

Story Ribbons: Reimagining Storyline Visualizations with Large Language Models

Catherine Yeh, Tara Menon, Robin Singh Arya, Helen He, Moira Weigel, Fernanda Viégas, and Martin Wattenberg

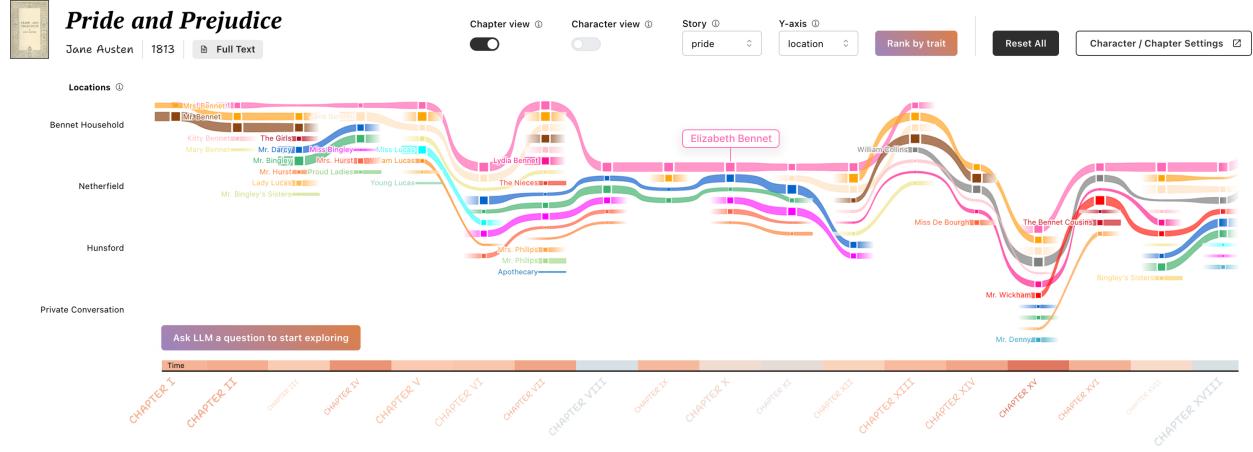


Fig. 1: STORY RIBBONS is an interactive narrative analysis tool that visualizes LLM-extracted insights about literary works. Here, a partial visualization of *Pride and Prejudice* by Jane Austen is shown. Each “ribbon” represents a different character (e.g., the top pink ribbon = Elizabeth Bennet), and can be used to track interactions across novel chapters (x-axis) and locations (y-axis). Chapter titles are colored by sentiment (red: positive, blue: negative). STORY RIBBONS enables users to explore stories at multiple narrative levels, and offers several features to customize visualizations to individual analysis workflows.

Abstract—Analyzing literature involves tracking interactions between characters, locations, and themes. Visualization has the potential to facilitate the mapping and analysis of these complex relationships, but capturing structured information from unstructured story data remains a challenge. As large language models (LLMs) continue to advance, we see an opportunity to use their text processing and analysis capabilities to augment and reimagine existing storyline visualization techniques. Toward this goal, we introduce an LLM-driven data parsing pipeline that automatically extracts relevant narrative information from novels and scripts. We then apply this pipeline to create STORY RIBBONS, an interactive visualization system that helps novice and expert literary analysts explore detailed character and theme trajectories at multiple narrative levels. Through pipeline evaluations and user studies with STORY RIBBONS on 36 literary works, we demonstrate the potential of LLMs to streamline narrative visualization creation and reveal new insights about familiar stories. We also describe current limitations of AI-based systems, and interaction motifs designed to address these issues.

Index Terms—Narrative visualization, interactive literary analysis, large language models

1 INTRODUCTION

Visualizing textual data is currently a major challenge. The key difficulty lies in extracting structured information from natural language. Typically, researchers use dedicated algorithms, ranging from counting words to make tag clouds [19, 26] to elaborate statistical methods such as topic modeling [6, 37]. Yet, even the most sophisticated, bespoke approaches often fail to capture important aspects of meaning.

Given recent successes of AI systems based on large language models (LLMs)¹, it is natural to ask whether their power and generality can help us build better text visualizations. Of course, LLMs are not a magic pixie dust that we can just sprinkle on existing visualizations for great

results. They bring their own challenges, producing output that can be unpredictable, mysterious, and even contain “hallucinations” [29, 31].

In this work, we explore how to harness the power of LLMs while addressing their limitations. Our focus is on visualizing stories, specifically through *storyline visualization* techniques. Storyline visualizations portray narrative timelines [21, 28, 53, 66] with the goal of helping people critically examine and interpret works of literature. For example, an ideal storyline visualization of *Pride and Prejudice* by Jane Austen might show Elizabeth’s evolving dynamic with Mr. Darcy, her sister Jane’s gentle romance with Mr. Bingley, and Lydia’s reckless elopement with Wickham – all unfolding across distinct settings from the grand estate of Pemberley to the regimented world of Longbourn.

The issue is how to convert the raw story text into concrete representations of a “gentle romance” or “reckless elopement.” Extracting relevant information to visualize how characters interact and how their relationships shift is no simple task [20, 32]. As such, preparing the input data for narrative visualizations – e.g., a film script or novel – often requires extensive time and manual effort [28]. This is especially challenging for novels, which do not contain metadata such as explicit scene divisions or character/location labels [20].

LLMs seem like a promising approach for extracting the data needed for storyline visualizations [20, 21, 24, 47, 71]. Our work investigates how LLMs can help build and extend traditional storyline visualizations

[•] Authors are with Harvard University. Viégas and Wattenberg are also with Google, but this work was done at Harvard. Emails: {catherineyeh, robinsingh_arya, fernanda, wattenberg}@g.harvard.edu, {taramenon, weigel}@fas.harvard.edu, helen_he@college.harvard.edu.

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.xxxxxx

¹Technically, for the AI systems we used, language modeling is just one step in their training process. However, for brevity, we refer to them as LLMs.

to aid literature analysis. Concretely, our research questions are:

- **RQ1:** How can we use LLMs to automate and extend the extraction of data for unstructured narratives?
- **RQ2:** What are the right forms of visualization and interaction to help people understand and calibrate trust in AI story insights?

To address these questions, we created an interactive system, STORY RIBBONS. Over six months, we co-designed the tool [5, 35] with three literary scholars, who became co-authors on this paper (**C1-3**) and have expertise in narrative theory, comparative literature, and literary criticism. STORY RIBBONS lets users explore stories at multiple narrative levels, visualizing locations, characters, and themes along a rich and customizable set of literary dimensions (Fig. 1). The system is based on an LLM-powered data processing pipeline, which is almost completely automated, for extracting detailed narrative information from stories.

We then evaluated our system in multiple ways. First, we quantitatively and qualitatively assessed the data pipeline performance on 36 stories. To evaluate our visualizations, we conducted a user study with 16 participants with varying levels of literature expertise, asking each to explore a story of their choice using STORY RIBBONS. Finally, we interviewed three additional literary scholars for expert feedback.

Our findings suggest that despite limitations, LLMs can meaningfully augment traditional text visualizations. Although LLMs proved unreliable at extracting information when used naively, we were able to design a data pipeline that was sufficiently reliable to be helpful to users. The flexibility of LLM-powered analysis allowed us to visualize a variety of high-level concepts, and enabled users to invent their own dimensions for visualization. Furthermore, the fact that LLMs could provide justifications for their outputs helped in calibrating trust.

To summarize, our main contributions are:

- **An LLM-powered pipeline** for extracting and organizing character, location, theme, and scene data from unstructured text. We believe our design can be helpful to others working with LLMs.
- **STORY RIBBONS, an interactive literary analysis tool.** The system illustrates what we believe are important LLM-based interaction motifs: providing custom text analytics on demand, as well as explanations for LLM-extracted information.
- **User study findings and expert feedback** highlighting how users interact with LLM-enhanced visualizations; namely, ways in which our tool is useful as well as areas for future research.

2 RELATED WORK

The history of finding a visual form for a story is long and rich [3, 14, 17]. Our work centers on storyline visualization, popularized in 2009 by Randall Munroe’s hand-drawn charts on xkcd [40]. While many early computational efforts to visualize storylines focused on optimizing layouts [2, 12, 32, 44, 58–60], we aim to enrich these visualizations from a data and interaction perspective by leveraging novel LLM technologies. Additionally, in contrast to recent efforts on AI for automatic visualization generation [13, 41, 61, 68, 69], we use AI to extract meaningful insights from stories to visualize. Below, we outline current challenges and opportunities for creating narrative visualizations (Sec. 2.1) and using NLP techniques to augment literary analysis (Sec. 2.2).

2.1 Visualizing Narratives: Challenges & Opportunities

Storyline visualizations help users analyze complex narratives across various domains [11, 20, 37, 51, 53], including news stories [9], political relationship data [23, 45], and interactions between LLM agents [33]. In literature and film, which is our focus, researchers have explored variations of traditional storyline visualizations [40], such as hierarchical and radial layouts (e.g., StoryPrint [66] and StoryCake [48]) or adding two time axes for nonlinear narratives (e.g., Story Curves [28]).

However, storyline visualizations are often limited in scalability and complexity due to the challenges of processing text data [62]. As described in [28]: “*To extract story elements (scenes, characters, etc.) we implemented a parser for segmenting a [movie] script... Unfortunately, not all scripts are well formatted... To work around this problem, we developed a tagging interface to fix the labels.*” With novels, parsing is even more difficult, due to the lack of metadata such as explicit scene

divisions and character labels [20]. We aim to reduce the manual effort involved in processing unstructured stories, while maintaining data quality and faithfulness, by experimenting with LLM capabilities.

2.2 NLP-Enhanced Story Analysis

Historically, computational forms of literary analysis have been fairly limited to vocabulary or syntactic measures, such as tracking word frequencies and average word lengths [27], analyzing concordances [52], or exploring dependency links [62]. Thus, with recent advances in natural language processing (NLP), researchers have begun to experiment with new analytical approaches. For instance, to create Portrayal [20], an interactive visualization system for character analysis, the authors developed an NLP pipeline to extract character traits from fiction novels using SOTA co-reference and sentiment analysis models. However, this process still required several elements of manual parsing and tagging, which is where we see an opportunity for LLMs to step in.

Given their impressive text processing and analysis capabilities, many works explore different ways of using LLMs to analyze stories [29, 54]. In [47], the authors train a small language model to understand literary social networks. [24] introduces a framework for prompting LLMs to uncover implicit character portrayals, and [46] studies the application of LLMs in narrative discourse understanding. Most similar to our vision, StoryExplorer [71] and Clover Connections [21] create LLM-enhanced, visualization-based interfaces to enhance user understanding of stories. However, StoryExplorer is a human-in-the-loop system that requires user annotations. Clover Connections uses LLMs to extract character traits, but we design and validate an almost fully-automated, LLM-driven data processing pipeline. We also place a larger focus on calibrating user trust in AI-extracted literary insights.

3 GOALS & TASKS

Literary analysis differs from conventional data analysis due to its open-ended, interpretive form. In contrast to analytical tasks where visualization is designed to uncover facts and numerical patterns, it often involves navigating multiple valid readings rather than converging on a single truth [21, 35, 42]. To explore how LLM-enhanced visualizations might facilitate this process, we interviewed three literary scholars, who are expert analysts: an English professor, a comparative literature professor, and an English Ph.D. student. We asked scholars (**C1-3**) about their current practices, as well as their hopes and concerns for incorporating LLMs into literary workflows. Following [35] and the tradition of co-design [5, 57, 72]², these experts provided feedback throughout our design process and are co-authors of this paper.

3.1 Design Goals

Overall, scholars wanted to see how LLMs could enhance and extend their existing knowledge of literary works: “*I feel like [LLM] analyses can be powerful in helping us see literature under a new interpretation*” (C2). Similarly, C3 viewed the prospect of integrating an LLM into their analysis process as “*having a partner to bounce ideas off of, which can help you clarify your own perspective.*” C1 was curious if LLMs could help capture and visualize unexpected story patterns, as in [42]: “*I want to see surprising things in the visualization. For example, is one character a lot more prominent than others?*”

From these formative discussions, we identified the following design goals to help analysts uncover new literary insights with LLMs:

G1 - Support flexible analysis workflows. Each scholar had unique analytical interests, reflecting the highly personalized nature of studying literature. C1 noted that elements like “*locations and themes [are] interesting, but not the most important [as] my work focuses on character prominence.*” C2 was interested in “*how characters are defined by language and dialogue,*” but said “*settings are interesting*” as well.

Scholars also performed different kinds of literary comparisons. Some made absolute comparisons, e.g., identifying the most important character in a scene (C2) or analyzing character gender distributions (C1). Others focused on dynamic trends, e.g., how character networks

²A collaborative research practice where real users of a system are included in the design process.

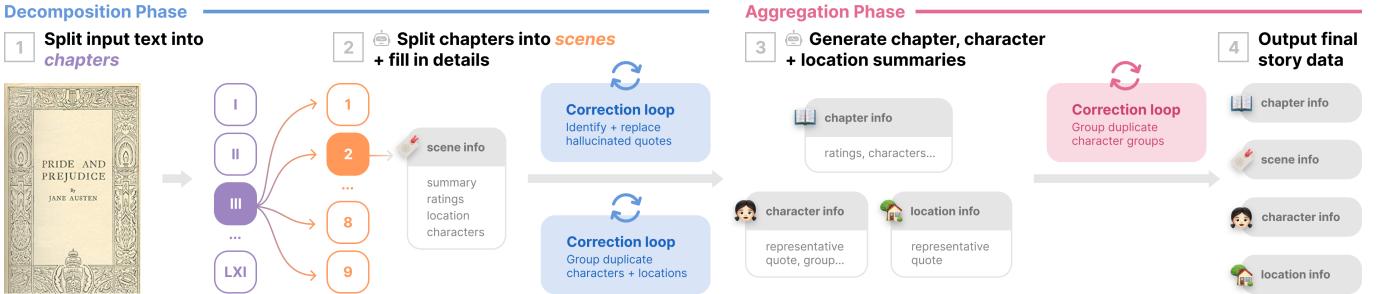


Fig. 2: Overview of our story analysis pipeline, which is organized into a data decomposition and aggregation phase. Steps involving an LLM are denoted with 🗂️. Correction loops are included to check and correct LLM output; each runs once per story. Our pipeline is highly customizable to different literary genres (e.g., novels, plays) and elements – all steps involving *character* data can be swapped out for another aspect such as *theme*.

and emotions change over time (C1, C3). Given these diverse goals, we wanted our visualizations to be customizable to individual users [52].

G2 - Calibrate user trust and provide system transparency. Although scholars expressed optimism and curiosity about LLM-mediated literary analysis, they also shared concerns. C3 explained the significant resistance in infusing technology into literary research: “*Many scholars are still wary of AI, [and] think of it as something that might replace them,*” echoing [35]. C1 and C2 were also worried about LLM hallucinations, emphasizing that “*it’s important to point out when things might be hallucinated*” and provide clear explanations “*in order to avoid making people feel suspicious.*” Our work aims to prioritize trust and transparency, while limiting the potential for hallucinations.

3.2 Design Tasks

We then translated these goals into the following set of design tasks:

T1 - Enable multiple levels of story exploration. To accommodate a wide range of analysis workflows [G1], we support both *high-level* (e.g., chapters) and *low-level* (e.g., scenes) exploration of stories.

T2 - Track key story elements over time. Similarly, we visualize information about many story aspects such as *characters*, *locations*, and *themes*, facilitating discovery based on user interests [G1].

T3 - Provide custom views on demand. Users can add new views by leveraging LLM capabilities, directly tailoring the visualization to answer individualized and spontaneous research questions [G1].

T4 - Explain AI decisions. We aim to provide explanations of LLM-generated data to maintain transparency and protect user trust [G2].

T5 - Connect visuals to raw text. To further calibrate trust [G2], we link visualizations to the story text so users can inspect AI insights.

4 STORY ANALYSIS PIPELINE

To explore **RQ1**, we created an LLM-powered analysis pipeline that automatically extracts and organizes narrative data from literary works.

4.1 Pipeline Design

We followed an iterative process to build our story analysis pipeline. Our pipeline consists of an LLM chaining workflow that decomposes narrative processing into subtasks, inspired by common crowd programming patterns [18]. The open-ended nature of literary analysis led to various undesirable model behaviors – including hallucinations and inconsistent scene segmentations – that required careful prompting and validation strategies. For example, we implemented multiple **correction loops** throughout the pipeline to detect and correct unexpected LLM outputs, similar to quality checks used in crowdsourcing [10].

4.1.1 Overview

Our pipeline contains four steps, comprising a decomposition (D) and aggregation (A) phase [4] (Fig. 2). We describe the process for analyzing *characters* in a *novel*, but our pipeline works for multiple genres (e.g., plays, poems) and analysis targets (e.g., themes). Each correction loop runs once per story; see Sec. 5.4 for full implementation details.

1 - Split input text into chapters (D). We first split the selected story (e.g., *Pride and Prejudice*) into *chapters* (or *acts* for a play). **This is the**

only step requiring human assistance.³ We tried using an LLM to identify chapter markers, e.g., Chapter I, but the model often produced inaccurate results (e.g., missing chapters or punctuation errors) [31].

Each chapter is further split into lines and annotated with line numbers to help with text parsing. These preprocessing steps were motivated by our observations that LLMs struggle with long contexts (e.g., analyzing the entire story at once), leading to hallucinations, narrative chronology mistakes, and omissions of key events [21, 29].

2 - Split chapters into scenes and fill in details (D). Next, we prompt the LLM to split each chapter into *scenes* to extract key plot points from the story. Initially, we saw inconsistent results across runs, suggesting that LLMs do not inherently have a clear sense of what a “*scene*” is [73]. C1 confirmed that scenes are a complex concept and there is not necessarily a “*ground truth*,” especially for novels, as “*much of the language of scenes comes from plays and films... even with people, how you define a scene could depend on what you’re looking at.*”

Ultimately, our literary scholars agreed that defining scenes based on changes in story location felt most sensible, which we implemented in our final pipeline. We found that providing this explicit definition to the LLM and asking the model to explain why it started a new scene, similar to chain-of-thought prompting [67], enhanced output consistency (to a degree, see Sec. 4.2.3).

For each scene, we ask for a summary, the location, and ratings important to understanding a narrative (conflict [16, 65], importance [14, 43, 74], sentiment [15, 37, 50]). Conflict and importance are specified between 0 and 1 (very high conflict or importance), while sentiment is rated between -1 (very negative) and 1 (very positive).⁴ The LLM also extracts *characters* (or *themes*) in this scene. For each character, the LLM describes their sentiment [21, 66] and emotion [20, 54] (e.g., “excited and carefree”), finding a direct quote from the text as evidence.

Once all scene details are generated, we run two correction loops:

🔗 Correction Loop: Check for Hallucinations

When extracting quotes, the LLM sometimes hallucinates or modifies story dialogue (e.g., changing a third-person POV to first-person).

Solution: We add an exact string match check, replacing all false or modified quotes with a brief LLM explanation of the character’s emotions.

🔗 Correction Loop: Group Duplicate Elements

Characters and locations may be referred to by different names throughout the story (e.g., Jane vs. Jane Bennet vs. Miss Bennet), which the LLM frequently fails to recognize on the first pass.

Solution: We use a second LLM to group duplicate elements to create the finalized character and location lists.

3 - Generate chapter, character, and location summaries (A). With the extracted scene details, we then compose:

³This surprised us: identifying chapter boundaries does not seem hard. One reason may be that it requires the longest LLM context window.

⁴We use LLMs instead of task-specific models (e.g., for sentiment analysis [6]), as our goal was to explore the capabilities of LLMs for story analysis.

- *Chapter summaries*, which contain a brief summary of each chapter, importance and conflict ratings, and a list of character and location counts. For each unique pair of interacting characters, we ask the LLM to summarize their chapter interactions.
- *Character summaries*, which contain a quote about each character and semantic group decided by the LLM (e.g., “main characters”). Each character is also assigned a unique color and explanation.
- *Location summaries*, which contain a quote about each location.

For character summaries, we run one more correction loop:

🔗 Correction Loop: Group Duplicate Elements

As with character names, the LLM may create similar character groups (e.g., Bennet family vs. family members).

Solution: We use a second LLM to group duplicate elements to create the finalized list of character groups.

4 - Output final story data (A). We output all structured *chapter*, *scene*, *character*, and *location* data as a single JSON file.

4.2 Pipeline Evaluation

Data. We assessed our pipeline on 36 stories, including 30 literary works from [Project Gutenberg](#) (21 novels, 5 plays, 2 poems, 2 non-fiction; Tab. 1). To examine potential training data effects (e.g., LLM memorization of popular texts), we tested both ● *lesser-known* ($n = 8$) and ◇ *well-known* ($n = 22$) stories. For evaluation, we considered a story “well-known” if it has a [SparkNotes](#) and [LitCharts](#) study guide. Story lengths also varied (mean: 8846 lines). Our shortest text is *The Metamorphosis* (1752 lines) and longest is *Ulysses* (25435 lines).

To further control for training data effects, the last 6 stories are synthetic novels authored by `gpt-4o-mini`. We generated these novels using an outline-conditioned AI writing workflow [49] with similar iterative decomposition and synthesis steps as our data pipeline [4, 18]. These LLM-generated stories are shorter (mean: 1588 lines) but were unlikely to have appeared verbatim in the training data.

4.2.1 Overall Performance

Tab. 2 provides output statistics on the longest and shortest texts in our corpus. There were no significant performance differences based on story length, or between well- and lesser-known texts. However, the **length of scenes** extracted by our pipeline differed by story type (Fig. 3A left). LLM-written stories had shorter scenes (mean: 33.3 lines) than human plays (mean: 124.1) and non-plays (mean: 52.4).

Quote accuracy – the percentage of real (i.e., non-hallucinated) quotes extracted by the LLM – also varied (Fig. 3A right). Plays scored the highest (mean: 0.97), followed by LLM-generated stories (mean: 0.90) and non-plays (mean: 0.85). These results underscore the importance of our correction loops; without them, the LLM returns a non-trivial number of hallucinated quotes. Similarly, accuracy was higher when finding quotes associated with *themes* (mean: 0.94) compared to *characters* (mean: 0.79), likely because character attribution requires subtle contextual clues when names are not explicitly mentioned [36]. This also makes sense given that LLMs were best at finding quotes in plays, which are largely composed of labeled dialogue.

4.2.2 Study Guide Analysis

As a baseline comparison, we examined SparkNotes and LitCharts study guides, which contain human-written analyses of well-known works. In particular, we compared our extracted characters, themes, and key events to the lists and chapter summaries provided by these guides. We analyzed 6 stories: *The Great Gatsby*, *Alice in Wonderland*, *Romeo and Juliet*, *The Odyssey*, *Pygmalion*, and *Don Quixote*.

Method. We analyzed one chapter or act from the start, middle, and end of each text to study performance across narrative sections. To compare chapter events, we listed key events from (1) our scene data, (2) SparkNotes summary, and (3) LitCharts summary. We then performed a diff-style comparison to identify discrepancies (including in chronology). Characters and themes were qualitatively matched when different names likely referred to the same entity (e.g., “the inevitability

Table 1: List of all 36 stories we processed with our LLM analysis pipeline. We include ◇ well-known, ● lesser-known, and ▲ LLM-generated stories.

Novels ($n = 21$):	◇ Alice in Wonderland, Anne of Green Gables, Candide, Don Quixote, Emma, Frankenstein, Great Expectations, Jane Eyre, Little Women, Pride and Prejudice, Tale of Genji, The Great Gatsby, The Metamorphosis, The Trial, The Wizard of Oz, Ulysses, War and Peace, ● Under the Mendips, Dream of the Red Chamber, The Marrow of Tradition, The Tenant of Wildfell Hall
Plays ($n = 5$):	◇ Hamlet, Pygmalion, Romeo and Juliet, ● Faust, The School for Scandal
Poems ($n = 2$):	◇ The Iliad, The Odyssey
Non-fiction ($n = 2$):	● Queen Victoria, The Art of War
LLM-generated novels ($n = 6$):	▲ Starlight Refugees, The Bookstore of Forgotten Dreams, The Color Thief, Threads of the Infinite, Time-Looped Detective, Whispers of the Tea Route

Table 2: Output statistics from our pipeline on the longest and shortest human and LLM-generated stories (👤 = characters, 🏠 = locations, 💬 = themes, 💬 = quotes). Some story titles are abbreviated for space.

Story	Lines	Chapters	Scenes	👤	🏡	📝	💬
◇ Ulysses	25435	18	190	271	138	274	583
◇ Metamorphosis	1752	3	24	10	5	26	82
▲ Whispers	1741	12	61	25	23	75	165
▲ Bookstore	1388	12	41	14	7	42	103

of fate” vs. “fate”). We report the percentage of overlapping characters, themes, and events between each pair of sources, and across all three.

Results. No major differences emerged across story sections. On average, our pipeline extracted 94.3% of **characters** from SparkNotes and 83.3% from LitCharts. The guides shared a 44.3% overlap, and all three sources had a 26.4% overlap (Fig. 4). Most characters we missed were minor or non-speaking, e.g., in *Romeo and Juliet*, both guides listed Rosaline (who does not speak or appear), while the LLM did not.

For **themes**, we shared a mean 73.8% overlap with SparkNotes and 72.6% overlap with LitCharts. The LLM tended to miss more complex themes (e.g., “Incompatible Systems of Morality”), or those related to language aspects (e.g., “Language and Wordplay”) and broader context (e.g., “The Roaring Twenties”). The low overlap between SparkNotes and LitCharts (36.0%) – and minimal 3-way overlap (3.1%) – also highlights the subjectivity of identifying literary themes.

42.6% of total characters and 91.3% of themes were only detected by the LLM. In *Alice in Wonderland*, for instance, we found 27 characters beyond SparkNotes’ list ($n = 7$), and 15 beyond LitCharts’ ($n = 19$), e.g., the Eaglet and Baby. Similarly, while these guides focused on analyzing 3-5 key themes, our pipeline often picked up 30+ (sometimes even 100s – Tab. 2). Additional themes the LLM found in *Romeo and Juliet* include “Existentialism,” “Political Manipulation,” and “Inaction and Reflection.” However, some of these characters and themes may be too minor or granular to provide user value (see Sec. 7.3).

We identified 90.4% of the same **events** as SparkNotes and 90.1% as LitCharts with 100% accurate chronology. The study guides overlapped 93.6% (3-way overlap: 80.5%). Our data is more structured and concise than prose summaries, but the LLM still captured most major scenes. 2.7% of events were only extracted by the LLM (e.g., when Nick confronts Meyer Wolfsheim after Gatsby’s death in *The Great Gatsby*).

4.2.3 Scene Boundary Analysis

While discrepancies in scene length may be expected due to inherent structural differences between stories (e.g., our LLM-generated texts are shorter and less complex than human writing), we wanted to learn more about the LLM’s understanding of a literary scene.

Method. To do this, we annotated a total of $n = 3796$ scene divisions across all 36 stories (excluding the first scene in each chapter). Each boundary was labeled by examining the explanation provided by the

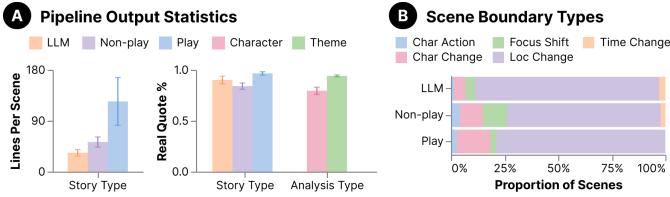


Fig. 3: (A) Comparing the mean number of lines extracted per scene (left) and percentage of real quotes (right) identified across different story & analysis types. Error bars indicate 95% CIs. (B) LLM classifications of scene divisions ($n = 3796$) by story type.

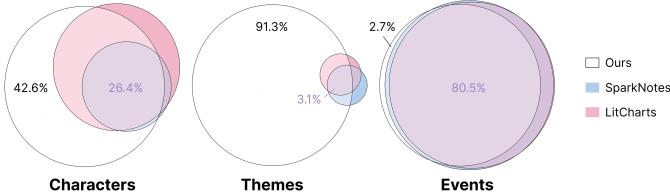


Fig. 4: Visualizing the overlap of key *characters*, *themes*, and *events* extracted by our pipeline vs. literary study guides for $n = 6$ stories.

LLM when starting a new scene and grouping these thematically.

Results. As shown in Fig. 3B, our pipeline extracted 5 main scene division types, meaning the LLM still deviates from its prompt (i.e., location). However, the most common type was *location change*, as expected, making up 72.7% of all scene boundaries. For LLM-generated stories in particular, 85.9% of boundaries were location-related.

The next most frequent scene division for LLM-generated stories and plays was *character change*, where characters enter or exit the scene, making up 10.6% of all scene transitions (or 15.5% for plays). 10.5% of scene boundaries occurred when the text's *focus shifted* (e.g., "The conversation shifts to their future political strategies"), which were the second most common division type for non-plays (11.1%). *Character action* (e.g., "Emma formulates a plan for Harriet's future.") and *time change* transitions (e.g., "K. returns to the office the next day") were the least frequent, making up 3.9% and 2.2% of all annotated scenes.

Our results reinforce how narrative scene segmentation is a challenging task for AI [73], and show that like humans, LLMs' perceptions of a scene may vary based on genre. C2 found it interesting how "*the LLM allows us to question seemingly minor things we take for granted. Like what exactly is a scene? Is it related to setting, character, both?*"

5 STORY RIBBONS DESIGN

Guided by our design tasks (Sec. 3) and **RQ2**, we designed and implemented STORY RIBBONS, an interactive tool for literary analysis. STORY RIBBONS visualizes narrative insights from our LLM-powered analysis pipeline (Sec. 4), allowing users to explore customizable character and theme trajectories for 36 stories (Tab. 1). While most story visualizations track interactions at a fixed time scale (e.g., chapter or scene) [28, 32, 59, 60, 66], we aim to provide a more comprehensive overview of story structure and evolution through visualizing detailed narrative information at multiple levels (e.g., chapter and scene).

5.1 Technique

We adapt the original storyline visualization technique [28, 40, 59], where each character is represented by a trajectory of points – i.e., a "ribbon" (see Fig. 1). Each point in a character trajectory denotes a particular moment in time (e.g., a scene) where they are present. Contiguous regions of points are connected by a curve, and gaps denote character absences. Thus, drawing multiple trajectories in parallel can elicit the "shape" of a story. One addition we make to this technique is using weighted paths to encode **importance**, where the thickness of a ribbon at any point reflects the character's significance in that scene.

We chose to base our work off this technique due to its established utility for visualizing narratives and flexibility for extension. However,

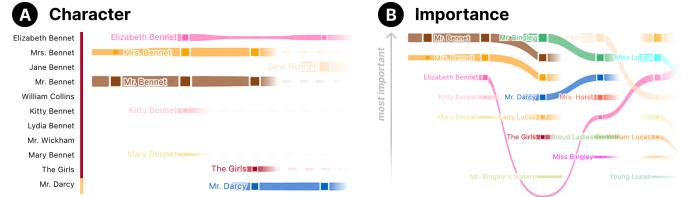


Fig. 5: (A) Static character or (B) dynamic character *importance* y-axis.

our experts also suggested including other visualizations, such as a character network (Sec. 5.3.2), and so a core strength of STORY RIBBONS is its support for switching between multiple views of a story.

5.2 Key Interaction Motifs

Our system includes three key LLM-powered interaction motifs:

★ **Explanations on demand.** When showing scholars (C1-3) early prototypes of STORY RIBBONS, several asked questions like: "*How is the LLM [deciding] what is a scene, and what sentiment and importance to assign to a character?*" (C1). Thus, we provide "explanations on demand," where the user can view explanations for AI-generated data by interacting with the corresponding UI components – similar to Ben Shneiderman's famous "details on demand" mantra for interactive visualization systems [55]. These explanations are designed to empower users to interrogate and verify the LLM's reasoning toward increased transparency and trust [T4].

★ **Natural language dimensions.** STORY RIBBONS also allows users to add new visualization dimensions through custom natural language prompts. For example, users can ask the LLM to rank the characters in each scene by an attribute like "sense of duty," or assign colors to characters based on how "evil" each of them is. In this way, users are not restricted to a fixed set of traits and can shape story exploration around their own interpretative goals [T3].

★ **Natural language queries.** As with any new, complex visualization, users may not know where to begin or which parts are most relevant to their interests. Thus, we include several "ask LLM" widgets throughout the tool to support exploration through natural language queries. With these widgets, the user can ask a question to receive LLM guidance in navigating our visualizations and understanding story insights more deeply [T4]. For example, if the user asks, "Where is the theme of social class most prominent?", the LLM will guide them to the corresponding chapter and segment of the visualization.

5.3 Interface

STORY RIBBONS consists of three views for exploring our storyline visualizations: *Story Overview*, *Detail Overlay*, and *Settings Sidebar*.

5.3.1 Story Overview

The **Story Overview** contains our main ribbon visualization (Fig. 1). In the top left corner, users can view story metadata. To the right, a menu bar provides visualization controls, e.g., the "Character view" toggle switches between visualizing *characters* and *themes* [T2].

Below, the ribbon plot maps time on the x-axis, segmented by *chapter* or *scene* (via the "Chapter view" toggle) [T1]. The color of each chapter label, and the corresponding band, encodes *sentiment* (adjustable in Settings). By default, the y-axis encodes *location* in order of chronological appearance in the text. We use a consistent y-axis to improve readability and preserve context about character interactions [2]; this diverges from traditional storyline visualizations where y-coordinates encode interaction via proximity [40, 59]. Users can switch the y-axis to track other narrative aspects as well [T2]:

- **Character:** plots each character ribbon in its own horizontal lane, organized by group, to help identify co-occurrences (Fig. 5A).
- **Importance:** plots ranked character importance over time, with more prominent characters at the top (Fig. 5B)⁵.

⁵We plot rankings instead of raw importance scores (0-1) because the LLM often assigns similar ratings to characters, making interpretation more difficult.

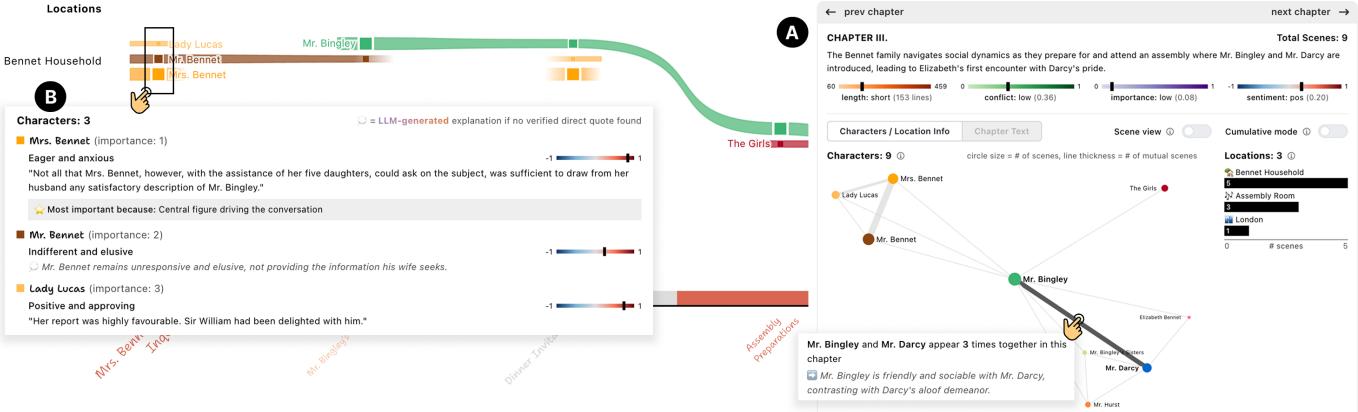


Fig. 6: (A) When the user clicks on a chapter, the *Detail Overlay* opens on the right, revealing additional details and (B) a network visualization of character interactions. (B) Users can also toggle the main plot to view scenes from this chapter and (B) hover for character details.

- **Sentiment:** plots character sentiment over time, with -1: negative at the bottom and 1: positive at the top.

We also support “Rank by trait,” a ★*natural language dimension* that adds a custom y-axis for users to explore by ranking characters or themes in each scene by a specific trait (e.g., inner conflict) [T3].

5.3.2 Detail Overlay

After gaining a high-level overview, users may want to explore the narrative in more fine-grained detail [T1]. To do this, users can invoke the **Detail Overlay** by (1) clicking on a chapter in the main plot, or (2) using the “Ask LLM” button near the x-axis (Fig. 1). (2) opens a prompt box, where users can ask a question about the story (e.g., “When does Elizabeth reject Darcy?”). The LLM identifies the most relevant chapter to the user’s ★*natural language query* and provides an explanation [T4], directing them to the corresponding overlay.

In this view, users can explore the selected chapter in depth (Fig. 6A). At the top, we show the chapter summary and ratings (length = normalized number of lines) [T2]. Below, there is an interactive network visualizing interactions (links) between characters (nodes) in this chapter (or the story *through* this chapter if “Cumulative mode” is on) [25, 30, 37]. The size of each character node encodes their chapter importance, based on the number of scenes they appear in. Edge thickness encodes character co-occurrences. Users can hover on a character or interaction for an ★*explanation* about their role or relationship in this chapter [T4]. On the right, there is a bar chart with chapter locations.

Above the network, users can toggle to “Scene view” (instead of “Chapter view”) to visualize only scenes *within this chapter* in the main ribbon plot [T1]. Hovering on a scene will open a similar overlay with a list of characters that are present in the scene, ranked by importance, as in Fig. 6B. For each character, we show the LLM’s description of their emotions and a corresponding quote [T5]. Here, the LLM did not find a direct quote for Mr. Bennet, so an explanation is displayed instead. Users can also see the LLM’s ★*explanation* for why it chose the top character as the most important in this scene [T4].

To view the “Chapter Text,” users can click the corresponding button [T5] (Fig. 7). Our visualizations automatically scroll as you read and show the corresponding scenes on the right. Hovering on the ? icon next to a scene title displays the LLM’s ★*explanation* for starting a new scene [T4]. Users can “ask LLM about this scene” (or chapter), which uses the story text to promote deeper exploration of the narrative or the LLM’s decision-making process through ★*natural language queries*.

5.3.3 Settings Sidebar

To further customize the ribbon plot, users can open the **Settings Sidebar** via the button in the top right corner of Story Overview (Fig. 1).

In the “Characters” settings, users can change the ribbon colors (Fig. 8A). By default, we use the *LLM-assigned* colors, but ribbons can also encode character group, importance, or sentiment [T2]. Group

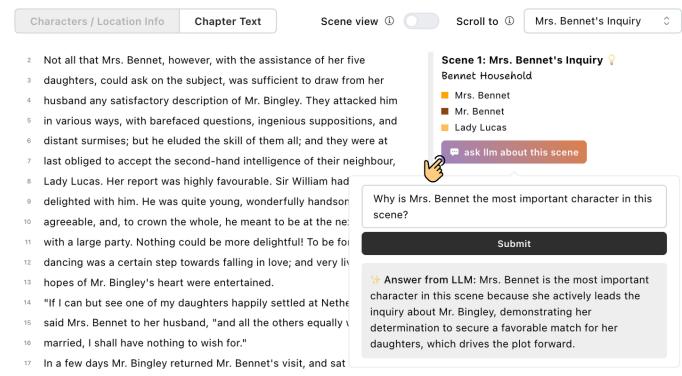


Fig. 7: Viewing raw chapter text inside of the *Detail Overlay*. (B) Here, users can ask the LLM a question about the current scene or chapter.

colors are chosen from a discrete color scale, while importance uses a continuous, sequential scale and sentiment uses a continuous, diverging scale between blues (negative) and reds (positive). Users can also “Categorize by color”, a ★*natural language dimension* that adds custom color palettes based on an attribute (e.g., wealth or age) [T3]. The LLM assigns each character or theme a value of the specified attribute (e.g., lower, middle, or upper class),⁶ along with a corresponding color.

Under the color dropdown, there is a filterable character legend organized into groups. Here, users can find characters or subgroups of interest and highlight/hide their corresponding ribbons. Upon hovering on a character in the legend, or a ribbon in the main plot, a popup opens with an LLM-selected quote about the character and ★*explanation* for their current color encoding (e.g., why Elizabeth is a “unique” beauty) [T4] (Fig. 8C). Hovering on a location brings up a similar popup.

Below, in the “Chapters” settings, users can customize the chapter labels and corresponding color bands (Fig. 8B). Label color, size, and weight can be used to encode information about chapter *importance*, *sentiment*, *conflict*, *length* (normalized), and *number of characters* (normalized) [T2]. Like the character ribbons, continuous, sequential color scales are used to depict each characteristic. Users can toggle “Scale by length” to scale the width of each color band to reflect true chapter length; by default, the x-axis is broken into equal segments.

5.4 Implementation

STORY RIBBONS is a full-stack web application with a Python/Flask backend that communicates with a React/TypeScript frontend. Our main ribbon plot is a custom SVG visualization constructed with Bézier curves [39]. The character network is implemented using D3.js.

⁶We use discrete values for easier attribute encoding and user interpretation.

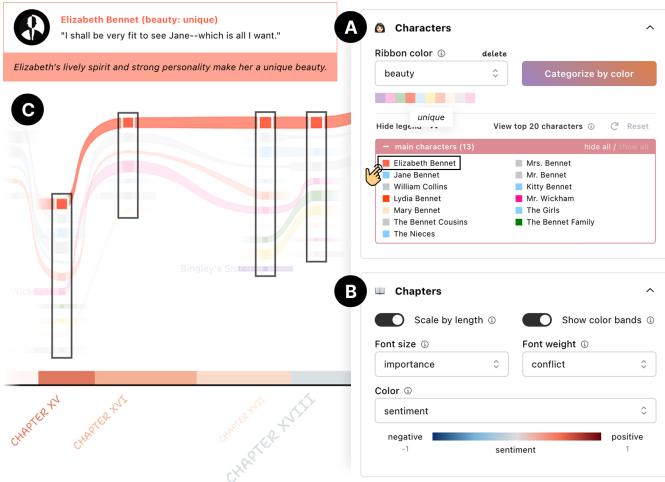


Fig. 8: Users can customize (A) character ribbons and (B) chapter labels in the Settings Sidebar. (C) Hovering on a character highlights their ribbon in the main plot, along with the corresponding overlay.

Our backend uses [Langchain](#) and [gpt-40-mini](#) to power all on-the-fly features (e.g., “Ask LLM”, “Rank by trait”) and most story analysis steps due to overall high output quality as well as speed and cost efficiency. On-the-fly operations take the user’s query and our extracted story data as input. Our LLM prompts and example user queries are included in the supplementary materials.

For correction loops, we use [claude-3-5-sonnet](#) as our second LLM to group duplicate story elements, as Claude outperformed GPT on this task during pilot studies. The idea of leveraging multiple LLMs based on their unique strengths is inspired by work such as [63]. We parallelize several pipeline steps (e.g., scene splitting & analysis) to reduce latency. Running our pipeline on *The Great Gatsby* (4710 lines) takes 2.2 minutes, while *Little Women* (16680 lines) takes 3.9 minutes.

5.5 Usage Example: *Pride and Prejudice*

This design was heavily informed by collaboration with our scholar co-authors. To illustrate how they used the tool, we provide a brief vignette from an exploration of *Pride and Prejudice*. While ranking characters by **importance**, C2 was interested in the LLM’s characterization of Mr. Bennet as the most important character at the beginning of the novel: “*That surprised me because it’s very instinctive to read Mr. Bennet as more of a side character*” (Fig. 5b). Upon reflection, C2 said, “*Now I do see how Mr. Bennet is very important. He’s in the middle of the scene and everyone is talking to him... He’s the one who arranges for his daughters to go to Mr. Darcy and Bingley’s house.*”

Next, C2 visualized the LLM’s concept of **beauty** in the novel by adding a custom color palette, and was surprised that instead of a scale from ugly to pretty, the LLM created different categories of beauty (e.g., natural, classic, exotic). C2 was intrigued that both Elizabeth and Charlotte were classified as “unique” beauties “*because in the novel, Charlotte is called downright ugly. And she’s Elizabeth’s best friend, so she’s really sad that she’s not as pretty*” (Fig. 8C). Ultimately, C2 thought “*this is great. The visualization makes you think about beauty in a different way... and it’s not that the LLM got it wrong, but it’s making us see the characters and story in a different way.*”

6 STORY RIBBONS EVALUATION

We evaluated our tool through a user study and expert interviews, synthesizing findings with feedback from our original scholars (C1-3).

6.1 User Study

Our user study involved asking participants to explore a story of their choice using STORY RIBBONS and its interactive visualizations.

Participants. We recruited a total of 16 undergraduates, graduate students, and working professionals who are strongly interested in liter-

ature (P1-16). 5 participants self-identified as *expert* literary analysts (e.g., English major who regularly analyzes literature), 5 as *intermediate* analysts (e.g., took multiple English classes and familiar with literary analysis), and 6 as *novice* analysts (e.g., avid reader but little to no formal literary training). 12 out of our 16 participants had never seen a narrative visualization before.

Procedure. Prior to the study, we asked each participant to select a story they are familiar with. Our goal was to make the task more personalized and ensure that users could meaningfully assess the LLM’s insights. In total, participants examined 11 unique literary works.

We started each 1-hour session by asking participants a few pre-task questions about their experience with literary analysis and visualizing stories. This was followed by a walkthrough of our interface and visual encodings using their chosen story. Next, participants had ~30 minutes to dive analytically into their story by interacting with STORY RIBBONS and its features. Participants were also asked to think aloud. The study concluded with a post-task interview to gain deeper insights into each participant’s experience with STORY RIBBONS, how they perceived the LLM-generated data, and other feedback about our visualizations.

6.2 Expert Interviews

We conducted semi-structured interviews with 3 new literary scholars to elicit feedback about STORY RIBBONS from a scholarly perspective. These scholars included an English Ph.D. student and two English professors (S1-3). Each interview followed a similar structure as the user studies but were shorter (~30 minutes) and more focused on collaboratively exploring the system and ideating potential use cases.

7 RESULTS

Overall, participants enjoyed using STORY RIBBONS to explore stories, noting that our system provided a powerful, intuitive way for getting a “*bird’s eye view*” (P6) and understanding “*how a story arc develops over time*” (P3). Users like P4 and P14 found our tool useful for revisiting “*specific scenes [in] familiar stories*” and “*keeping track of everything – especially for a complex story like *The Odyssey**” ($n = 12$). Our visualizations often reminded users about story elements as well: e.g., “*I’d forgotten there was a king of hearts!*” (P1, *Alice in Wonderland*) or “*I didn’t realize Nick is in every chapter*” (P8, *The Great Gatsby*).

7.1 Case Study: *The Metamorphosis*

To illustrate STORY RIBBONS’s utility, we share how P4 used our tool to gain new insights about *The Metamorphosis* by Franz Kafka. P4 began exploring the story by viewing our **scene breakdown** along the x-axis: “*Oh, Gregor’s death [is] called The Family’s Decision. That’s an interesting reframing because they do just decide to let him die and Gregor doesn’t have much agency over anything during the story*” (Fig. 9). They added, “*If you compare this to SparkNotes where it’s more like a description of what happens, this has a lot more sauce. It’s trying to say something about the scene, like why is it there? And that’s central to the point of analysis, which really fascinates me.*”

Next, P4 changed the y-axis to **hope** to trace Gregor’s ups and downs: “*That seems accurate. First he’s shocked about the situation and then sort of accepts things. But then, his family has to get jobs and he feels bad. Then he feels better because he’s like my family cares about me. But then he’s like they threw an apple at me. And now I’m okay with dying.*” P4 also found it “*funny that the sentiment*” along the x-axis shows “*they’re feeling relieved and happier after Gregor dies*” and wondered if the LLM understood the ending’s “*dramatic irony*.”

However, P4 noted that the LLM seemed to miss the nuance in Grete’s character. When ranking characters by **importance**, Grete was often near the bottom: “*Maybe she’s more of a minor character if you count number of appearances, but [I think] she’s actually one of the most important people, because of the themes about family and alienation [and] Grete’s betrayal of Gregor is such an important scene.*” Coloring by **sense of duty**, P4 said “*it’s interesting that [the LLM] captures Grete’s caring nature towards Gregor*” in classifying her as “*high*.” They also disagreed with the LLM labeling Gregor as “*low*”: “*It’s true Gregor’s being alienated from his human self, but the primary source of his oppression is feeling like he’s a burden to his family.*”

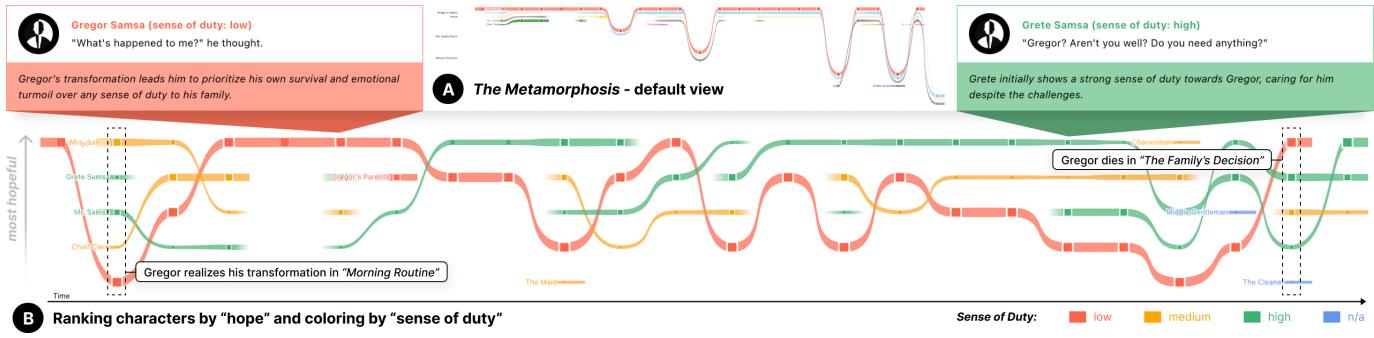


Fig. 9: STORY RIBBONS visualization of scenes in *The Metamorphosis*. (A) The default view with locations along the y-axis. (B) After ranking characters by “hope” and coloring the ribbons by “sense of duty.” Corresponding color explanations for Gregor and Grete are included.

7.2 User Feedback

Users appreciated STORY RIBBONS’ high degree of **customization** ($n = 12$): “*The customizability is very cool. You’re practically able to do anything*” (P12) and can “*adapt it to what [you] want to see*” (P3). S1 said it encouraged them to explore new story aspects: “*I don’t look at sentiment analysis much, but I really enjoyed seeing the sentiment view and how [the LLM] summed it up with color.*” However, 6 users noted the visualization and customization options could be “*too much at times*” (P6), especially for longer stories with many characters, and they “*felt a bit overwhelmed by all the things happening at once*” (P3), suggesting opportunities to better manage cognitive overload.

Participants also emphasized that story **familiarity** makes the visual analysis process more meaningful ($n = 8$): “*I’d probably only use this tool for books that I’ve read because then I kind of know what to look for. If I haven’t read a book, it’ll take my joy away [in] discovering themes and analyses myself*” (P13). Similarly, P9 thought STORY RIBBONS “*would be most helpful for [people] who are familiar with the book [and] want to explore new questions or interpretations.*” C1 and P12 had concerns about people “*taking [the LLM’s] words as the truth and trusting it too much*” if they have not read the story before.

7.2.1 Most & least useful features

14 participants enjoyed using “**Categorize by color**” as a “*fun entryway into analysis*” (P4) that “*lets me consider hypotheses that I might not have had otherwise*” (P2), visualizing attributes like authority, loyalty, and evilness. Some explored applying modern concepts to classic works (e.g., P11 colored *Pride and Prejudice* characters by MBTIs.) Similarly, 12 participants found “**Rank by trait**” useful for gaining “*different views [and] insights*” (P5), especially in “*how characters embody a theme*” (S2) such as vengeance or feminism. C2 also tried ranking characters by language traits such as humor or wordplay.

9 users described “**ask LLM**” as a “*more precise Ctrl+F*” that can “*focus me on a chapter based on my interest*” (P3), which is helpful “*especially [for] sifting through a huge book*” (P9). 4 participants used it to locate key events: e.g., “*My favorite part is when Mr. Rochester proposed to Jane Eyre. [I’m] curious what I can see from the visualization about that*” (P13). Users also enjoyed reading LLM-extracted quotes and explanations ($n = 8$), tracking and filtering characters in the legend ($n = 4$), and viewing story text to verify AI insights ($n = 3$). P2 and P11 thought the x-axis color bands were useful in “*seeing how the LLM chopped up the book [and] observing trends like conflict*.”

10 participants said the **character networks** were their least favorite feature. While some like P5 saw them as “*a cool new way to interpret the text*” ($n = 5$), others found it “*confusing*” when “*it had many nodes [and] everything was connected*” (P1). P9 noted that “*the number of interactions [is] not as important as the weight or nature of those interactions*” (P4). Users were also split on the utility of visualizing **locations**. P9 and P14 enjoyed seeing “*the physical journey*” of characters and themes, and P12 valued this view as someone with aphantasia who “*doesn’t have mental imagery*.” However, 3 users found this y-axis less informative for stories with fewer settings and 2 thought it “*could be misleading if it’s not ordered by proximity*” (S3).

7.3 Limitations of LLMs

User also discovered several interesting behaviors and limitations of LLMs while interacting with STORY RIBBONS.

7.3.1 Context and granularity challenges

We observed in Sec. 4.1 that LLMs may refer to the same character or location by different names. Our correction loops aim to consolidate duplicate elements; however, this task requires **context** about the current scene or entire story, which the LLM struggles with. As P14 noticed in *The Odyssey*, both Polyphemus and Cyclops were listed as characters, but “*Polyphemus is the name of the Cyclops, unless I’m remembering wrong?*” Similarly, in *The Metamorphosis*, P4 was surprised that besides Mr. and Mrs. Samsa, “*there’s another category of Gregor’s parents. It seems [the LLM] can’t infer those relationships.*”

Users also observed the LLM having trouble finding the right **granularity** for analysis. 5 participants commented that our themes felt “*not super well defined*” (P15) and “*maybe too granular*” (P3). For example, in *Jane Eyre*, there were many related themes like “*nature’s beauty*” and “*nature’s harshness*,” but P5 was “*curious about [not seeing] feminism*,” a core “*theme I enjoyed in the book*.” Users like P8 wondered how the LLM defines a character, as in *The Great Gatsby*, it included T.J. Eckleberg, “*a symbol with a human name*,” and very minor figures like First Girl. P6 asked how character groups were determined, noticing that *Pride and Prejudice* had “*main characters*” and “*upper class*,” which could overlap (e.g., Mr. Darcy).

7.3.2 Lack of advanced analytical capabilities

8 participants noted the LLM’s **inability to surface deeper literary insights**. For instance, P1 asked the LLM about takeaways from *Alice in Wonderland* but “*its answers [were] more surface level*,” e.g., the absurdity of authority. P2 and P13 wanted “*the LLM [to] perform holistic analyses*” that synthesized and “*weren’t so bounded by chapters*.”

Like P4 (Fig. 9B), P7 and P15 noticed that the LLM overlooked a lot of complexity in its character analyses, e.g., fixating on Amy’s selfishness and Jo’s kindness in *Little Women*, while “*Amy and Jo are both flawed characters [and] a bit selfish in their own ways*.” 4 other participants observed similar behaviors where the LLM’s explanations or “*quotes aren’t super representative [and] don’t tell me anything interesting about how the characters relate to*” a theme (P16).

7.3.3 Impact on trust

14 participants said unexpected LLM behaviors **did not affect** their overall experience with or trust in our visualizations. 8 noted, “*It’s not surprising there were errors*” (P14) and “*we should have some degree of doubt about what [LLMs] are saying*” (P12). As P4 put it, “*it’s like literary criticism. You don’t have to agree. It’s still interesting [and] just one point of view you can gain something from*.” P9 added, “*I believe in my own authority over the LLM*,” and P11 said, “*If anything, it would just make me dig deeper to see where [the LLM] is getting at*.”

However, 5 users shared concerns about the LLM’s **more subjective** ratings such as importance, as “*that’s such an interesting literary category and a large part of the critic’s job*” (S2). P10 was apprehensive

because even with our provided explanations, “*I have no idea how accurate it is [and] no gauge for what they see as important.*” Similarly, P8 said they would be cautious about trusting “ask LLM” for “*personal or philosophical questions that require deep analysis and discussions.*”

7.4 Use Cases & Applications

■ **Pedagogy.** Both participants and scholars saw immense potential for a tool like STORY RIBBONS in pedagogical settings. 7 users described how looking at “*themes and unexpected connections could [be] great for starting discussions*” or “*coming up with writing prompts for high schoolers or undergrads*” (S1). 8 participants mentioned that our tool could provide valuable essay writing support as well: “*If I were writing an essay [but] didn’t flag everything while reading, then I could use the tool to find quotes and text evidence that I forgot*” (P9).

● **Book clubs.** Similarly, 4 users envisioned our tool facilitating discussions in book clubs. P2 noted, “*This would be fascinating to have the whole timeline [and] help us see where the interesting discussions are.*” P7 said our visualizations “*would be fun for people [to] look at things together and see if they got the same things out of a book.*”

? **Scholarship.** Some experts noted that in its current form, STORY RIBBONS may not fit their specific workflows, but they ideated ways of adapting our tool to meet various scholarly goals. 4 were interested in using LLMs to incorporate multiple perspectives or “*knowledge sources like SparkNotes and other literary experts*” into our visualizations to provide “*meta analysis about the author [and] cultural context*” (C2, P4). 8 users wanted to add a more comparative dimension, enabling cross-novel analysis and “*reading at a nonhuman scale [where] you could query across a large corpus of works*” (C3, S2). Users also asked to visualize different kinds of texts, including historical timelines (S3), nonlinear narratives (P6), and stories in other languages (S1, P9).

✍ **Writing.** 7 participants suggested extending our tool to have more of a writing or authoring focus. For instance, P5 wanted to modify the visualization to explore “what if” questions like: “*Hamlet is a tragedy [but] let’s say Hamlet survived. How would the story be?*” C3 also imagined using LLMs to “*regenerate stories from different perspectives [or] fill in missing gaps,*” while S3 saw an opportunity to integrate similar storyline visualizations “*inside of writing tools to help writers while writing complex novels and TV shows.*”

8 DISCUSSION & FUTURE WORK

Our results offer insight into current limitations and opportunities for AI-enhanced text visualizations.

8.1 Sparking and Drawing From Literary Conversation

Multiple users suggested that STORY RIBBONS could serve as a valuable **conversation catalyst**, particularly in group settings such as classrooms or book clubs (Sec. 7.4). That is, the visualization could serve as a visual aid or reminder of a book, consistent with cognitive psychology research on the effectiveness of visual memory aids [56]. Interestingly, several users observed that although the LLM’s interpretation of narrative elements did not always align with their perspectives, that in itself could spark conversation in meaningful ways (Sec. 7.2).

Several participants noted the fact that our system is limited by only having access to the story text (e.g., P15: “*if I were to read this book in a vacuum, I’d probably miss a lot of the things the LLM did*”). P5 also reported the importance of “*knowing what others have observed [and] their perspectives on how pivotal a given part is in the story.*” An important future direction would be to give STORY RIBBONS access to this **wider literary conversation**, adding critical work to the base text.

8.2 Toward More Integrated LLM-Visualization Interactions

Users hinted at the value of tightly integrating LLM insights and their corresponding visualizations for *authoring* and *question-answering*.

Visual authoring. After interacting with STORY RIBBONS, many users wanted to manipulate our ribbon plots and see how the underlying stories would change (Sec. 7.4). Some works are starting to explore this idea of linking text and visualizations for creative storytelling. For example, [34] introduces the idea of “visual writing,” where users can author stories by manipulating character traits and timelines.

TaleBrush [7] and Patchview [8] are two other visualization tools that champion interactive story sketching and worldbuilding with LLMs.

In this way, LLM hallucinations can be leveraged for “good,” enabling new modes of visual authoring and literature exploration. P2 and S3 thought our tool’s visualization of scene structure could be especially valuable for writers to learn “*how different authors present [their] stories*” and “*what a literature-quality story looks like visually.*”

Visual question-answering. Our work suggests new directions for visual question-answering, which typically focuses on answering questions about image and video data [1]. We propose extending this paradigm to text data, where users interact with dynamic visualizations to explore and answer literature analysis questions.

While related to prior work on using natural language to generate or modify data visualizations [41, 61], we envision a more integrated workflow where LLMs proactively guide exploration of user queries by customizing visualizations and directing attention to relevant aspects of the text. For example, users like P12 wanted to directly probe literary visualizations with LLMs: “*I’d like to pick a section of the ribbon and [have] the LLM tell me something about that,*” going beyond our current natural language features. P14 also suggested having a linked history view for ask LLM requests that “*shows you all the questions [and] where you asked them in relation to*” a visualization.

8.3 Scaling Analytical Power and Trust: A Tradeoff

Another emergent theme was the tradeoff between wanting to leverage LLM-infused visualizations for more complex and large-scale tasks but having concerns about output faithfulness (Sec. 7.3, 7.4). Most participants agreed that the LLM was best at “*objectively organizing things like where certain characters and themes pop up*” that are “*tangible and concrete*” (P9, P10). Users hoped to capture more nuanced analyses (e.g., cross-chapter comparisons or philosophical queries), but were hesitant to trust the LLM in these cases, as “*it’s harder to know what it’s looking for*” (P8), echoing findings from [70].

Participant sentiments highlight the larger need for greater **explainability and interpretability**, especially in fields like literature where “*someone is passionate about it and wants details. So if [the LLM] misses out and hallucinates, that’s a major lack in trust*” (P5). Our participants were on average highly educated and likely more familiar with LLMs than the general public, but other users may be prone to AI overreliance. While we provide explanations for LLM decisions, these are currently prompt-based and may not always be fully faithful to model internals – highlighting an important area for future research. For example, literary visualization tools could include more robust explainability measures such as confidence scores (P3) [22] or RAG-based textual evidence for LLM responses (P7, P9) [31, 64]. This is particularly crucial for distant reading applications [38] where scholars compare works at scale, and “*it would be harder to know whether to trust the LLM’s interpretations, especially if I’m not familiar with everything it’s showing*” (S2).

9 CONCLUSION

STORY RIBBONS augments traditional storyline visualization techniques with AI text-processing capabilities. Our system is based on a custom LLM-powered data processing pipeline, which extracts detailed semantic content from novel-length stories. The pipeline design uses multiple “correction loops” to make it resilient to AI errors. STORY RIBBONS’ visualizations extend the expressive power of standard storylines, with additional visual encodings to display the rich details produced by the data pipeline. Our tool also supports the interactive use of LLMs to produce explanations on demand and add custom visualization dimensions to personalize story exploration.

Feedback from a user study, along with conversations with literary scholars, demonstrate that STORY RIBBONS can provide new insights about and illuminate interesting paths for further exploration of a text. At the same time, our findings reveal limitations in what current LLMs can achieve – particularly in grappling with literary nuance, understanding narrative context, and resolving ambiguities. Even so, our results suggest that STORY RIBBONS represents a promising tool for literary analysis, and its design strategies for the integration of AI may be helpful in creating and enhancing other visualizations.

ACKNOWLEDGMENTS

We thank the participants in our user study and expert interviews for their time and invaluable insights. Additionally, we would like to thank Andrew Lee and Lucia Gordon, along with the anonymous reviewers, for their helpful feedback and suggestions in revising this paper. We are also grateful for the support provided by the members of the Harvard Insight + Interaction Lab.

This research was supported by CY’s National Science Foundation Graduate Research Fellowship under Grant No. DGE 2140743 and Kempner Institute Graduate Research Fellowship. MW and FV received support from the Effective Ventures Foundation, Effektiv Spenden Schweiz, an Open AI Superalignment grant, and the Open Philanthropy Project.

REFERENCES

- [1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015. doi: [10.1109/ICCV.2015.279](https://doi.org/10.1109/ICCV.2015.279) 9
- [2] D. Arendt and M. Pirrung. The “y” of it matters, even for storyline visualization. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 81–91, 2017. doi: [10.1109/VAST.2017.8585487](https://doi.org/10.1109/VAST.2017.8585487) 2, 5
- [3] M. Bernstein. Patterns of hypertext. In *Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space*, pp. 21–29, 1998. doi: [10.1145/276627.276630](https://doi.org/10.1145/276627.276630) 2
- [4] M. S. Bernstein, G. Little, R. C. Miller, B. Hartmann, M. S. Ackerman, D. R. Karger, D. Crowell, and K. Panovich. Soylent: a word processor with a crowd inside. In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, pp. 313–322, 2010. doi: [10.1145/1866029.1866078](https://doi.org/10.1145/1866029.1866078) 3, 4
- [5] I. Burkett. An introduction to co-design. *Sydney: Knode*, 12:12, 2012. 2
- [6] K. E. Chu, P. Keikhosrokiani, and M. P. Asl. A topic modeling and sentiment analysis model for detection and visualization of themes in literary texts. *Pertanika Journal of Science & Technology*, 30(4):2535–2561, 2022. doi: [10.4783/pjst.30.4.14](https://doi.org/10.4783/pjst.30.4.14) 1, 3
- [7] J. J. Y. Chung, W. Kim, K. M. Yoo, H. Lee, E. Adar, and M. Chang. Talebrush: Sketching stories with generative pretrained language models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–19, 2022. doi: [10.1145/3491102.3501819](https://doi.org/10.1145/3491102.3501819) 9
- [8] J. J. Y. Chung and M. Kreminski. Patchview: Llm-powered worldbuilding with generative dust and magnet visualization. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–19, 2024. doi: [10.1145/3654777.3676352](https://doi.org/10.1145/3654777.3676352) 9
- [9] M. Costa and S. Nunes. Newslines: Narrative visualization of news stories. In *Text2Story@ ECIR*, pp. 37–46, 2023. 2
- [10] F. Daniel, P. Kucherbaev, C. Cappiello, B. Benatallah, and M. Allahbakhsh. Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions. *ACM Computing Surveys (CSUR)*, 51(1):1–40, 2018. doi: [10.1145/3148148](https://doi.org/10.1145/3148148) 3
- [11] S. Di Bartolomeo, Y. Zhang, F. Sheng, and C. Dunne. Sequence braiding: Visual overviews of temporal event sequences and attributes. *IEEE transactions on visualization and computer graphics*, 27(2):1353–1363, 2020. doi: [10.1109/TVCG.2020.3030442](https://doi.org/10.1109/TVCG.2020.3030442) 2
- [12] E. Di Giacomo, W. Didimo, G. Liotta, F. Montecchiani, and A. Tappini. Storyline visualizations with ubiquitous actors. In *International Symposium on Graph Drawing and Network Visualization*, pp. 324–332. Springer, 2020. doi: [10.1007/978-3-030-68766-3_25](https://doi.org/10.1007/978-3-030-68766-3_25) 2
- [13] V. Dibia. Lida: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 113–126, 2023. doi: [10.18653/v1/2023.acl-demo.11](https://doi.org/10.18653/v1/2023.acl-demo.11) 2
- [14] T. Dobson, P. Michura, S. Ruecker, M. Brown, and O. Rodriguez. Interactive visualizations of plot in fiction. *Visible Language*, 45(3):169–191, 2011. 2, 3
- [15] K. Elkins. *The shapes of stories: sentiment analysis for narrative*. Cambridge University Press, 2022. doi: [10.1017/9781009270403](https://doi.org/10.1017/9781009270403) 3
- [16] L. Frermann, J. Li, S. Khanehzar, and G. Mikolajczak. Conflicts, villains, resolutions: Towards models of narrative media framing. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8712–8732, 2023. doi: [10.18653/v1/2023.acl-long.486](https://doi.org/10.18653/v1/2023.acl-long.486) 3
- [17] G. Freytag. *Technique of the drama: An exposition of dramatic composition and art*. S. Griggs, 1895. 2
- [18] M. Grunde-McLaughlin, M. S. Lam, R. Krishna, D. S. Weld, and J. Heer. Designing llm chains by adapting techniques from crowdsourcing workflows. *ACM Trans. Comput.-Hum. Interact.*, 2025. doi: [10.1145/3716134](https://doi.org/10.1145/3716134) 3, 4
- [19] Y. Hassan-Montero and V. Herrero-Solana. Improving tag-clouds as visual information retrieval interfaces. *arXiv preprint arXiv:2401.04947*, 2024. doi: [10.48550/arXiv.2401.04947](https://doi.org/10.48550/arXiv.2401.04947) 1
- [20] M. N. Hoque, B. Ghai, K. Kraus, and N. Elmquist. Portrayal: Leveraging nlp and visualization for analyzing fictional characters. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, pp. 74–94, 2023. doi: [10.1145/3563657.3596000](https://doi.org/10.1145/3563657.3596000) 1, 2, 3
- [21] N. House and A. Johnston. Clover connections: Visualising character dynamics in novels for non-experts. In *Proceedings of the 35th Australian Computer-Human Interaction Conference*, pp. 191–201, 2023. doi: [10.1145/3638380.3638384](https://doi.org/10.1145/3638380.3638384) 1, 2, 3
- [22] Y. Huang, J. Song, Z. Wang, S. Zhao, H. Chen, F. Juefei-Xu, and L. Ma. Look before you leap: An exploratory study of uncertainty measurement for large language models. *IEEE Trans. Softw. Eng.*, 51(2):413–429, 17 pages, 2025. doi: [10.1109/TSE.2024.3519464](https://doi.org/10.1109/TSE.2024.3519464) 9
- [23] G. Hulstein, V. Peña-Araya, and A. Bezerianos. Geo-storylines: Integrating maps into storyline visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):994–1004, 2022. doi: [10.1109/TVCG.2022.3209480](https://doi.org/10.1109/TVCG.2022.3209480) 2
- [24] B. Jaipersaud, Z. Zhu, F. Rudzicz, and E. Creager. Show, don’t tell: Uncovering implicit character portrayal using llms. *arXiv preprint arXiv:2412.04576*, 2024. doi: [10.48550/arXiv.2412.04576](https://doi.org/10.48550/arXiv.2412.04576) 1, 2
- [25] M. John, M. Baumann, D. Schuetz, S. Koch, and T. Ertl. A visual approach for the comparative analysis of character networks in narrative texts. In *IEEE Pacific Visualization Symposium (PacificVis)*, pp. 247–256, 2019. doi: [10.1109/PacificVis.2019.00037](https://doi.org/10.1109/PacificVis.2019.00037) 6
- [26] O. Kaser and D. Lemire. Tag-cloud drawing: Algorithms for cloud visualization. *arXiv preprint cs/0703109*, 2007. doi: [10.48550/arXiv.cs/0703109](https://doi.org/10.48550/arXiv.cs/0703109) 1
- [27] D. A. Keim and D. Oelke. Literature fingerprinting: A new method for visual literary analysis. In *IEEE Symposium on Visual Analytics Science and Technology*, pp. 115–122, 2007. doi: [10.1109/VAST.2007.4389004](https://doi.org/10.1109/VAST.2007.4389004) 2
- [28] N. W. Kim, B. Bach, H. Im, S. Schriber, M. Gross, and H. Pfister. Visualizing nonlinear narratives with story curves. *IEEE transactions on visualization and computer graphics*, 24(1):595–604, 2017. doi: [10.1109/TVCG.2017.2744118](https://doi.org/10.1109/TVCG.2017.2744118) 1, 2, 5
- [29] Y. Kim, Y. Chang, M. Karpinska, A. Garimella, V. Manjunatha, K. Lo, T. Goyal, and M. Iyyer. Fables: Evaluating faithfulness and content selection in book-length summarization. In *Proceedings of the 1st Conference on Language Modeling (COLM)*, 2024. doi: [10.48550/arXiv.2404.01261](https://doi.org/10.48550/arXiv.2404.01261) 1, 2, 3
- [30] V. Labatut and X. Bost. Extraction and analysis of fictional character networks: A survey. *ACM Computing Surveys (CSUR)*, 52(5):1–40, 2019. doi: [10.1145/3344548](https://doi.org/10.1145/3344548) 6
- [31] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, article no. 793, 16 pages, 2020. doi: [10.5555/3495724.3496517](https://doi.org/10.5555/3495724.3496517) 1, 3, 9
- [32] S. Liu, Y. Wu, E. Wei, M. Liu, and Y. Liu. Storyflow: Tracking the evolution of stories. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2436–2445, 2013. doi: [10.1109/TVCG.2013.196](https://doi.org/10.1109/TVCG.2013.196) 1, 2, 5
- [33] J. Lu, B. Pan, J. Chen, Y. Feng, J. Hu, Y. Peng, and W. Chen. Agentlens: Visual analysis for agent behaviors in llm-based autonomous systems. *IEEE Transactions on Visualization and Computer Graphics*, 2024. doi: [10.1109/TVCG.2024.3394053](https://doi.org/10.1109/TVCG.2024.3394053) 2
- [34] D. Masson, Z. Zhao, and F. Chevalier. Visual writing: Writing by manipulating visual representations of stories. *arXiv preprint arXiv:2410.07486*, 2024. doi: [10.48550/arXiv.2410.07486](https://doi.org/10.48550/arXiv.2410.07486) 9
- [35] N. McCurdy, J. Lein, K. Coles, and M. Meyer. Poemage: Visualizing the sonic topology of a poem. *IEEE transactions on visualization and computer graphics*, 22(1):439–448, 2015. doi: [10.1109/TVCG.2015.2467811](https://doi.org/10.1109/TVCG.2015.2467811) 2, 3
- [36] G. Michel, E. V. Epure, R. Hennequin, and C. Cerisara. Improving quotation attribution with fictional character embeddings. In *Findings of*

- the Association for Computational Linguistics: EMNLP 2024*, pp. 12723–12735, 2024. doi: [10.18653/v1/2024.findings-emnlp.744](https://doi.org/10.18653/v1/2024.findings-emnlp.744) 4
- [37] S. Min and J. Park. Modeling narrative structure and dynamics with networks, sentiment analysis, and topic modeling. *PLoS one*, 14(12):e0226025, 2019. doi: [10.1371/journal.pone.0226025](https://doi.org/10.1371/journal.pone.0226025) 1, 2, 3, 6
- [38] F. Moretti. *Distant reading*. Verso Books, 2013. doi: [10.1093/lle/fqu010](https://doi.org/10.1093/lle/fqu010) 9
- [39] M. E. Mortenson. *Mathematics for computer graphics applications*. Industrial Press Inc., 1999. doi: [10.5555/520335](https://doi.org/10.5555/520335) 6
- [40] R. Munroe. Movie narrative charts. <https://xkcd.com/657/>, December 2009. 2, 5
- [41] A. Narechania, A. Srinivasan, and J. Stasko. NL4dv: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):369–379, 2020. doi: [10.1109/TVCG.2020.3030378](https://doi.org/10.1109/TVCG.2020.3030378) 2, 9
- [42] D. Oelke, D. Kokkinakis, and M. Malm. Advanced visual analytics methods for literature analysis. In *Proceedings of the 6th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pp. 35–44, 2012. doi: [10.5555/2390357.2390364](https://doi.org/10.5555/2390357.2390364) 2
- [43] T. Otake, S. Yokoi, N. Inoue, R. Takahashi, T. Kurabayashi, and K. Inui. Modeling event salience in narratives via barthes' cardinal functions. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 1784–1794, 2020. doi: [10.18653/v1/2020.coling-main.160](https://doi.org/10.18653/v1/2020.coling-main.160) 3
- [44] K. Padia, K. H. Bandara, and C. G. Healey. A system for generating storyline visualizations using hierarchical task network planning. *Computers & Graphics*, 78:64–75, 2019. doi: [10.1016/j.cag.2018.11.004](https://doi.org/10.1016/j.cag.2018.11.004) 2
- [45] V. Peña-Araya, T. Xue, E. Pietriga, L. Amsaleg, and A. Bezerianos. Hyperstorylines: Interactively untangling dynamic hypergraphs. *Information Visualization*, 21(1):38–62, 2022. doi: [10.1177/14738716211045007](https://doi.org/10.1177/14738716211045007) 2
- [46] A. Piper and S. Bagga. Using large language models for understanding narrative discourse. In *Proceedings of the The 6th Workshop on Narrative Understanding*, pp. 37–46, 2024. doi: [10.18653/v1/2024.wnu-1.4](https://doi.org/10.18653/v1/2024.wnu-1.4) 2
- [47] A. Piper, M. Xu, and D. Ruths. The social lives of literary characters: Combining citizen science and language models to understand narrative social networks. In *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, pp. 472–482, 2024. doi: [10.18653/v1/2024.nlp4dh-1.45](https://doi.org/10.18653/v1/2024.nlp4dh-1.45) 1, 2
- [48] L. Qiang and C. Bingjie. Storycake: A hierarchical plot visualization method for storytelling in polar coordinates. In *International Conference on Cyberworlds (CW)*, pp. 211–218. IEEE, 2016. doi: [10.1109/CW.2016.43](https://doi.org/10.1109/CW.2016.43) 2
- [49] H. Rashkin, A. Celikyilmaz, Y. Choi, and J. Gao. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4274–4295, 2020. doi: [10.18653/v1/2020.emnlp-main.349](https://doi.org/10.18653/v1/2020.emnlp-main.349) 4
- [50] S. Rebora et al. Sentiment analysis in literary studies. a critical survey. *Digital Humanities Quarterly*, 17(2):1–17, 2023. 3
- [51] H. Schwan, J. Jacke, R. Kleymann, J.-E. Stange, and M. Dörk. Narrelations—visualizing narrative levels and their correlations with temporal phenomena. *DHQ: Digital Humanities Quarterly*, 13(3), 2019. 2
- [52] O. Scrivner and J. Davis. Interactive text mining suite: Data visualization for literary studies. In *CDH@ TLT*, pp. 29–38, 2017. 2, 3
- [53] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE transactions on visualization and computer graphics*, 16(6):1139–1148, 2010. doi: [10.1109/TVCG.2010.179](https://doi.org/10.1109/TVCG.2010.179) 1, 2
- [54] J. Shen, J. Mire, H. W. Park, C. Breazeal, and M. Sap. Heart-felt narratives: Tracing empathy and narrative style in personal stories with llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1026–1046, 2024. doi: [10.18653/v1/2024.emnlp-main.59](https://doi.org/10.18653/v1/2024.emnlp-main.59) 2, 3
- [55] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings IEEE Symposium on Visual Languages*, pp. 336–343, 1996. doi: [10.1109/VL.1996.545307](https://doi.org/10.1109/VL.1996.545307) 5
- [56] J. D. Spence. *The memory palace of Matteo Ricci*. Penguin, 1985. doi: [10.2307/2056104](https://doi.org/10.2307/2056104) 9
- [57] M. Steen. Co-design as a process of joint inquiry and imagination. *Design issues*, 29(2):16–28, 2013. doi: [10.1162/DESI_a_00207](https://doi.org/10.1162/DESI_a_00207) 2
- [58] Y. Tanahashi and K.-L. Ma. Design considerations for optimizing storyline visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2679–2688, 2012. doi: [10.1109/TVCG.2012.212](https://doi.org/10.1109/TVCG.2012.212) 2
- [59] T. Tang, R. Li, X. Wu, S. Liu, J. Knittel, S. Koch, T. Ertl, L. Yu, P. Ren, and Y. Wu. Plotthread: Creating expressive storyline visualizations using reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):294–303, 2020. doi: [10.1109/TVCG.2020.3030467](https://doi.org/10.1109/TVCG.2020.3030467) 2, 5
- [60] T. Tang, S. Rubab, J. Lai, W. Cui, L. Yu, and Y. Wu. istoryline: Effective convergence to hand-drawn storylines. *IEEE transactions on visualization and computer graphics*, 25(1):769–778, 2018. doi: [10.1109/TVCG.2018.2864899](https://doi.org/10.1109/TVCG.2018.2864899) 2, 5
- [61] P. Vaithilingam, E. L. Glassman, J. P. Inala, and C. Wang. Dynavis: Dynamically synthesized ui widgets for visualization editing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2024. doi: [10.1145/3613904.3642639](https://doi.org/10.1145/3613904.3642639) 2, 9
- [62] F. Van Ham, M. Wattenberg, and F. B. Viégas. Mapping text with phrase nets. *IEEE transactions on visualization and computer graphics*, 15(6):1169–1176, 2009. doi: [10.1109/TVCG.2009.165](https://doi.org/10.1109/TVCG.2009.165) 2
- [63] S. Venkatraman, N. I. Tripto, and D. Lee. Collabstory: Multi-lm collaborative story generation and authorship analysis. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3665–3679, 2025. doi: [10.18653/v1/2025.findings-naacl.203](https://doi.org/10.18653/v1/2025.findings-naacl.203) 7
- [64] T. Wang, J. He, and C. Xiong. Ragviz: Diagnose and visualize retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 320–327, 2024. doi: [10.18653/v1/2024.emnlp-demo.33](https://doi.org/10.18653/v1/2024.emnlp-demo.33) 9
- [65] S. Ware and R. Young. Modeling narrative conflict to generate interesting stories. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 6, pp. 210–215, 2010. doi: [10.1609/aaide.v6i1.12411](https://doi.org/10.1609/aaide.v6i1.12411) 3
- [66] K. Watson, S. S. Sohn, S. Schriber, M. Gross, C. M. Muniz, and M. Kapadia. Storyprint: An interactive visualization of stories. In *Proceedings of the 24th international conference on intelligent user interfaces*, pp. 303–311, 2019. doi: [10.1145/3301275.3302302](https://doi.org/10.1145/3301275.3302302) 1, 2, 3, 5
- [67] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, article no. 1800, 14 pages, 2022. doi: [10.5555/3600270.3602070](https://doi.org/10.5555/3600270.3602070) 3
- [68] A. Wu, Y. Wang, X. Shu, D. Moritz, W. Cui, H. Zhang, D. Zhang, and H. Qu. Ai4vis: Survey on artificial intelligence approaches for data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):5049–5070, 2021. doi: [10.1109/TVCG.2021.3099002](https://doi.org/10.1109/TVCG.2021.3099002) 2
- [69] W. Yang, M. Liu, Z. Wang, and S. Liu. Foundation models meet visualizations: Challenges and opportunities. *Computational Visual Media*, pp. 1–26, 2024. doi: [10.1007/s41095-023-0393-x](https://doi.org/10.1007/s41095-023-0393-x) 2
- [70] Z. Yang, Z. Liu, J. Zhang, C. Lu, J. Tai, T. Zhong, Y. Li, S. Zhao, T. Yao, Q. Liu, et al. Analyzing nobel prize literature with large language models. *arXiv preprint arXiv:2410.18142*, 2024. doi: [10.48550/arXiv.2410.18142](https://doi.org/10.48550/arXiv.2410.18142) 9
- [71] L. Ye, L. Wang, S. Ruan, Y. Meng, Y. Wang, W. Chen, and Z. Zhou. Storyexplorer: A visualization framework for storyline generation of textual narratives. *arXiv preprint arXiv:2411.05435*, 2024. doi: [10.48550/arXiv.2411.05435](https://doi.org/10.48550/arXiv.2411.05435) 1, 2
- [72] T. Zamenopoulos and K. Alexiou. *Co-design as collaborative research*. Bristol University/AHRC Connected Communities Programme, 2018. 2
- [73] A. Zehe, L. Konle, L. K. Dümpelmann, E. Gius, A. Hotho, F. Jannidis, L. Kaufmann, M. Krug, F. Puppe, N. Reiter, et al. Detecting scenes in fiction: A new segmentation task. In *Proceedings of the 16th conference of the European chapter of the association for computational linguistics: Main volume*, pp. 3167–3177, 2021. doi: [10.18653/v1/2021.eacl-main.276](https://doi.org/10.18653/v1/2021.eacl-main.276) 3, 5
- [74] X. Zhang, M. Chen, and J. May. Salience-aware event chain modeling for narrative understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 1418–1428, 2021. doi: [10.18653/v1/2021.emnlp-main.107](https://doi.org/10.18653/v1/2021.emnlp-main.107) 3