narration text. An alternative approach suggested in the formative study is directly highlighting and annotating existing text, similar to annotating PDF files. This presents a potential variation of annotated narration but also introduces challenges in handling unstructured data. In the user study, P9 also suggested adding an intermediate layer between the annotated narration and the data video. This layer would visualize the script's structure or video timeline, such as using sentence-level blocks or visually annotated narration with semantic icons. Beyond storytelling, annotated narration can be enhanced to support analysis tasks like data retrieval [31, 33], insight extraction [48], and visualization [39].

Build mixed-initiative interfaces for human-AI collaboration. This paper introduces a new NL-based data video creation paradigm that offers a fresh and user-friendly experience. The participants in the user study highly appreciated the experience but also expressed a demand for more intelligent multimodal interactions [28], particularly in assisting with writing annotated narration and performing post-generation editing. Here, we discuss some potential multimodal combinations. Firstly, instead of writing complete narration text, integrating AI-powered writing features can enhance efficiency and creativity [29]. Users can outline the narrative structure and provide NL commands, transforming the process of writing annotated narration into writing concise code [56]. AI can then generate high-quality narration text prototypes. Moreover, to improve system discoverability and users' writing efficiency and accuracy, command recommendations can be provided during users' writing process. A semantic-aware format painter could be an intriguing design to enhance reusability. When fine-tuning videos, optimizing the combination of multiple modalities (*e.g.,* pen, speech, mouse, and typing) is crucial, especially with support for interactive editing on the video [62]. This enables more intuitive selection of target visual elements and the ability to input NL commands for subsequent modifications. In addition to NL, we also anticipate the integration of more intent expression methods, such as sketching [32], examples [69], or learning from users' interaction history [23].

Support more application scenarios. We envision that this paradigm can empower everyone to be the playwright of their data and support more scenarios. Firstly, in terms of application, the annotated narration and its interpreter can be packaged into an independent suite for data video creation. It can be further integrated as plugins into various environments such as notebooks, browsers, PowerPoint, Figma, and various video tools, enabling the transformation of static materials into dynamic presentations. It can also be extended to support application-specific videos, such as educational tutorial videos and museum exhibition videos [38]. Secondly, regarding user experience, Data Playwright enables users to express their narrative and authoring intents exclusively through NL. Users can write anytime and anywhere, even in a simple notepad. We also observed two patterns in writing annotated narration: some participants simultaneously wrote narration text while inserting NL commands, while others first drafted the narration text and then inserted NL commands. These patterns reflect different user preferences and application scenarios. Moreover, this is an AI-resilient alternative experience [17]. Thirdly, in terms of representation and communication, annotated narration can serve as a shared medium that can be understood by novices, designers, and AI systems. It facilitates effective communication among these stakeholders and serves as a universal prompt method for AI systems.

Limitations and Future Work. Our evaluation primarily relied on a user study, though the participant pool was limited in size despite their diverse backgrounds. To broaden and deepen the evaluation, future work could recruit more participants for targeted tasks using their own materials. Expert interviews and reflections with authors of existing relevant tools could also provide a deeper understanding of this work's value and limitations, and insights on the current state and future directions of this field. In addition, this work focuses more on improving learnability rather than expressiveness [41]. There remains a gap between the generated data videos and the popular videos on mainstream platforms (*e.g.,* YouTube, TikTok). However, the approach can be extended to support more complex video creation with additional engineering and technical innovations discussed above.

7 Conclusion

To streamline data video creation, we propose a novel natural language-based paradigm that seamlessly merges users' narrative and authoring intents into a unified format called annotated narration. We also develop a prototype system, Data Playwright, allowing users to write annotated narration to articulate their desired outcomes and directly obtain the final data video with the automatic interpreter. Users can also preview and fine-tune the video. The user study indicated that participants highly appreciated this new NL-based paradigm and can effectively create data videos that satisfied their intents. We hope that this work can empower everyone to be the playwright of their data and inspire more future research that democratizes engaging story creation.

Acknowledgments

The authors wish to thank Leni Yang, Liwenhan Xie, and Jiachen Wang for their valuable suggestions that have enhanced the paper's quality. Additionally, the authors express their gratitude to all the study participants and reviewers for their contributions and feedback.

References

[1] Air travel safer story. https://informationisbeautiful.net/beautifulnews/77-air-travel-safer/. 4

[2] Gpt-4 model. https://openai.com/gpt-4. 6

[3] Gsap animation platform. https://gsap.com/. 7

[4] Iris dataset. https://archive.ics.uci.edu/dataset/53/iris. 5

[5] F. Amini, N. Henry Riche, B. Lee, C. Hurter, and P. Irani. Understanding Data Videos: Looking at Narrative Visualization through the Cinematography Lens. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI'15*, pp. 1459–1468. ACM, 2015. 2

[6] F. Amini, N. H. Riche, B. Lee, J. Leboe-McGowan, and P. Irani. Hooked on data videos: assessing the effect of animation and pictographs on viewer engagement. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, AVI '18, pp. 1–9. ACM, 2018. 1, 2

[7] F. Amini, N. H. Riche, B. Lee, A. Monroy-Hernandez, and P. Irani. Authoring Data-Driven Videos with DataClips. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):501–510, 2017. 2

[8] Y. Cao, J. L. E, Z. Chen, and H. Xia. DataParticles: Block-based and Language-oriented Authoring of Animated Unit Visualizations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, CHI '23*, pp. 1–15. ACM, 2023. 1, 2, 3, 9

[9] Q. Chen, S. Cao, J. Wang, and N. Cao. How Does Automation Shape the Process of Narrative Visualization: A Survey of Tools. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–20, 2023. 2

[10] Z. Chen, Q. Yang, X. Xie, J. Beyer, H. Xia, Y. Wu, and H. Pfister. Sporthesia: Augmenting Sports Videos Using Natural Language. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):918 – 928, 2023. 2, 3

[11] H. Cheng, J. Wang, Y. Wang, B. Lee, H. Zhang, and D. Zhang. Investigating the role and interplay of narrations and animations in data videos. *Computer Graphics Forum*, 41(3):527–539, 2022. 1, 2, 3

[12] P. Chi, T. Dong, C. Frueh, B. Colonna, V. Kwatra, and I. Essa. Synthesis-Assisted Video Prototyping From a Document. In *The 35th Annual ACM Symposium on User Interface Software and Technology, UIST'22*, pp. 1–10. ACM, 2022. 6, 9

[13] J. M. Clark and A. Paivio. Dual coding theory and education. *Educational Psychology Review*, 3(3):149–210, 1991. 2

[14] W. Cui, X. Zhang, Y. Wang, H. Huang, B. Chen, L. Fang, H. Zhang, J. G. Lou, and D. Zhang. Text-to-Viz: Automatic Generation of Infographics from Proportion-Related Natural Language Statements. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):906–916, 2020. 2, 3

[15] T. Ge, B. Lee, and Y. Wang. CAST: Authoring Data-Driven Chart Animations. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, pp. 1–15. ACM, 2021. 2

[16] T. Ge, Y. Zhao, B. Lee, D. Ren, B. Chen, and Y. Wang. Canis: A High-Level Language for Data-Driven Chart Animations. *Computer Graphics Forum*, 39(3):607–617, 2020. 2, 5

[17] Z. Gu, I. Arawjo, K. Li, J. K. Kummerfeld, N. S. Wales, and E. L. Glassman. An AI-Resilient Text Rendering Technique for Reading and Skimming Documents. In *Proceedings of the CHI Conference on Human Factors in Computing Systems, CHI '24*, pp. 1–22. ACM, 2024. 9

[18] J. Heer. Agency plus automation: Designing artificial intelligence into interactive systems. *Proceedings of the National Academy of Sciences of the United States of America*, 116(6):1844–1850, 2019. 2, 4

[19] J. Heer and G. G. Robertson. Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1240–1247, 2007. 2

[20] T. S. Kim, D. Choi, Y. Choi, and J. Kim. Stylette: Styling the Web with Natural Language. In *Proceedings of CHI Conference on Human Factors in Computing Systems, CHI'22*, pp. 1–17. ACM, 2022. 2, 6

[21] Y. Kim and J. Heer. Gemini: A Grammar and Recommender System for Animated Transitions in Statistical Graphics. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):485–494, 2021. 2

[22] Y. Kim and J. Heer. Gemini2: Generating Keyframe-Oriented Animated Transitions Between Statistical Graphics. In *Proceedings of the 2021 IEEE Visualization Conference, VIS'21*, pp. 201–205, 2021. 2

[23] S. Lalle, D. Toker, and C. Conati. Gaze-Driven Adaptive Interventions for Magazine-Style Narrative Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):2941–2952, 2021. 3, 9

[24] X. Lan, Y. Shi, Y. Wu, X. Jiao, and N. Cao. Kineticharts: Augmenting affective expressiveness of charts in data stories with animation design. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):933–943, 2021. 2

[25] S. Latif, Z. Zhou, Y. Kim, F. Beck, and N. W. Kim. Kori: Interactive Synthesis of Text and Charts in Data Documents. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):184–194, 2022. 3, 5

[26] M. Leake, H. V. Shin, J. O. Kim, and M. Agrawala. Generating Audio-Visual Slideshows from Text Articles Using Word Concreteness. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, CHI'20*, pp. 1–11. ACM, 2020. 2

[27] B. Lee, R. H. Kazi, and G. Smith. SketchStory: Telling more engaging stories with data through freeform sketching. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2416–2425, 2013. 3

[28] B. Lee, A. Srinivasan, P. Isenberg, and J. Stasko. Post-wimp interaction for information visualization. *Foundations and Trends in Human-Computer Interaction*, 14(1):1–95, 2021. 1, 3, 9

[29] M. Lee, K. Ilonka Gero, J. Joon Young Chung, and et al. A Design Space for Intelligent and Interactive Writing Assistants. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, CHI '24*, pp. 1–33, 2024. 9

[30] H. Li, Y. Wang, and H. Qu. Where Are We So Far? Understanding Data Storytelling Tools from the Perspective of Human-AI Collaboration. In *Proceedings of CHI Conference on Human Factors in Computing Systems, CHI'24*, pp. 1–28, 2024. 4, 5

[31] H. Li, Y. Wang, A. Wu, H. Wei, and H. Qu. Structure-aware Visualization Retrieval. In *Proceedings of the CHI Conference on Human Factors in Computing Systems, CHI'22*, vol. 1, pp. 1–14. ACM, 2022. 5, 6, 9

[32] Y. Lin, H. Li, L. Yang, A. Wu, and H. Qu. InkSight: Leveraging Sketch Interaction for Documenting Chart Findings in Computational Notebooks. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):944 – 954, 2024. 3, 9

[33] Y. Luo, Y. Zhou, N. Tang, G. Li, C. Chai, and L. Shen. Learned Data-aware Image Representations of Line Charts for Similarity Search. In *Proceedings of the ACM on Management of Data, SIGMOD'23*, pp. 1–29. ACM, 2023. 9

[34] D. Masson, S. Malacria, G. Casiez, and D. Vogel. Charagraph: Interactive Generation of Charts for Realtime Annotation of Data-Rich Paragraphs. In *ACM Conference on Human Factors in Computing Systems, CHI'23*. ACM, 2023. 3

[35] A. M. McNutt. No Grammar to Rule Them All: A Survey of JSON-style DSLs for Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):160 – 170, 2023. 5

[36] A. M. McNutt and R. Chugh. Integrated Visualization Editing via Parameterized Declarative Templates. In *Proceedings of CHI Conference on Human Factors in Computing Systems, CHI'21*, pp. 1–14. ACM, 2021. 3

[37] H. O. Obie, C. Chua, I. Avazpour, M. Abdelrazek, J. Grundy, and T. Bednarz. A study of the effects of narration on comprehension and memorability of visualisations. *Journal of Computer Languages*, 52(April):113–124, 2019. 1, 2

[38] Y. Ouyang, L. Shen, Y. Wang, and Q. Li. NotePlayer: Engaging Jupyter Notebooks for Dynamic Presentation of Analytical Processes. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology, UIST'24*, pp. 1–15. ACM, 2024. 9

[39] A. Pandey, A. Srinivasan, and V. Setlur. MEDLEY: Intent-based Recommendations to Support Dashboard Composition. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1135–1145, 2023. 3, 9

[40] D. M. Russell and E. H. Chi. Looking Back: Retrospective Study Methods for HCI. In J. S. Olson and W. A. Kellogg, eds., *Ways of Knowing in HCI*, pp. 373–393. Springer, 2014. 3

[41] A. Satyanarayan, B. Lee, D. Ren, J. Heer, J. Stasko, J. Thompson, M. Brehmer, and Z. Liu. Critical Reflections on Visualization Authoring Systems. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1–11, 2019. 9

[42] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE transactions on visualization and computer graphics*, 16(6):1139–1148, 2010. 2

[43] L. Shen, H. Li, Y. Wang, and H. Qu. From Data to Story: Towards Automatic Animated Data Video Creation with LLM-based Multi-Agent Systems. *arXiv: 2408.03876*, pp. 1–8, 2024. 6

[44] L. Shen, E. Shen, Y. Luo, X. Yang, X. Hu, X. Zhang, Z. Tai, and J. Wang. Towards Natural Language Interfaces for Data Visualization: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 29(6):3121–3144, 2023. 1, 2, 4

[45] L. Shen, E. Shen, Z. Tai, Y. Song, and J. Wang. TaskVis: Task-oriented Visualization Recommendation. In *Proceedings of the 23th Eurographics Conference on Visualization (Short Papers), EuroVis'21*, pp. 91–95. Eurographics, 2021. 3

[46] L. Shen, E. Shen, Z. Tai, Y. Wang, Y. Luo, and J. Wang. GALVIS: Visualization Construction through Example-Powered Declarative Programming. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM'22*, pp. 4975–4979. ACM, 2022. 3

[47] L. Shen, E. Shen, Z. Tai, Y. Xu, J. Dong, and J. Wang. Visual Data Analysis with Task-Based Recommendations. *Data Science and Engineering*, 7(4):354–369, 2022. 3

[48] L. Shen, Z. Tai, E. Shen, and J. Wang. Graph Exploration With Embedding-Guided Layouts. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):3693–3708, 2024. 9

[49] L. Shen, Y. Zhang, H. Zhang, and Y. Wang. Data Player: Automatic Generation of Data Videos with Narration-Animation Interplay. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):109–119, 2024. 1, 2, 3, 4, 5, 6, 7

[50] C. Shi, W. Cui, C. Liu, C. Zheng, H. Zhang, Q. Luo, and X. Ma. NL2Color: Refining Color Palettes for Charts with Natural Language. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):814 – 824, 2024. 2

[51] D. Shi, F. Sun, X. Xu, X. Lan, D. Gotz, and N. Cao. AutoClips: An Automatic Approach to Video Generation from Data Facts. *Computer Graphics Forum*, 40(3):495–505, 2021. 1, 2, 4

[52] Y. Shi, X. Lan, J. Li, Z. Li, and N. Cao. Communicating with motion: A design space for animated visual narratives in data videos. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, CHI'21*, pp. 1–13, 2021. 2

[53] M. Shin, J. Kim, Y. Han, L. Xie, M. Whitelaw, B. C. Kwon, S. Ko, and N. Elmqvist. Roslingifier: Semi-Automated Storytelling for Animated Scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 29(6):2980–2995, 2023. 2

[54] L. S. Snyder and J. Heer. DIVI: Dynamically Interactive Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):403–413, 2024. 5, 6

[55] S. Srinivasa Ragavan, Z. Hou, Y. Wang, A. D. Gordon, H. Zhang, and D. Zhang. GridBook: Natural Language Formulas for the Spreadsheet Grid. In *Proceedings of the 27th International Conference on Intelligent User Interfaces, IUI'22*, pp. 345–368. ACM, 2022. 2

[56] S. Suh, J. Zhao, and E. Law. CodeToon: Story Ideation, Auto Comic Generation, and Structure Mapping for Code-Driven Storytelling. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology, UIST'22*, pp. 1–16. ACM, 2022. 4, 9

[57] N. Sultanum, F. Chevalier, Z. Bylinskii, and Z. Liu. 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*, CHI '21, pp. 1–17. ACM, 2021. 3

[58] J. Tang, L. Yu, T. Tang, X. Shu, L. Ying, Y. Zhou, P. Ren, and Y. Wu. Narrative transitions in data videos. In *IEEE Visualization Conference, VIS'20*, pp. 151–155. IEEE, 2020. 2, 9

[59] J. Thompson, Z. Liu, W. Li, and J. Stasko. Understanding the Design Space and Authoring Paradigms for Animated Data Graphics. *Computer Graphics Forum*, 39(3):207–218, 2020. 2

[60] J. R. Thompson, Z. Liu, and J. Stasko. Data Animator: Authoring Expressive Animated Data Graphics. In *Proceedings of CHI Conference on Human Factors in Computing Systems, CHI'21*, pp. 1–18. ACM, 2021. 2

[61] T. Tseng, R. Cheng, and J. Nichols. Keyframer: Empowering Animation Design using Large Language Models. *arXiv: 2402.06071*, pp. 1–31, 2024. 2

[62] B. Wang, Y. Li, Z. Lv, H. Xia, Y. Xu, and R. Sodhi. LAVE: LLM-Powered Agent Assistance and Language Augmentation for Video Editing. In *Proceedings of the 29th International Conference on Intelligent User Interfaces, IUI'24*, pp. 1–15, 2024. 9

[63] Y. Wang, Y. Gao, R. Huang, W. Cui, H. Zhang, and D. Zhang. Animated Presentation of Static Infographics with InfoMotion. *Computer Graphics Forum*, 40(3):507–518, 2021. 2

[64] Y. Wang, Z. Hou, L. Shen, T. Wu, J. Wang, H. Huang, H. Zhang, and D. Zhang. Towards Natural Language-Based Visualization Authoring. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1222 – 1232, 2023. 2

[65] Y. Wang, L. Shen, Z. You, X. Shu, B. Lee, J. Thompson, H. Zhang, and D. Zhang. WonderFlow: Narration-Centric Design of Animated Data Videos. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15, 2024. 2, 3, 4, 6, 7

[66] Y. Wu, L. Yan, L. Shen, Y. Wang, N. Tang, and Y. Luo. ChartInsights: Evaluating Multimodal Large Language Models for Low-Level Chart Question Answering. In *The Conference on Empirical Methods in Natural Language Processing (Findings), EMNLP'24*, pp. 1–9, 2024. 2

[67] H. Xia. Crosspower: Bridging graphics and linguistics. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, UIST'20*, pp. 722–734. ACM, 2020. 3

[68] H. Xia, J. Jacobs, and M. Agrawala. Crosscast: Adding Visuals to Audio Travel Podcasts. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, UIST'20*, pp. 735–746. ACM, 2020. 3

[69] L. Xie, Z. Zhou, K. Yu, Y. Wang, H. Qu, and S. Chen. Wakey-Wakey: Animate Text by Mimicking Characters in a GIF. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, UIST'23*, pp. 1–14. ACM, 2023. 3, 9

[70] X. Xu, A. Wu, L. Yang, Z. Wei, R. Huang, D. Yip, and H. Qu. Is It the End? Guidelines for Cinematic Endings in Data Videos. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, CHI'23*, pp. 1–16. ACM, 2023. 9

[71] L. Yang, X. Xu, X. Y. Lan, Z. Liu, S. Guo, Y. Shi, H. Qu, and N. Cao. A Design Space for Applying the Freytag's Pyramid Structure to Data Stories. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):922–932, 2022. 9

[72] W. Yang, M. Liu, Z. Wang, and S. Liu. Foundation Models Meet Visualizations: Challenges and Opportunities. *Computational Visual Media*, pp. 1–21, 2024. 3, 5

[73] L. Ying, Y. Wang, H. Li, S. Dou, H. Zhang, X. Jiang, H. Qu, and Y. Wu. Reviving Static Charts into Live Charts. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15, 2024. 2, 4

[74] Y. Yu, L. Shen, F. Long, H. Qu, and H. Chen. PyGWalker: On-the-fly Assistant for Exploratory Visual Data Analysis. In *Proceedings of IEEE Visualization and Visual Analytics, IEEE VIS'24*, pp. 1–5. IEEE, 2024. 3

[75] J. Zong, J. Pollock, D. Wootton, and A. Satyanarayan. Animated Vega-Lite: Unifying Animation with a Grammar of Interactive Graphics. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):149–159, 2023. 2