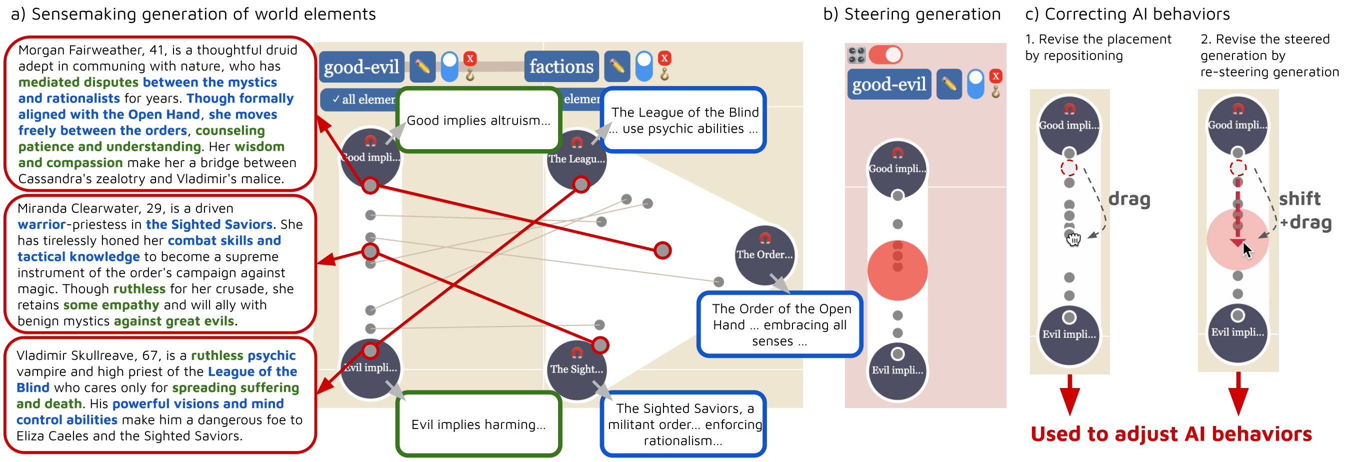
Patchview: LLM-Powered Worldbuilding with Generative Dust and Magnet Visualization

John Joon Young Chung
jchung@midjourney.com
Midjourney
San Francisco, CA, USA

Max Kreminski
mkreminski@midjourney.com
Midjourney
San Francisco, CA, USA



Morgan Fairweather, 41, is a thoughtful druid adept in communing with nature, who has **mediated disputes between the mystics and rationalists** for years. **Though formally aligned with the Open Hand, she moves freely between the orders**, **counseling patience and understanding**. Her **wisdom and compassion** make her a bridge between Cassandra's zealotry and Vladimir's malice.

Miranda Clearwater, 29, is a driven **warrior**-priestess in **the Sighted Saviors**. She has tirelessly honed her **combat skills and tactical knowledge** to become a supreme instrument of the order's campaign against magic. Though **ruthless** for her crusade, she retains **some empathy** and will ally with benign mystics **against great evils**.

Vladimir Skullreave, 67, is a **ruthless psychic** vampire and high priest of the **League of the Blind** who cares only for **spreading suffering and death**. His **powerful visions and mind control abilities** make him a dangerous foe to Eliza Caeles and the Sighted Saviors.

Good implies altruism...

Evil implies harming...

The League of the Blind ... use psychic abilities ...

The Order of the Open Hand ... embracing all senses ...

The Sighted Saviors, a militant order... enforcing rationalism...

Figure 1: For generating story world elements with LLMs, Patchview leverages a dust and magnet visual representation to help users a) sensemake, b) steer, and c) correct LLM generation. a) In dust and magnet visual representation, user-defined concepts serve as "magnets" (larger dark circles), attracting "dust particle" elements (smaller light circles) more strongly if an element is more relevant to the concept. Note that we only show partial excerpts of magnets due to the limited space. b) By placing a red marker between magnets, the user can steer the generation with a mix of different concepts. c) When LLM behaviors (steering, recognition) do not align with the user's perception, the user can correct them simply by moving the element, either 1) revising the element position or 2) re-steering generation to rewrite the element. The corrected placement will be fed into the LLM pipeline as an example to improve future steering and recognition.

ABSTRACT

Large language models (LLMs) can help writers build story worlds by generating world elements, such as factions, characters, and locations. However, making sense of many generated elements can be overwhelming. Moreover, if the user wants to precisely control aspects of generated elements that are difficult to specify verbally, prompting alone may be insufficient. We introduce Patchview, a customizable LLM-powered system that visually aids worldbuilding by allowing users to interact with story concepts and elements through the physical metaphor of magnets and dust. Elements in Patchview are visually dragged closer to concepts with high relevance, facilitating sensemaking. The user can also steer the generation with verbally elusive concepts by indicating the desired position of the element between concepts. When the user disagrees with the LLM's visualization and generation, they can correct those by repositioning the element. These corrections can be used to align the LLM's future behaviors to the user's perception. With a user study, we show that Patchview supports the sensemaking of world elements and steering of element generation, facilitating exploration during the worldbuilding process. Patchview provides insights on how customizable visual representation can help sensemake, steer, and align generative AI model behaviors with the user's intentions.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; *Visualization systems and tools*; • **Computing methodologies** → **Natural language generation**.

KEYWORDS

worldbuilding, large language models, dust and magnet visualization

UIST '24, October 13–16, 2024, Pittsburgh, PA, USA

ACM ISBN 979-8-4007-0628-8/24/10
https://doi.org/10.1145/3654777.3676352

ACM Reference Format:
John Joon Young Chung and Max Kreminski. 2024. Patchview: LLM-Powered Worldbuilding with Generative Dust and Magnet Visualization. In *The 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24), October 13–16, 2024, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3654777.3676352

1 INTRODUCTION

Rapid progress in the development of generative large language models (LLMs) [8, 14] has recently led to the introduction of numerous LLM-based tools for storywriting [1, 17, 34, 61]. While many of these tools aim to generate text for direct inclusion in a finished story, opportunities also lie in using LLMs to support other aspects of the writing process, such as worldbuilding. Worldbuilding—the act of constructing a coherent fictional world [24]—establishes a setting from which a variety of stories could arise. It requires writers to envision myriad aspects of a world, from abstract values (e.g., religion, ideology) to more specific elements, such as factions, characters, places, or props. As worldbuilding involves creating many different world elements, writers often put a lot of time and effort into it. Generative LLMs could be used to support this process, for instance by producing additional world elements that fit into the established setting or even inspire writers to expand their conception of the world they are creating.

However, when generating many world elements with LLMs, understanding their overall landscape can be challenging. That is, to unfold a story where different elements interact with each other, the writers would need to have a holistic view regarding what kind of attributes and values those elements hold. As LLMs can quickly add many elements to the world, it can be challenging for a writer to understand the rapidly growing world. Moreover, once the writer has understood existing world elements, they might want to generate a specific type of world elements. One way to guide LLMs for such a purpose would be to write natural language prompts. However, if the writer wants to express verbally elusive or ambiguous concepts, writing natural language prompts can either be cumbersome [49] or have limited expressivity [15].

To support sensemaking and steering of world element generation, we propose *generative dust and magnet* (GD&M) visualization, which adapts dust and magnet visual representation [11, 60] to the use of generative models. GD&M visualizes elements as "particles of iron dust" which are attracted to different concepts, or "magnets", based on their relevance to each concept (i.e., placed more closely if more relevant) (Figure 1a). This approach supports flexible visualization of semantic association between concepts and elements with an arbitrary number of concepts, by leveraging intermediate spaces between extreme "anchoring" concepts. Moreover, spaces between concepts can be used for guiding generation, even allowing expression of ambiguity between concepts (Figure 1b). When the user disagrees with steered generation and recognition results, the user can straightforwardly correct them by simply moving dust particles to other positions (Figure 1c). With repositioning, the user can indicate the generated element's correct placement (Figure 1c1) or command AI to revise the element to fit in the new position (Figure 1c2). These corrections can feed back into the AI as examples of the user's perspective for future steering and recognition.

We instantiated these interactions in Patchview, an LLM-powered story worldbuilding tool. Via a user study with eight hobbyists and one professional in worldbuilding, we show that Patchview allows users to quickly understand the landscape of elements within the story world. Moreover, we find that visual steering of LLM generation could function as an intuitive alternative to natural language prompting, allowing users to express nuances that are difficult to articulate with natural language. While participants found the interaction of correcting AI results on the visual space straightforward, user-provided corrections did not have a significant impact on aligning AI behaviors to user intent. However, participants found the tool overall helpful for worldbuilding, flexibly creating worlds that they found to be of interest. We conclude with discussions on visual representations for interacting with generative AI; using worlds from Patchview for story writing; technical alternatives for prompt engineering and closed models; and limitations.

In summary, this work has three main contributions:

(1) Generative Dust and Magnet (GD&M) visualization, a novel visual interaction approach to make sense of, guide, and intervene in the use of generative AI models.
(2) Patchview, an LLM-powered tool that supports story world element creation with GD&M.
(3) An evaluation that shows how Patchview supports sensemaking and steering of LLM outputs while revealing limitations in aligning LLM behaviors to the user's perspective.

2 RELATED WORK

2.1 Worldbuilding

Worldbuilding is a process of architecting fictional worlds that can be cornerstones of narrative fiction [24]. It considers various aspects, such as places, characters, or even cultures, and well-constructed worlds add believability to the stemming narrative stories. A well-built story world also entertains readers, as readers build out the conception of a coherent world out of various stemming stories [21, 39]. With a story world, readers can also participate in active consumer experiences, such as creating fan fiction and even transforming the canon world into alternative worlds [21, 46]. While worldbuilding can be a complex process with many aspects to consider, there have been practical frameworks and structures that practitioners use. Practitioners would likely first focus on the frameworks of the world, which can include scope (geography of the world), sequence (temporal history of the world), and perspective (from whom the world is explained) [24]. Under such frameworks, practitioners would create structures of the world. Governance (e.g., government presence, rule of law), economics (e.g., economic strength, wealth distribution), social relations (e.g., class, race, and ethnic relations), cultural influences (e.g., religious influences, cultural influences), and character alignments (e.g., good-evil, lawful-chaotic) [24] are some examples of world structures. Then, around frameworks and structures, practitioners would create catalogs of fictional worlds, or elements of the world, such as characters, places, props, and events [24]. Worldbuilding can be done either solely (e.g., Tolkien's world of Lord of Rings) or collectively (e.g., Marvel Universe), and commercial projects often tend to be collaborative as doing creative work can be overloading to an individual. WorldSmith is one of the few AI-powered tools to support worldbuilding but focuses on creating visual aspects of the world [20]. In this work, which focuses on supporting world element creation, we introduce an AI-powered worldbuilding tool that co-constructs the story world with users by generating new world elements based on what the user has. Specifically, we facilitate the use of AI models by incorporating visual means for users to sensemake and control world element generation.

2.2 AI-Powered Story Writing

With advancing AI technologies, researchers and practitioners have developed many tools to support story writing. For example, TaleStream supports story ideation by showing potentially inspiring story tropes [13]. Loose Ends is a rule-based mixed-initiative AI system that allows users to explore plot threads with some constraints [31]. Portrayal leveraged NLP and visualizations to help writers analyze characters in their stories [25]. LLM's advanced generative capabilities introduced tools that suggest texts that users can incorporate into their writing [9, 19, 35]. Researchers investigated diverse interactions for such tools, from allowing distinct suggestion operations [61] to incorporating multimodality [23], hierarchical generation [40], and sketching inputs [17]. With these rapidly advancing capabilities, researchers also studied story writer's expectations for these technologies, such as what they would take as a benefit and what they want [7, 22, 29]. Lee et al. [34] reflected on the design space of writing tools through a literature survey. LLMs also enabled story applications where the story is generated with minimal writer interventions, directly facing the audience [44, 55]. While many LLM-powered story writing tools focused on supporting prose text writing, some focused on other types of support. For example, CALYPSO leverages LLMs to provide support to dungeon masters when playing Dungeons & Dragons [63]. In a similar vein, we design Patchview to provide LLM-powered support in worldbuilding, which is other than writing story texts themselves.

2.3 Visually Interacting with Generative AI

While natural language-based interfaces (e.g., prompts, chat) have been widely used for generative AI models, many previous systems used visual interactions to complement natural language interactions. Some tools leverage node-based input interactions to control generation, such as chaining subtasks [4, 6, 30, 58]. Among them, ChainForge [6] and Cells-Generators-Lenses framework [30] also allowed evaluation of generated results with visualization nodes. While these tools allowed flexible control, steering and evaluation happened in separated interfaces, leading to visual complexity. As another type, Scenescape [51] and Graphologue [28] leveraged graph and tree visualization to help understand complex information. While the user can steer further generations by clicking on the node which the user is willing to learn more details about, these focus more on presenting information than allowing flexible steering. Some tools allow steering or evaluation of multiple generation results on dimensional spaces of attributes, represented in either sliders [38], mixed color spaces [15], temporal line drawing [17], or scatter plots organized in grids [50]. Among them, TaleBrush [17] and Luminate [50] tied steering and evaluation interactions on a single visual representation, minimizing clutters. TaleBrush considers a continuous dimensional scale but on a fixed attribute. On the other hand, Luminate allows arbitrary dimensional attributes but only with categorical/ordinal attributes. Moreover, all aforementioned tools do not allow users to correct AI behaviors when AI's steered generation and recognition results do not align with the user's thoughts. Patchview extends previous work by allowing generation steering, evaluation, and user corrections on an integrated single visual representation with the flexibility of allowing continuous scales of any arbitrary concepts of interest.

3 GENERATIVE DUST AND MAGNET

Patchview's central design metaphor—*generative dust and magnet* (GD&M)—leverages a dust and magnet (D&M) visual representation [60] to facilitate interaction with generative AI models. In this section, we first describe settings where GD&M can be helpful (Section 3.1). Then, we describe the original D&M visualization and how we translate its components for use with generative models (Section 3.2). Finally, we describe specific GD&M interactions that close gaps in the interactive alignment of AI models: evaluation support, specification alignment, and process alignment [52] (Section 3.3).

3.1 Need for Generative Dust and Magnet

Interaction with generative AI might benefit from a wide range of different interaction approaches in different settings. In general, we expect GD&M interaction to be most effective when **the user must generate many distinct units of output (e.g., storyworld elements) that vary along diverse and expressive conceptual dimensions**. Breaking this ideal setting down further, we arrive at a set of three conditions that typify good application domains for GD&M interaction.

First, the user must make use of generative models to gather a collection of many generated outputs. This imposes a need for **sensemaking** (**N1**), as understanding how outputs distribute along the user's conceptual dimensions of interest is difficult due to the large scale of generation.

Second, the user must have desires to create artifacts within their unique characteristics and values, which often occurs in artistic creation [16]. This imposes a need for **configurability** (**N2**), where behaviors of AI functions (e.g., generation and evaluation of generated results) consider the user's unique styles and interests.

Third, the user must need to express nuanced specifications that align generation with the user's specific intentions and facilitate exploration of subtly different options. This imposes a need for **expressivity** (**N3**) where the user can guide generation even with subtle intentions.

GD&M interaction would be ideal for user tasks with the above characteristics. Worldbuilding meets all of these conditions: the writer must create many world elements to fill out a unique and idiosyncratic world, and created elements can have nuanced differences between each other [24]. In the following sections, we describe how GD&M can fulfill the aforementioned needs.

3.2 From D&M Visualization to GD&M

Yi et al.'s original dust and magnet visualization represents individual data elements as "dust particles" while representing each variable for which data elements can possess different values as a "magnet". Both dust particles and magnets are rendered as glyphs

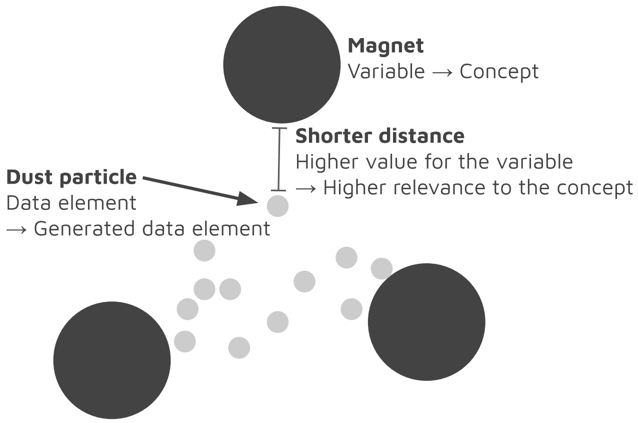


Figure 2: Compared to dust and magnet visualization, generative dust and magnet replaces data elements (dust particles) and variables (magnets) with generated data elements and concepts, respectively. In generative dust and magnet, the distance between a magnet and a dust particle indicates the intensity of relevance between them.

on a 2D plane; a data element with a particularly high value for a certain variable will be placed closer to the magnet representing that variable. This approach can facilitate the accessibility of understanding many multivariate data instances [60] while allowing users to identify notable patterns within a dataset [11].

We extend D&M visualization to an interface for generative models (Figure 2). Generative D&M replaces multivariate data elements with generated data elements in the output modality of a generative model (e.g., passages of text for LLMs, and images for text-to-image models). Accordingly, variables in the original D&M visualization translate to concepts that characterize the generated outputs (e.g., "positive sentiment" for texts, "pastel colors" for images). Under this translation, a generated element that is more strongly relevant to a specific concept is drawn closer to the magnet for the corresponding concept.

3.3 Specific GD&M Interactions

Several specific GD&M interactions are designed to meet user needs discussed in Section 3.1 (Figure 3). We organize these interactions in terms of how they support interactive alignment of AI models [52]. Extending challenges of the gulf of evaluation and execution [42], Terry et al. [52] emphasized three facets of interactive alignment of AI models: 1) evaluation support (**I1**), or users making sense of AI outputs; 2) specification alignment (**I2**), or users efficiently and reliably communicating their objectives to AI; and 3) process alignment (**I3**), or users verifying or controlling AI's execution process.

3.3.1 I1: Evaluation Support - User Configurable Dust and Magnet Visualization (N1, N2). To support users' sensemaking of many generated elements according to their concepts of interest, the user can add generated elements to the magnet space configured with concepts of the user's interest (Figure 3-I1). Then, an AI model measures the relevance of each element to different concepts and visualizes this information as the relative distance to those concepts (e.g., good characters being closer to the concept of "good" than to "evil"). With this support, the user can quickly get an overview of how generated elements are different from each other. GD&M further provides users with flexible configurability, as the user can add, remove, or even edit concepts. With this configurability (Figure 4a, b, and c), users can easily reassess the set of generated outputs in terms of the concepts that are most relevant to their current focus. Moreover, the user can adjust the layout of concepts (Figure 4d), aligning their visual presentation with how they want to think about these concepts and elements.

3.3.2 I2: Specification Alignment - Generation Specifications within Magnet Space (N2, N3). GD&M interaction also allows users to guide the generation of new elements by indicating the ideal placement of these elements within the visual-semantic magnet space defined by a set of user-configured concepts. That is, the user can place a marker on the magnet space to request the generation of elements

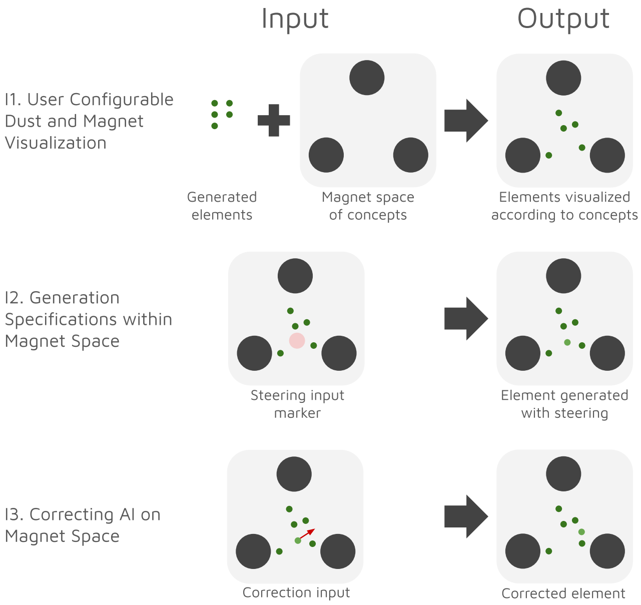

Input
Output
I1. User Configurable Dust and Magnet Visualization
Generated elements
Magnet space of concepts
Elements visualized according to concepts
I2. Generation Specifications within Magnet Space
Steering input marker
Element generated with steering
I3. Correcting AI on Magnet Space
Correction input
Corrected element

Figure 3: Input-Output schemes for GD&M Interactions

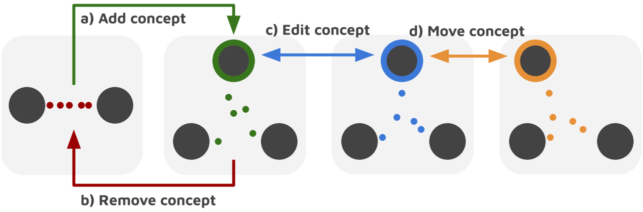


Figure 4: Configuration interactions for evaluation support in generative dust and magnet. As the user adds, removes, edits, and moves concepts according to how they want to organize elements and concepts, the positions of data elements get updated.

that would be placed near the specified marker (as in Figure 1b and 3-I2). For instance, if the user wants to generate a good character, in the magnet space between "good" and "evil," they can place the marker closer to "good." One benefit of this visual magnet space is that the user can express vague or ambiguous specifications in this continuous space between concepts (e.g., generating an array of characters that vary subtly along the "good"-"evil" spectrum).

3.3.3 I3: Process Alignment - Correcting AI on Magnet Space (N2). AI behaviors may not always align with user intent: for instance, the user might not agree with how the AI interprets concepts during generation and placement of generated elements. In such cases, the user can freely re-specify concepts to more accurately convey how they think about each concept (e.g., adding more details about what "good" means in a specific story world, Figure 4c). They can also leverage the magnet space itself to correct AI behavior, by simply moving a misplaced generated element to wherever the user thinks it should be in the magnet space (Figure 1c and 3-I3). Repositioning an element can convey two intentions: either 1) that the element's "correct" placement is in a new position (e.g., indicating that the character should sit in the middle of "good" and "evil" as in Figure 1c1) or 2) that the element should be revised to better fit the indicated position (e.g., request AI to rewrite the character description to sit in the middle of "good" and "evil", as in Figure 1c2). These corrections can then be used as examples to better align future generation and placement with user perception of concepts.

4 PATCHVIEW: INTERFACE AND TECHNICAL DETAILS

With GD&M, we built Patchview, an LLM-powered tool for world element creation (Figure 5). Specifically, Patchview supports sensemaking and steering of world element generation. To demonstrate the effectiveness of GD&M for sensemaking and steering, Patchview focuses on creating initial "seeds" of story world elements in two to three sentences. Afterward, users can develop details of these seed elements either by themselves or with the help of AI; the final rendering of seed elements into a more complete form is left to future work.

Patchview's user interface consists of the list module, which shows existing world elements as a list of notes (Figure 5b), and the view module, which organizes world elements via GD&M (Figure 5a). Note that Patchview leverages AI to generate specific world elements (e.g., characters, places) rather than generating frameworks or structures of the world (e.g., ideology, values). The user can manually specify frameworks and structures as open-ended text in notes.

We explain the envisioned usage pattern with a hypothetical user, Alex. Alex is a game scenario writer who is trying to design a story world for the new game her team is developing. To get help with the process, Alex decides to use Patchview.

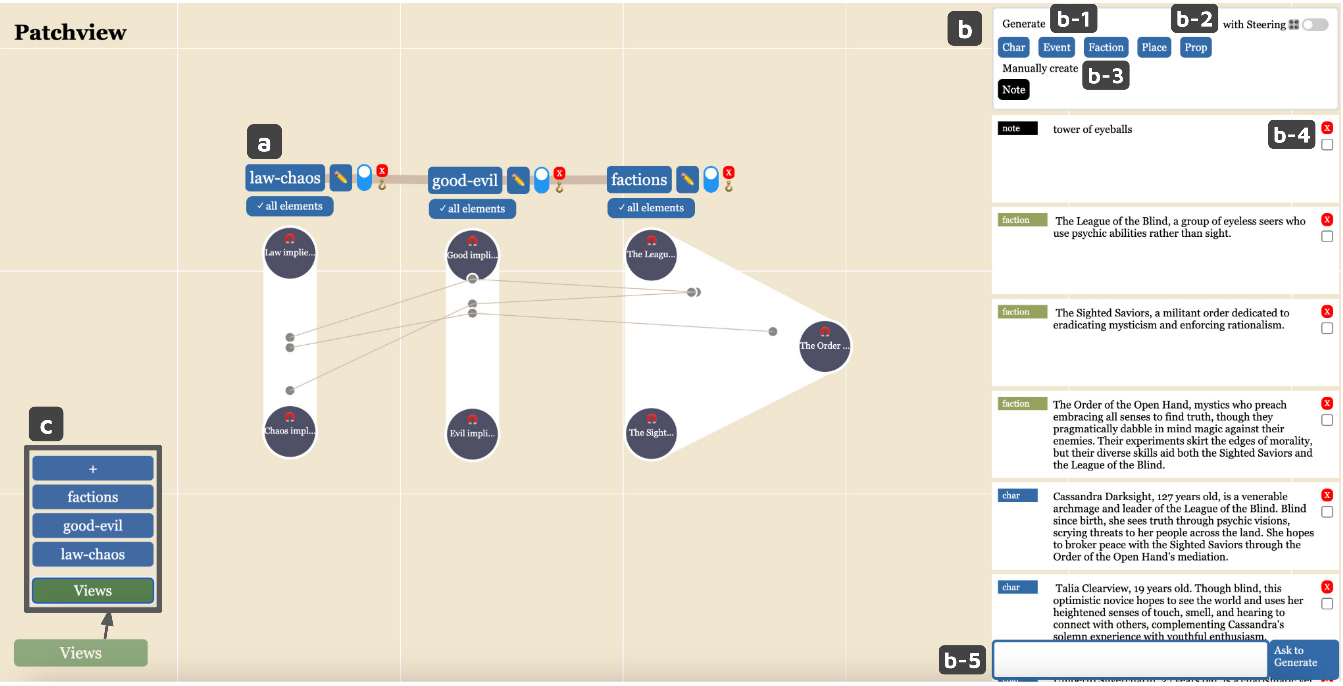
4.1 List Module

As Alex loads Patchview, she first sees the list module on the right. With this module, Alex can generate and create an initial set of world elements as textual notes. To set an initial high-level concept for the world, Alex decides to manually create a note by clicking the `note` button (Figure 5b-3) and modifying the text to "tower of eyeballs." Next, extrapolating from this high-level idea, Alex decides to generate factions in the story world. Generating world elements with AI is straightforward, as Alex can simply click on the button that corresponds to the type of the element that Alex wants to introduce (Figure 5b-1, thin solid line in Figure 6). When generating the new element, by default, Patchview will take all existing elements into context to ensure that the generated element is relevant to the current story world. If Alex wants the generation to consider only a subset of existing elements, she can select only those notes as context for generation. In case Alex wants to generate a more specific world element, Alex can also directly prompt the AI with natural language (Figure 5b-5, thick solid line in Figure 6). With this generation function, Alex first generates a few factions and then a handful of characters. As the number of world elements increases, Alex can organize them in the list by reordering them with dragging. However, at a certain point, she feels that the list is getting longer and becoming hard to understand.

4.2 View Module

4.2.1 Creating and Configuring View (I1). To make sense of this proliferation of world elements, Alex decides to use the view module to organize them. In Patchview, a *view* is a single GD&M visual-semantic space that organizes world elements in relation to a specific set of user-defined concepts. Alex can create a new view by first clicking the View button in the bottom left corner and then clicking the + button. The user can set the concept associated with each magnet in the view either by dropping existing notes into placeholder magnets (i.e., using elements as concepts, Figure 7a) or clicking the "Type in a new magnet" button that shows up when the user hovers their mouse close to the placeholder or existing concepts (Figure 7b). Once Alex configures the view with a set of concepts, she adds relevant elements as dust particles in the view by dragging and dropping elements from the list module to the target view. As Alex adds an element to the view, Patchview calculates its position within the view visualization space. Alex can also add multiple elements by first checking multiple of those in the list module. Note that Alex can add elements of different types in a single view, if they are relevant (e.g., putting a good character and a good faction under "good"-"evil" view). For concepts and elements in the view, Alex can read their full descriptions by hovering the cursor over them (Figure 8). When Alex selects added elements from the list module, to let her know where they are in the view, the tool highlights them on the visualization.

4.2.2 Correcting View Visualization (I3). For some elements added to the view, Alex does not agree with how Patchview positioned them. If Alex thinks the description of a particular concept is not detailed enough for the tool to grasp, she can modify it via the list module. Alternatively, she can edit the concept's definition text directly within the view module by hitting enter while hovering the cursor over the concept's magnet (similar to Figure 8, but with concepts). As Alex updates the concept, Patchview tries to reposition elements in relation to the concept. For elements still misplaced from Alex's perspective, Alex can manually adjust their positions by dragging them in the view (Figure 1c1). When positioning future



note — tower of eyeballs ☒ □

faction — The League of the Blind, a group of eyeless seers who use psychic abilities rather than sight. ☒ □

faction — The Sighted Saviors, a militant order dedicated to eradicating mysticism and enforcing rationalism. ☒ □

faction — The Order of the Open Hand, mystics who preach embracing all senses to find truth, though they pragmatically dabble in mind magic against their enemies. Their experiments skirt the edges of morality, but their diverse skills aid both the Sighted Saviors and the League of the Blind. ☒ □

char — Cassandra Darksight, 127 years old, is a venerable archmage and leader of the League of the Blind. Blind since birth, she sees truth through psychic visions, scrying threats to her people across the land. She hopes to broker peace with the Sighted Saviors through the Order of the Open Hand's mediation. ☒ □

char — Talia Clearview, 19 years old. Though blind, this optimistic novice hopes to see the world and uses her heightened senses of touch, smell, and hearing to connect with others, complementing Cassandra's solemn experience with youthful enthusiasm. ☒ □

Figure 5: Patchview interface. a) View module visualizes world elements in relation to concepts of the user's interest. Specific interactions are shown in Figure 1. b) List module lists world elements as notes (b-4). This module allows users to generate elements by clicking buttons for different element types (b-1) or by prompting an LLM with specific natural-language instructions (b-5). The user can steer generation with a view interface (as in Figure 1b) by entering the steering mode with a toggle switch (b-2). They can also manually create notes (b-3). c) The user can see a list of existing views by clicking the Views button and create a new view with the + button.

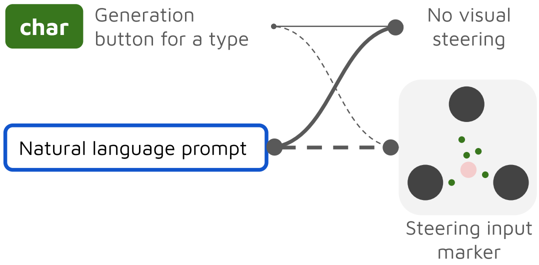
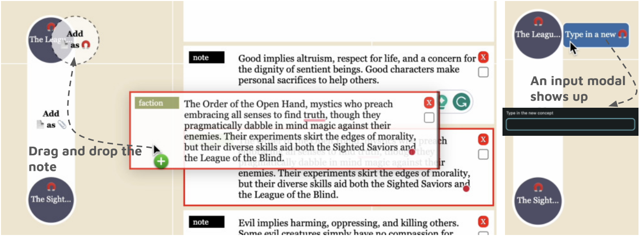


Figure 6: Possible inputs to generate elements.



note — Good implies altruism, respect for life, and a concern for the dignity of sentient beings. Good characters make personal sacrifices to help others.

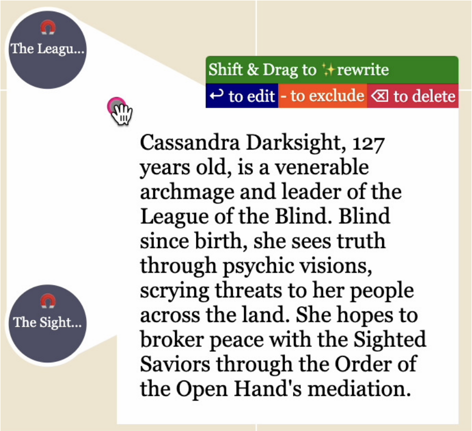
faction — The Order of the Open Hand, mystics who preach embracing all senses to find truth, though they pragmatically dabble in mind magic against their enemies. Their experiments skirt the edges of morality, but their diverse skills aid both the Sighted Saviors and the League of the Blind.

note — Evil implies harming, oppressing, and killing others. Some evil creatures simply have no compassion for

a) Adding an existing note as a concept

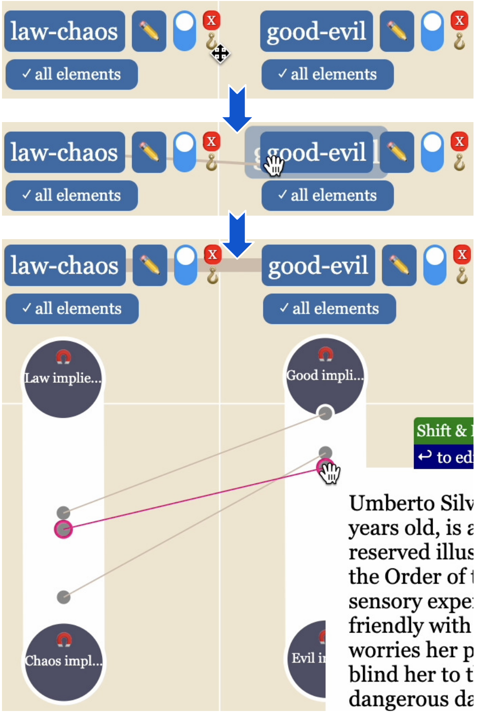
b) Typing in a new concept

Figure 7: Interactions to add a new concept to the view.



Cassandra Darksight, 127 years old, is a venerable archmage and leader of the League of the Blind. Blind since birth, she sees truth through psychic visions, scrying threats to her people across the land. She hopes to broker peace with the Sighted Saviors through the Order of the Open Hand's mediation.

Figure 8: The user can read each concept and element by hovering their mouse over them. While hovering the mouse, they can 1) edit them by hitting the enter key, 2) exclude them from the view by hitting the - key, and 3) delete them by hitting backspace. For elements, the user can re-steer the generation by dragging the element to a new position while holding the shift key; the LLM will then attempt to rewrite the element's text to better match the target position.



Umberto Silv
years old, is a
reserved illus
the Order of t
sensory expe
friendly with
worries her p
blind her to t
dangerous da

Figure 9: By dragging and dropping the anchor, the user can connect multiple views to have a better understanding of how elements distribute along those views. The same element is connected by the thin line, and for the highlighted element, the connecting line is also highlighted.

elements, Patchview leverages user-adjusted elements as examples to better follow the user's perspective.

4.2.3 Sensemaking Multiple Views (I1). Alex continues organizing world elements by creating multiple views. Alex organizes these views by dragging view names and concepts. At a certain point, Alex realizes that it is difficult to understand how characters are distributed along the conceptual dimensions of two views, good-evil and lawful-chaotic alignments. To have a better understanding, Alex anchors them together and Patchview connects the same elements in both views with a thin line (Figure 9). Note that only elements that exist in both views get connected. As Alex hovers her cursor over an element in one of the views, the identical element in another view and the thin connecting line between these elements are highlighted (Figure 9). After connecting these views, as each view is defined by only two concepts, Alex thinks that it would be easiest to make sense of these elements via a 2-dimensional visualization with two axes. For that, Alex can cross two views, and Patchview renders the view in 2D plane visualization instead of connecting elements with lines (Figure 10). Note that Patchview only visualizes elements that exist in both crossed views. As Alex adds more views, she continues to experiment with other visual arrangements, such as radar charts and parallel coordinate charts [18, 41] (Figure 11).

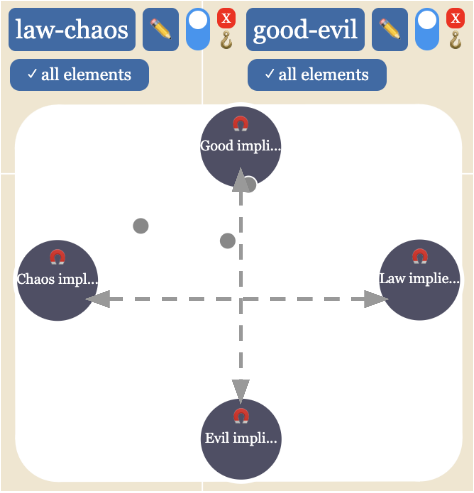
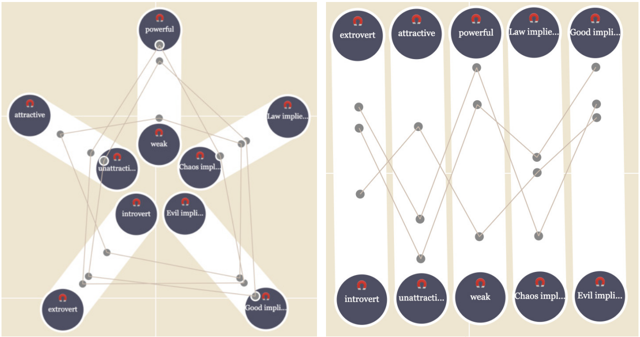


Figure 10: When two separate views are each defined by exactly two concepts, the user can cross these views into a 2D plane visualization. Analogically, this would correspond to putting dust particle elements under the simultaneous influence of two uniform magnet fields of concepts.



a. Radar chart b. Parallel coordinate chart

Figure 11: Multiple views can be flexibly organized to form a) a radar chart or b) a parallel coordinate chart.

4.2.4 Steering Generation in the View (I2). As Alex organizes world elements in the view, she finds herself wanting to add more characters to populate empty spaces within view visualizations. To steer the generation with this nuanced intention, Alex leverages a generative steering function on each view. Alex first clicks on the "with Steering" toggle switch at the top right to enter the generation mode. Then, Alex places the generation control right on the view space itself, indicating that Alex wants the newly generated element to be placed near the specified position (Figure 1b). To steer generation along multiple aspects, Alex can also place multiple generation controls on multiple views. After placing controls, Alex clicks on one of the type buttons or prompts the LLM (thin and thick dashed lines in Figure 6, respectively) to generate an element with the steering constraints applied. After generating the element, Patchview calculate its position in the view to place it on the view. Similar to how Patchview visualizes elements in the view, Patchview leverages elements added and edited by Alex to adjust its steering behavior to the user's perception of concepts.

4.2.5 Correcting Generation (I3). Sometimes, generated items do not perfectly align with Alex's specifications. To iterate on those, she can directly modify the text of the element or ask Patchview to rewrite it by dragging the element while holding the shift key (Figure 1c2). If Alex disagrees with how Patchview places a generated element, she can reposition it by dragging (Figure 1c1). For elements that Alex does not want to keep, Alex can either remove them from the view by hitting the minus key while hovering the cursor over the element or delete them from the view and the list by hitting the backspace key while hovering the cursor. Alex continues generating, editing, and organizing elements until she is done.

4.3 Technical Details and Implementation

We built Patchview as a web application with a React-based frontend and a Node-based backend server. We provide technical details on 1) mapping between the position of the element and the its relevance to the concepts and 2) LLM prompting.

4.3.1 Mapping Between Position and Weight. To enable visualization and visual steering on the view, Patchview needs to map the visualized position of an element to its relevance to considered concepts and vice versa. Here, we quantified the relevance of an element to the concept as *weight* values between 0 and 1. We first forced the placement of concepts to be convex, as the non-convex arrangement of concepts can bring in more complexities with those mappings. With the convex arrangement of concepts, we can compute the element position easily by weight-summing the concept positions with weights on those concepts. However, deciding weights from the element position is not trivial if there are more than three concepts, as a single position does not fall into one weight combination. That is, with more than three concepts, there can be more than three weights that need to be decided, but there would be only three equations with a 2D arrangement of concepts:

$$\begin{aligned} \Sigma_{i=1}^{n} w_i x_i &= x_e \\ \Sigma_{i=1}^{n} w_i y_i &= y_e \\ \Sigma_{i=1}^{n} w_i &= 1 \end{aligned} \tag{1}$$

x_i and y_i stand for the position of each concept, while x_e and y_e indicate the position of the element. w_i stands for the weight that needs to be inferred and n is the number of concepts.

Due to the above reason, with more than three concepts, we used the following heuristic to compute one weight combination: With all combinations of three concepts from all concepts, we first calculated weights for each combination. Then, we filtered out combinations with negative weights. After that, we calculated a weight for each concept by summing all weights from all the left combinations. Then, we finalized the weights by dividing each weight by the sum of all weights for all concepts. Note that with this approach, steering element generation with more than three concepts does have a limitation as not all possible weights are expressible with one geometric positioning of concepts. When two axes are crossed to form a 2D plane visualization (as in Figure 10), for each view, we first calculated the crossing point of the following two lines: 1) the line that passes through two concepts of the view and 2) the line that passes through the position of the element and is parallel to the line of two concepts from the other view. Then, as this calculated point is on the line that passes through two concepts of the view, we can calculate the weight from the equation 1.

4.3.2 LLM Prompting. To generate elements with steering inputs and recognize the relevance of elements to concepts, we prompted `claude-2.0` and `claude-instant-1.2` from Anthropic [5], respectively. We chose these models because they have shown better performance in creative writing contexts than leading alternatives [10].

Prompts for both generation and recognition began by introducing a set of existing world elements for context, as in Figure 12a. By default, all existing world elements were supplied as part of this context; the user could also select a subset of existing elements to ensure that only those elements would be provided as context.

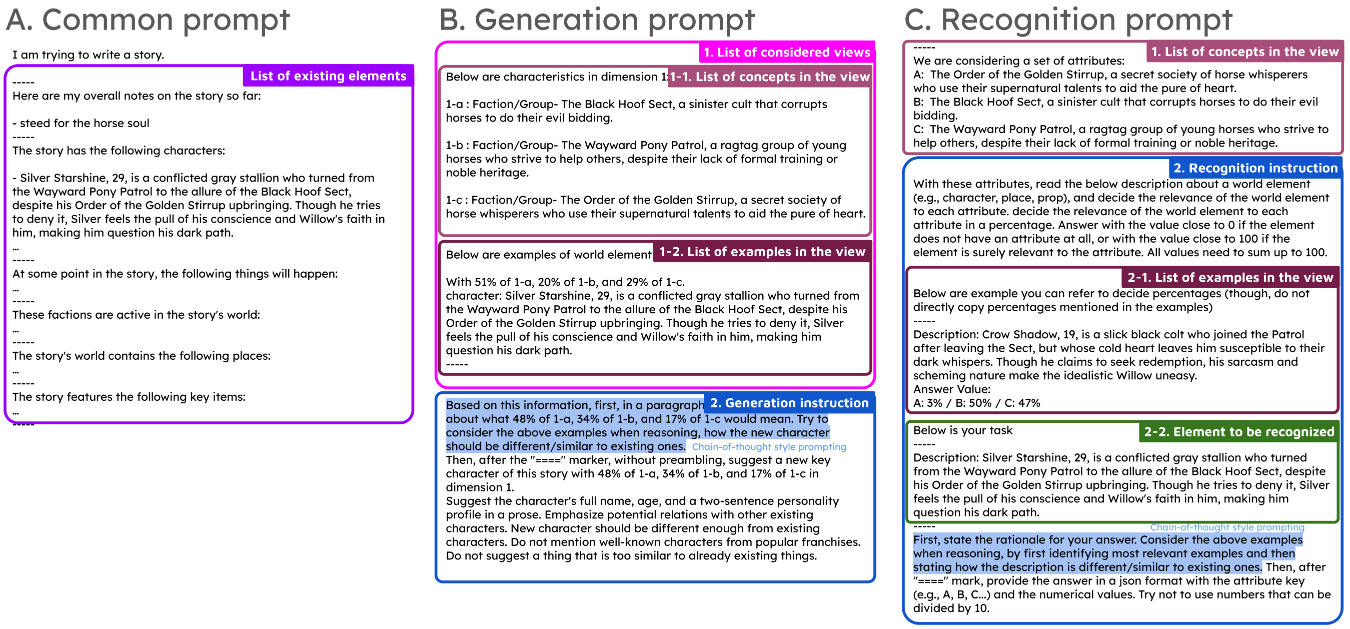
When generating new world elements without visual steering input, introductory context was followed directly by an instruction describing what kind of element to generate. When generating with visual steering input, we first appended concepts of all views (Figure 12b-1-1) and examples of how existing elements have relevance to those concepts (Figure 12b-1-2). These examples came from elements that the user has already placed in the view, including those repositioned by the user. Note that in the prompt, all views and concepts are phrased as "dimensions" and "characteristics", respectively. A chain-of-thought [57] style generation instruction prompt followed after (Figure 12b-2), which asked the LLM to first reason about how the element description should be written considering steering inputs and then to write the element description.

Recognition of concept relevance values takes place on a per-view basis, so a prompt for the recognition task included concept descriptions for only a particular considered view, as in Figure 12c. In the recognition prompts, introductory context (Figure 12a) and concept descriptions (Figure 12c-1) were followed by instructions about how to interpret numbers (Figure 12c-2), and then by examples of the correct performance of a recognition task for this set of concepts (Figure 12c-2-1). Because these examples were taken from past placements of elements into this specific view, information about how the user repositioned elements in this view were taken into account at this step. Finally, the world element to be analyzed was attached (Figure 12c-2-2), with the chain-of-thought [57] instruction that the LLM should provide reasoning before the result. The LLM was asked to provide recognition results in a JSON format with concept identifiers as keys and concept relevance weights as values.

5 USER STUDY

We conducted a user study on Patchview to learn if it supports sensemaking and steering of world element generation under the user's unique story world context. Specifically, we tried to answer the following research questions to determine if Patchview effectively supports the interactions described in Section 3.3.

- RQ1: Does Patchview help the user with sensemaking world elements? (I1)
- RQ2: Does Patchview help users express nuanced intentions with visual steering? (I2)
- RQ3: Does Patchview help users correct AI results and behaviors? (I3)


A. Common prompt
List of existing elements
B. Generation prompt
1. List of considered views
1-1. List of concepts in the view
1-2. List of examples in the view
2. Generation instruction
Chain-of-thought style prompting
C. Recognition prompt
1. List of concepts in the view
2. Recognition instruction
2-1. List of examples in the view
2-2. Element to be recognized
Chain-of-thought style prompting

```
I am trying to write a story.
-----
Here are my overall notes on the story so far:

- steed for the horse soul
-----
The story has the following characters:

- Silver Starshine, 29, is a conflicted gray stallion who turned from
the Wayward Pony Patrol to the allure of the Black Hoof Sect,
despite his Order of the Golden Stirrup upbringing. Though he tries
to deny it, Silver feels the pull of his conscience and Willow's faith in
him, making him question his dark path.
…
-----
At some point in the story, the following things will happen:
…
-----
These factions are active in the story's world:
…
-----
The story's world contains the following places:
…
-----
The story features the following key items:
…
-----
```

```
Below are characteristics in dimension 1:

1-a : Faction/Group- The Black Hoof Sect, a sinister cult that corrupts
horses to do their evil bidding.

1-b : Faction/Group- The Wayward Pony Patrol, a ragtag group of young
horses who strive to help others, despite their lack of formal training or
noble heritage.

1-c : Faction/Group- The Order of the Golden Stirrup, a secret society of
horse whisperers who use their supernatural talents to aid the pure of heart.

Below are examples of world element

With 51% of 1-a, 20% of 1-b, and 29% of 1-c.
character: Silver Starshine, 29, is a conflicted gray stallion who turned from
the Wayward Pony Patrol to the allure of the Black Hoof Sect, despite his
Order of the Golden Stirrup upbringing. Though he tries to deny it, Silver
feels the pull of his conscience and Willow's faith in him, making him
question his dark path.
-----

Based on this information, first, in a paragraph
about what 48% of 1-a, 34% of 1-b, and 17% of 1-c would mean. Try to
consider the above examples when reasoning, how the new character
should be different/similar to existing ones.
Then, after the "====" marker, without preambling, suggest a new key
character of this story with 48% of 1-a, 34% of 1-b, and 17% of 1-c in
dimension 1.
Suggest the character's full name, age, and a two-sentence personality
profile in a prose. Emphasize potential relations with other existing
characters. New character should be different enough from existing
characters. Do not mention well-known characters from popular franchises.
Do not suggest a thing that is too similar to already existing things.
```

```
-----
We are considering a set of attributes:
A:  The Order of the Golden Stirrup, a secret society of horse whisperers
who use their supernatural talents to aid the pure of heart.
B:  The Black Hoof Sect, a sinister cult that corrupts horses to do their evil
bidding.
C:  The Wayward Pony Patrol, a ragtag group of young horses who strive to
help others, despite their lack of formal training or noble heritage.

With these attributes, read the below description about a world element
(e.g., character, place, prop), and decide the relevance of the world element
to each attribute. decide the relevance of the world element to each
attribute in a percentage. Answer with the value close to 0 if the element
does not have an attribute at all, or with the value close to 100 if the
element is surely relevant to the attribute. All values need to sum up to 100.

Below are example you can refer to decide percentages (though, do not
directly copy percentages mentioned in the examples)
-----
Description: Crow Shadow, 19, is a slick black colt who joined the Patrol
after leaving the Sect, but whose cold heart leaves him susceptible to their
dark whispers. Though he claims to seek redemption, his sarcasm and
scheming nature make the idealistic Willow uneasy.
Answer Value:
A: 3% / B: 50% / C: 47%

Below is your task
-----
Description: Silver Starshine, 29, is a conflicted gray stallion who turned
from the Wayward Pony Patrol to the allure of the Black Hoof Sect, despite
his Order of the Golden Stirrup upbringing. Though he tries to deny it, Silver
feels the pull of his conscience and Willow's faith in him, making him
question his dark path.
-----
First, state the rationale for your answer. Consider the above examples
when reasoning, by first identifying most relevant examples and then
stating how the description is different/similar to existing ones. Then, after
"====" mark, provide the answer in a json format with the attribute key
(e.g., A, B, C...) and the numerical values. Try not to use numbers that can be
divided by 10.
```

Figure 12: Prompts used for Patchview.

Table 1: Participant backgrounds. AI Exp* stands for experience with generative AI technologies (e.g., LLM, text-to-image models), the former denoting any experience and the latter indicating the use in their writing practice.

	Expertise	Year	Domain	AI Exp*
P1	Hobbyist	10	novel	Y/Y
P2	Hobbyist	19	novel, TRPG	N/N
P3	Hobbyist	8	novel	Y/N
P4	Hobbyist	6	novel	Y/Y
P5	Hobbyist	7	novel	N/N
P6	Hobbyist	5	novel	Y/Y
P7	Expert	8	screenwriting, game, TRPG	Y/Y
P8	Hobbyist	25	novel, fan fiction	N/N
P9	Hobbyist	5	novel	Y/Y

Additionally, we aimed to discover how Patchview might be used in the worldbuilding process.

- RQ4: How do users leverage features of Patchview for worldbuilding?

To answer these questions, we conducted a study that mixes a within-subject comparative task and an observational task, along with both quantitative and qualitative analyses of collected data.

5.1 Participants

We recruited nine participants (four women, three men, one non-binary, and one who did not disclose gender, ages 24-51, M=33.4, SD=8.7) through Upwork[1], a gigwork platform. We focused on recruiting hobbyists with extensive years of experience (at least five) or professionals who make a living out of story writing and worldbuilding. Participants were proficient in English. Six participants had experience using AI for story writing, and among them, five actively used AI for their practice. We detail participants in Table 1.

5.2 Procedure

The study was conducted remotely via Google Meet[2]. After welcoming the participants, we asked if they were okay with recording the session. Then, we asked participants to watch two instruction videos, each on 1) the overview of Patchview and ways to generate or create notes on the list module and 2) reading view visualizations. After each video, participants were given an opportunity to experiment with the functions that had just been introduced.

After two instruction videos, we asked participants to conduct the first task, answering sensemaking questions (RQ1). Specifically, we provided two types of questions: 1) *landscape questions*, characterizing the distribution of world elements in relation to specific concepts (e.g., To which faction most characters are associated with?), and 2) *comparison questions*, comparing different characters according to their relevance to concepts (e.g., Which character is most associated with faction A?). These were multiple choice questions with one correct option. We measured whether the participants were correct and the time taken to answer. We expected that if the visualization could help users with sensemaking, they would answer more accurately in less time.

Participants conducted the task in a within-subject fashion, in two conditions: only with the list interface of Figure 5b (baseline) and together with the view visualization (treatment). We prepared two collections of elements, both focusing on character descriptions. One collection considered three different factions to characterize

[1] https://www.upwork.com/

[2] https://meet.google.com/

elements. Another considered two axes of good-evil and law-chaos as concepts, which are often used as character alignment structures for role-playing games such as *Dungeons & Dragons* [37]. We populated each collection with 10 characters generated with Patchview. To visualize characters, we leveraged Patchview's recognition results. The authors crafted questions after carefully reading through all generated elements (Appendix A). For each collection, we asked participants to answer both types of questions, with one collection given the baseline condition and the other with the treatment condition. We randomized the order of conditions to minimize ordering effects. For each question, we asked participants first to open the link to the question. After they understood the question, we asked them to open the link to the tool and answer the question with the story world provided in the tool. We timed the time taken to answer questions.

After the first task, participants watched three more videos on Patchview's functions: 1) creating and configuring views, 2) generating and editing world elements with views, and 3) rewriting elements and connecting multiple views. As before, participants were allowed to experiment with the just-introduced functions after finishing each video.

Once all functions were introduced, we asked participants to perform a second task: building a story world with Patchview while thinking aloud. We asked them to create at least one view and put five elements in the view. Moreover, we asked participants to place elements in the view where they think should be when finishing the task. Through this task, we wanted to understand if participants could use Patchview with concepts of their interest, visualizing elements (RQ1), steering generation with their nuanced intentions (RQ2), and correcting AI results and behaviors during the usage (RQ3). Moreover, we wanted to learn how Patchview supports the worldbuilding process (RQ4).

To understand participant behavior during this task, we collected logs of Patchview usage, including concepts that participants considered, steering inputs they made, outputs they received from Patchview, and corrections that they made to outputs. We also collected screen and think-aloud recordings. Participants could spend at most 40 minutes on this task. After the task, we asked participants to complete a survey and an interview. The survey asked about the helpfulness of each Patchview feature and included Creativity Support Index [12] questions on enjoyment, exploration, expressiveness, immersion, and the results of tool usage being worth the effort. Note that we did not use Creativity Support Index questions to compare the tool to others, only to gather participants' overall impressions of the tool. The interview aimed to elicit detailed perceptions about functions of Patchview and how Patchview could be used in participants' actual practices. The whole study took at most 120 minutes. Each participant received $60 for participation.

5.3 Results

We analyzed survey responses (Figure 13), recognition and steering errors from log data (Figure 14 and 16), answer time and correctness of the sensemaking questions (Figure 15), video recordings, and interview data. We measured recognition errors by the difference between Patchview's automatic placements of elements and the user's final placements of the same elements in views. This error will be zero if the user does not reposition the placement, and one if the user repositions an extreme value to another extreme one (e.g., fully good to fully evil). We calculated steering errors by the difference between where the user placed steering inputs and the user's final placement of world elements generated with these inputs. This error will be zero if the content of the generated element perfectly aligns with the user input, and one when the AI generates a totally misaligned element with the user input (e.g., a fully evil character generated with fully good input). This approach measures errors from the natural usage of the tool. However, note that this approach also has a limitation, as correcting the error does have the cost of moving the element. Moreover, it cannot consider errors of elements deleted during usage, as it requires the final placement of the element in the view by the user. Note that participants did not delete a high number of elements—participants deleted 10 elements out of 181 for recognition and two elements out of 33 for steering. Moreover, we could not collect errors for rewriting interactions due to a technical issue. We analyzed video recordings and interview data by iterative coding with inductive analysis.

5.3.1 *RQ1: Visualization helped users with sensemaking world elements.*

The participants seemed to largely agree with how Patchview placed world elements in the view. Figure 14 shows that the mean recognition error was measured to be 0.04 on a 0-to-1 scale for the user's arbitrary concepts. This result resonates with participants' interview responses ($N = 6$). For instance, P9 mentioned that Patchview accurately recognized concept relevance even in challenging cases: "*It actually grasped my intention even though I gave two words, basically.*"

With largely accurate automatic visualization, in the first survey question (Figure 13), participants responded that Patchview helped them understand the landscape of elements in the story world. The helpfulness of visualization also manifests in the sensemaking question results (Figure 15), specifically for landscape questions. When answering landscape questions, participants were significantly faster with visualization than without (Mann-Whitney $U = 79, n_1 = n_2 = 9, p < 0.001$[3]) and more correctly answered questions. However, for comparison questions, there was no significant difference in time taken to answer questions between conditions (Mann-Whitney $U = 60$, $n_1 = n_2 = 9$, $p > 0.05$). Moreover, participants were similarly accurate in answering comparison questions.

Interview results resonated with these findings: participants mentioned that they could easily understand the landscape of world elements with the help of visualization ($N = 9$), allowing them to track generated elements while keeping the world under the rule and the structure. P1 mentioned: "*The different views and stuff, actually seeing that on there and keeping track of it, I think, would be helpful. ... Because I end up building up too many and then I forget what the differences in each one's personality are sometimes.*" P9 also appreciated the customizability of the visualization.

Patchview's visualization also influenced how participants thought about each concept. That is, when participants do not agree with Patchview's placement of elements, some participants reflected on their own perception of concepts ($N = 5$), often concretizing how they think about concepts. For example, P4 mentioned: "*When I*

[3] We used non-parametric test due to small sample size and non-equal variances.

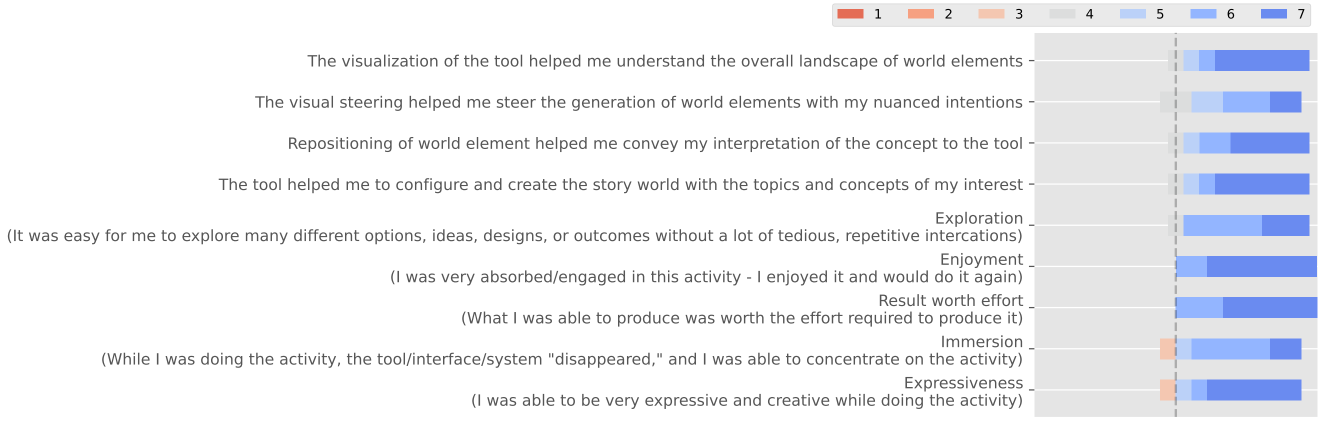


Figure 13: Survey results.

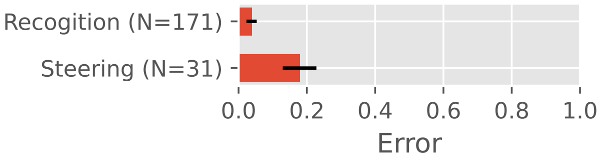


Figure 14: Errors in 1) recognizing the concept weights for elements placed in the visualization and 2) steering the generation of elements according to concept weights specified by the user, measured on a 0-to-1 scale. The error bars in this paper indicate the 95% confidence intervals.

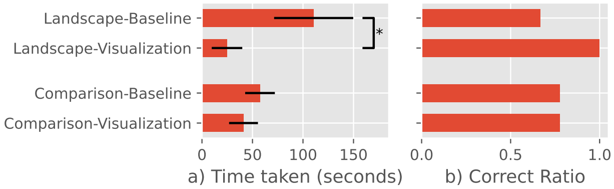


Figure 15: Sensemaking question results. a) Time taken for answering questions and b) correct ratio. * indicates $p < 0.001$.

put this 'strength', I was, kind of, just thinking about overall strength. And then, it interpreted it as physical strength. So I was like, 'this is now all across these (elements), and now that really gives a more nuanced idea of… power'" (Figure 17). However, there were also cases where the participant had a strong idea of how they think about concepts, and for those cases, they realized that they would need to sharpen their verbiage about the concept ($N = 3$). Some mentioned that future versions of the tool can explicitly support it. For example, P3 mentioned: "*I think if there's a pop-up that says, 'Can you give more information' or something like that, I think that would help me to force some clarity before it visualizes.*"

5.3.2 RQ2: Visual steering of Patchview allowed users to steer the generation with nuanced intentions. The results indicate that the steering function was fairly accurate when used for arbitrary concepts of the participants' interests. The mean steering error was measured to be 0.18 on a 0-to-1 scale (Figure 14). As the ordinal scale of five on a bi-directional dimension is often considered to be easily discernible by people [48], if we assume uniform intervals between levels (which is often used in ML [48, 62]), the error of 0.18 would be smaller than a single gap in a five-level ordinal scale (0.25). Hence, we conclude that Patchview allows users to steer the generation accurately in a granularity finer than easily discernible five-level scale on dimensions with two concepts. While this standard would need to be different for cases when a view has more than two concepts, in our study, only four steered generation results considered more than two concepts. As in the second question in Figure 13, participants also perceived that the tool helped them steer the world element generation with their nuanced intentions.

Participants mentioned that visual steering for element generation and rewriting was intuitive ($N = 7$). For example, for visual rewriting interaction, P6 mentioned: "*All I had to do is to move where I wanted the story element to be reconnected to and that's like a no-brainer that just takes a couple of mouse clicks and you're good to go.*" Participants also noted that visual steering helped them express nuanced intentions, even allowing them to realize the semantic space that they could not think about ($N = 6$). For example, P2 mentioned that they could use visual steering to create a set of characters that would make more conflicts than randomly generating them. On the other hand, participants thought that natural language prompts often require more cognitive effort as they need to bring up specific instructions ($N = 2$). However, participants thought that natural language prompts are beneficial as the user can be more specific in the instruction ($N = 4$). With different strengths, some participants ($N = 2$) thought that visual steering and natural language prompting complement each other, as P7 mentioned, by "*choosing the point via steering and then giving it a little bit of (natural language) input.*" For example, one limitation of the current visual rewriting interaction is that it often changes aspects the user likes. Adding a natural

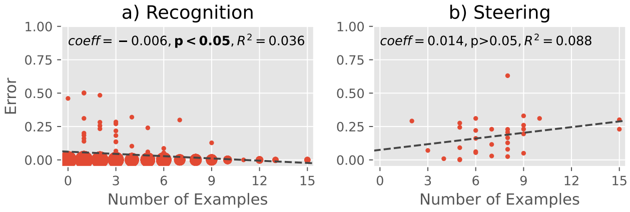


Figure 16: a) Recognition and b) steering errors according to the number of examples added by the user. Dot sizes indicate the number of data points with the same errors and example numbers. Each plot contains the linear regression result.

language prompt, such as specifying which aspects not to change, could have alleviated the issue.

5.3.3 RQ3: With more user examples, Patchview could only improve recognition, not the steering, but to a small extent. In the third question of Figure 13, most participants answered that they could convey their interpretations of the concepts through repositioning elements. Similarly, during the interview, participants mentioned that it was easy to revise AI results by simply moving elements on the view or by rewriting interactions ($N = 6$). For example, P7 mentioned: "*I really like that you have the ability to say, kind of like, 'No I'm telling you where this should go' versus 'I want you to actually adjust it to fit there.'*"

However, the user's correction of concepts through the addition of more examples did not turn into dramatic changes in AI behaviors. As in Figure 16a, when we conducted a linear regression on the relation between the number of examples and the recognition error, the addition of more examples significantly decreased errors ($p < 0.05$), but with a small magnitude ($coeff = -0.006$) and a small ability to explain variations ($R^2 = 0.036$). The analysis on steering errors (Figure 16b) revealed no significance in the correlation between the number of added examples and errors ($p > 0.05$). These resonated with the interview responses: participants felt that the study session was not long enough to sense that the tool is learning from what they are doing in the tool ($N = 3$). P7 mentioned that rather than having such tool behavior changes implicit, making them more explicit to the user would be helpful: "*I would have had to play with it a lot more to know if it actually was learning ... It'd be interesting if I could have a feature to refresh ... So a refresh thing would help me see what it was learning from me.*"

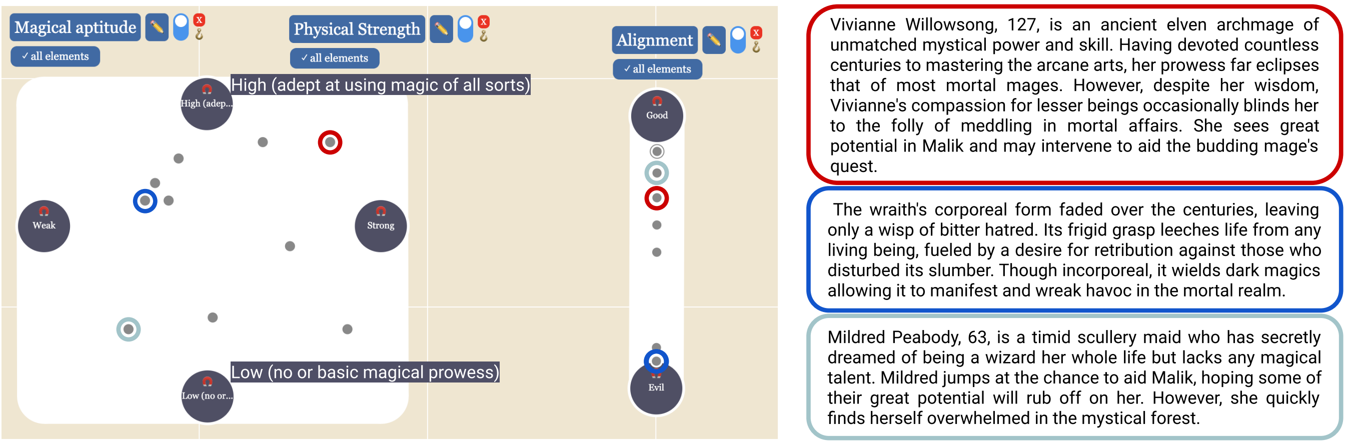
5.3.4 RQ4: Participants could flexibly create their own story world and suggested ways to improve the tool for more comprehensive story writing. With Patchview, participants could structure the story world according to concepts of their interest. Table 2 shows the summary of views participants created. Many participants created views for alignments [24], either good-evil ($N = 5$) or law-chaos ($N = 4$) dimensions. It might be because these alignments are widely used to organize characters or because we used these dimensions as examples in the tutorial. Participants also created views with custom concepts, such as factions ($N = 3$), locations ($N = 1$), or other concepts of the participant's interest (e.g., magical aptitude from Figure 17, $N = 9$). One interesting view type was story timeline, where participants tried to align events between the story's beginning and end (Figure 18, $N = 2$). It shows that participants would eventually want to create a coherent storyline with the world elements they created. Participants added various element types to the views, but the character was most frequently added.

Table 2: Views created by participants. × indicates that multiple views are either tied or crossed with each other. For cases with more than two concepts in view, we noted commonalities between concepts instead of directly showing the concepts themselves. El and char stand for element and character, respectively.

	View concepts	El #	El Type
P1	good - evil × law - chaos	9	char
P2	four factions	9	char
P3	three life focuses	6	char
P4	story timeline (beginning - end)	4	event
	magical aptitude (high - low) × physical strength (strong - weak)	10	char, faction
	good - evil	8	char, faction
P5	good - evil × law - chaos	3	faction, place
	three locations	1	character
P6	cats - pugs × library - tombs	13	char, faction, prop, event
P7	good - evil × law - chaos	5	char
	story timeline (beginning - end)	3	event
	two factions	5	char
P8	good - evil	8	char
	law - chaos	8	char
	honest - deceitful	5	char
P9	logical - emotional × science-oriented - belief-oriented	7	char, event
	positive event - negative event	3	event
	three factions	5	char

As shown in the results of the fifth to ninth questions of Figure 13, participants felt that Patchview helped them expressively explore various ideas while enjoying and being immersed in the process, ending up with a result that was worth their efforts. Participants thought that the tool could help with ideation or filling in details for the part they are not good at ($N = 8$). As generative features could add new things to the user's world, some participants mentioned that AI generation would be most useful for the ideation stage or the settings where the user needs to constantly come up with new elements (e.g., D&D), rather than for the cases where the user already has a highly-structured and consistent world ($N = 2$). P9 also mentioned that the visualization could facilitate collaboration when multiple people are working on the worldbuilding: "*If you're writing, let's say with a group of people, ... it's helpful to be able to quickly see 'Okay this character is in this faction', instead of having to go through it because you might not be the one who wrote it.*"

While participants appreciated Patchview, they also made suggestions to improve the tool for worldbuilding and story writing practices. First, as story elements can have relationships with

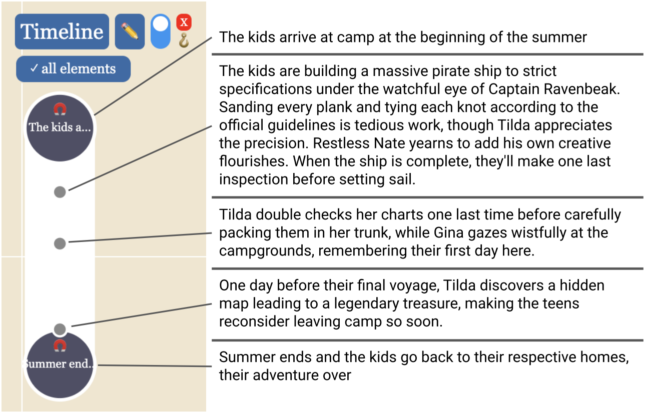


Vivianne Willowsong, 127, is an ancient elven archmage of unmatched mystical power and skill. Having devoted countless centuries to mastering the arcane arts, her prowess far eclipses that of most mortal mages. However, despite her wisdom, Vivianne's compassion for lesser beings occasionally blinds her to the folly of meddling in mortal affairs. She sees great potential in Malik and may intervene to aid the budding mage's quest.

The wraith's corporeal form faded over the centuries, leaving only a wisp of bitter hatred. Its frigid grasp leeches life from any living being, fueled by a desire for retribution against those who disturbed its slumber. Though incorporeal, it wields dark magics allowing it to manifest and wreak havoc in the mortal realm.

Mildred Peabody, 63, is a timid scullery maid who has secretly dreamed of being a wizard her whole life but lacks any magical talent. Mildred jumps at the chance to aid Malik, hoping some of their great potential will rub off on her. However, she quickly finds herself overwhelmed in the mystical forest.

Figure 17: Views created by P4, organized by magical aptitude, physical strength (which is noted as Weak-Strong), and good-evil alignment. Views include characters and factions as elements. Example elements are presented on the right and where they are positioned in the view is marked with the circles of the same border color.



The kids arrive at camp at the beginning of the summer

The kids are building a massive pirate ship to strict specifications under the watchful eye of Captain Ravenbeak. Sanding every plank and tying each knot according to the official guidelines is tedious work, though Tilda appreciates the precision. Restless Nate yearns to add his own creative flourishes. When the ship is complete, they'll make one last inspection before setting sail.

Tilda double checks her charts one last time before carefully packing them in her trunk, while Gina gazes wistfully at the campgrounds, remembering their first day here.

One day before their final voyage, Tilda discovers a hidden map leading to a legendary treasure, making the teens reconsider leaving camp so soon.

Summer ends and the kids go back to their respective homes, their adventure over

Figure 18: A timeline view created by P7.

each other (e.g., relationships between characters), participants suggested features to visualize such relationships, such as arrows between elements ($N = 4$). Specifically with the temporal relationships, as shown in the participants' usage patterns, participants mentioned that a view more specialized for timeline would help ($N = 2$). As some elements could change with the story's progress, P9 mentioned that it would be helpful to have the feature to "draw out" the trajectory of changes so that it can direct the tool to generate those changes. Second, as world elements could constantly change with the use of the tool, participants wanted the tool to handle the consistency between elements ($N = 2$). For example, if the user edits one character's name in one note, participants wanted the tool to automatically update the character's name appearing in other notes. Third, some participants wanted to flesh out world elements by adding relevant images with AI image generation models ($N = 2$). They mentioned that additional visuals could help them not only concretize the world element but also quickly grasp it. Lastly, some wanted a feature to import their own world to the tool if they already have the world that they are working on ($N = 2$).

6 DISCUSSION

We discuss Patchview's interaction design, utility in worldbuilding and story writing practices, technical alternatives, and limitations.

6.1 Visually Bridging User and Generative AI

We introduce GD&M as a visual interaction to bridge the user and generative AIs. As mentioned in Section 3, GD&M is most applicable when the user generates a lot of things within the conceptual dimensions of their interests. The interaction helps with the user's evaluation, specification, and alignment of AI behaviors [42, 52]. We believe this interaction can be adopted to other use cases. In the text domain, for instance, GD&M could be used for organizing and steering idea generation [47] with LLMs. Beyond text, it could also be extended to the curation of image or video generation [2], showing thumbnails instead of plain glyphs in the visualization.

As mentioned in Section 2.3, GD&M extends previous work [50] by allowing flexible configuration of arbitrary concepts on continuous dimensions. It has a benefit over the previous approach for cases when a single element has a mix of multiple attributes (e.g., a single character is associated with three factions). One finding regarding this continuous dimension was that people occasionally disagree with where the tool places elements in the view. This might be because finer granularity in continuous spaces allows users to easily see such disagreements. Our findings indicate that these disagreements facilitate reflection [33] and critical thinking about the user's concepts.

GD&M also extends previous work by allowing users to correct AI behaviors directly in the visual spaces. While the interaction itself holds promise, aligning AI behaviors to user-corrected examples was challenging in our version of the tool. This might be because there was little room for improvement as the error was already low without any user-added examples. Future work can explore technical improvements, such as selecting examples that can maximize the performance of recognition and steering [36, 45].

6.2 AI-Supported Worldbuilding and Storytelling

We found that LLMs could support worldbuilding by providing ideas and filling in parts of the world on behalf of the users. However, AI may have both positive and negative effects on creative tasks like worldbuilding. For example, previous work showed that LLM usage could drive users to produce more homogenous responses to a divergent ideation task [3]. This could be a problem in a worldbuilding context if users hope to create truly unique worlds. As future work, it would be worthwhile to investigate what other problems LLMs might introduce in the context of story writing and what measures might be taken to tackle those problems.

Study participants expressed a clear desire to use worlds created with Patchview as the basis for longer-form stories. In particular, participants repeatedly expressed a desire for a specialized timeline view that would enable them to organize world events into a coherent chronology, and in two cases even improvised a timeline view using the existing Patchview feature set (Figure 18). AI-supported storytelling might involve many different levels of user control, ranging from humans writing every aspect of the story to the full simulation of the story world by AI [44]; previous work hinted at user specification of a high-level story arc [17] and participation as a character in the story [44], but many novel interaction paradigms remain for future work to explore. Technically, LLM-based story world simulation would likely face consistency issues [32] due to the tendency of LLMs to "hallucinate" [26], which would need to be addressed to support coherent storytelling.

6.3 Technical Alternatives to Prompt Engineering and Closed Models

The prompt engineering techniques we used in Patchview (including chain-of-thought) yielded good performance on both recognition and steering tasks without any additional training or control techniques. This suggests that current general-purpose LLMs are capable of reasoning effectively about concept relatedness, even along continuous dimensions defined between arbitrary concepts. However, the current Patchview prototype is limited in its interactivity by relatively high latency. For instance, with two concepts in a view, both steering the generation of a new world element and recognizing its position took longer than 15 seconds. This latency is due partly to the large size of the underlying LLM and partly to our prompting techniques: increasing numbers of world elements, concepts, and user examples cause our prompts to become progressively longer, and chain-of-thought prompting increases the number of tokens generated in response to each prompt, further driving up latency. Additionally, current general-purpose LLMs are not specifically tuned for creative applications, resulting in clear weaknesses for creative work [10]. These options are often trained on loosely defined preferences [43], rather than using rewards more targeted for creative applications. As they are often closed models, improving on those models would be challenging.

We suggest that future work can explore other options than prompt engineering and closed models. To control generation with the concepts on the continuous dimensions, we can consider the manipulation of task representations in the hidden layers of the LLM [54, 64]. One benefit of this approach is that, once we have the vector about the concept, it does not require tokens for conveying concepts and examples in the prompts. Moreover, if this approach is robust enough, chain-of-thought also might not be necessary. However, future work would need to validate if this approach can effectively steer generation with higher efficiency than prompt engineering. Moreover, how to consider the user-corrected examples with this approach is also moot. Alternative to closed models would be smaller-sized open models that are fine-tuned to creative use cases, which is becoming more feasible with many open-source LLM options [27, 53]. Building upon these open-source LLMs, researchers recently introduced models specialized for creative writing [56]. Building upon these efforts, avenues we can explore include having a higher quality creative writing dataset with annotations on various aspects of the text quality (e.g., engagingness, novelty, diversity) and tuning models while considering those aspects as losses or rewards [59].

6.4 Limitations

Our tool's usability and functionalities could be improved in the future. For instance, adding filtering functions to the list module would likely help the user find relevant elements when there are many elements to sort through. Future versions could also better handle under-specified or irrelevant concepts. The current Patchview would try to get how the user interprets those concepts from examples provided during the usage. However, such an approach would still have limitations as Patchview relies on prompt engineering, and alleviating those issues can be future work.

One limitation of our study is that we could not collect steering results data for rewriting interactions due to technical issues. Moreover, our technical analysis is not the most rigorous but prioritized analyzing data naturally collected during the study. For instance, the user study data might have only covered a subset of genres, settings, and concepts that would frequently used for worldbuilding. The error rates could also have been confounded by the fact that marking errors could incur a small additional cost to users as they need to move elements manually. Due to these reasons, future work might involve a more rigorous technical evaluation. This future work could also evaluate other technical implementation options mentioned in Section 6.3.

While Patchview might be most effective in long-term projects (as it is designed to help users create and organize an expansive fictional world consisting of many distinct elements), our study involved only a single session. Future work might investigate the use of Patchview for long-term projects, including the extension of already existing story worlds. Furthermore, we did not compare the design of Patchview to other alternatives when the user creates their own story world with LLMs; we instead focused more on identifying usage patterns and whether the tool is technically able to achieve its design goals. Future work may investigate comparison to other tools.

7 CONCLUSION

We introduce Patchview, an LLM-powered worldbuilding tool that adopts generative dust and magnet (GD&M) interactions to support interaction with generative AI. With GD&M, Patchview facilitates sensemaking of generated story elements by placing elements close to concepts of high relevance, similar to how magnets attract iron dust particles. It also supports generation steering and AI behavior correction by leveraging the visual space configured by concepts. A user study showed that Patchview could facilitate understanding the landscape of story world elements and steering of element generation with nuanced intentions that are difficult to express in natural language alone. The interaction of correcting misaligned AI results was intuitive, but those corrections minimally improved the alignment of AI behaviors to the user's perception, indicating one possible direction for future work. We hope Patchview and GD&M provide insights on visual interactions for evaluation, specification, and alignment of generative AI behaviors to the user's intention.

ACKNOWLEDGMENTS

We thank the many people at Midjourney who provided infrastructural and logistical support for this work. We also thank Yoonjoo Lee, Jordan Huffaker, and Kihoon Son for giving feedback to the early prototype of Patchview, and user study participants for their valuable insights on the tool.

REFERENCES

[1] [n. d.]. Sudowrite. https://www.sudowrite.com/

[2] Shm Garanganao Almeda, J. D. Zamfirescu-Pereira, Kyu Won Kim, Pradeep Mani Rathnam, and Bjoern Hartmann. 2024. Prompting for Discovery: Flexible Sense-Making for AI Art-Making with Dreamsheets. arXiv:2310.09985 [cs.HC]

[3] Barrett R. Anderson, Jash Hemant Shah, and Max Kreminski. 2024. Homogenization Effects of Large Language Models on Human Creative Ideation. arXiv:2402.01536 [cs.HC]

[4] Tyler Angert, Miroslav Suzara, Jenny Han, Christopher Pondoc, and Hariharan Subramonyam. 2023. Spellburst: A Node-Based Interface for Exploratory Creative Coding with Natural Language Prompts. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 100, 22 pages. https://doi.org/10.1145/3586183.3606719

[5] Anthropic. 2023. Model Card and Evaluations for Claude Models. https://www-files.anthropic.com/production/images/Model-Card-Claude-2.pdf

[6] Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena Glassman. 2023. ChainForge: A Visual Toolkit for Prompt Engineering and LLM Hypothesis Testing. arXiv:2309.09128 [cs.HC]

[7] Oloff C. Biermann, Ning F. Ma, and Dongwook Yoon. 2022. From Tool to Companion: Storywriters Want AI Writers to Respect Their Personal Values and Writing Strategies. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference* (Virtual Event, Australia) *(DIS '22)*. Association for Computing Machinery, New York, NY, USA, 1209–1227. https://doi.org/10.1145/3532106.3533506

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[9] Alex Calderwood, Vivian Qiu, K. Gero, and Lydia B. Chilton. 2020. How Novelists Use Generative Language Models: An Exploratory User Study. In *HAI-GEN+user2agent@IUI*. https://api.semanticscholar.org/CorpusID:233479959

[10] Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. 2024. Art or Artifice? Large Language Models and the False Promise of Creativity. arXiv:2309.14556 [cs.CL]

[11] Nan-Chen Chen, Jina Suh, Johan Verwey, Gonzalo Ramos, Steven Drucker, and Patrice Simard. 2018. AnchorViz: Facilitating Classifier Error Discovery through Interactive Semantic Data Exploration. In *23rd International Conference on Intelligent User Interfaces* (Tokyo, Japan) *(IUI '18)*. Association for Computing Machinery, New York, NY, USA, 269–280. https://doi.org/10.1145/3172944.3172950

[12] Erin Cherry and Celine Latulipe. 2014. Quantifying the Creativity Support of Digital Tools through the Creativity Support Index. *ACM Trans. Comput.-Hum. Interact.* 21, 4, Article 21 (jun 2014), 25 pages. https://doi.org/10.1145/2617588

[13] Jean-Peïc Chou, Alexa Fay Siu, Nedim Lipka, Ryan Rossi, Franck Dernoncourt, and Maneesh Agrawala. 2023. TaleStream: Supporting Story Ideation with Trope Knowledge. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 52, 12 pages. https://doi.org/10.1145/3586183.3606807

[14] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. PaLM: Scaling Language Modeling with Pathways. arXiv:2204.02311 [cs.CL]

[15] John Joon Young Chung and Eytan Adar. 2023. PromptPaint: Steering Text-to-Image Generation Through Paint Medium-like Interactions. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 6, 17 pages. https://doi.org/10.1145/3586183.3606777

[16] John Joon Young Chung, Shiqing He, and Eytan Adar. 2022. Artist Support Networks: Implications for Future Creativity Support Tools. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference* (Virtual Event, Australia) *(DIS '22)*. Association for Computing Machinery, New York, NY, USA, 232–246. https://doi.org/10.1145/3532106.3533505

[17] John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. 2022. TaleBrush: Sketching Stories with Generative Pretrained Language Models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 209, 19 pages. https://doi.org/10.1145/3491102.3501819

[18] John Joon Young Chung, Hijung Valentina Shin, Haijun Xia, Li-yi Wei, and Rubaiat Habib Kazi. 2021. Beyond Show of Hands: Engaging Viewers via Expressive and Scalable Visual Communication in Live Streaming. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 109, 14 pages. https://doi.org/10.1145/3411764.3445419

[19] Elizabeth Clark, Anne Spencer Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith. 2018. Creative Writing with a Machine in the Loop: Case Studies on Slogans and Stories. In *23rd International Conference on Intelligent User Interfaces* (Tokyo, Japan) *(IUI '18)*. Association for Computing Machinery, New York, NY, USA, 329–340. https://doi.org/10.1145/3172944.3172983

[20] Hai Dang, Frederik Brudy, George Fitzmaurice, and Fraser Anderson. 2023. WorldSmith: Iterative and Expressive Prompting for World Building with a Generative AI. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 63, 17 pages. https://doi.org/10.1145/3586183.3606772

[21] Karin Fast and Henrik Örnebring. 2017. Transmedia world-building: The Shadow (1931–present) and Transformers (1984–present). *International Journal of Cultural Studies* 20, 6 (2017), 636–652. https://doi.org/10.1177/1367877915605887 arXiv:https://doi.org/10.1177/1367877915605887

[22] Katy Ilonka Gero, Tao Long, and Lydia B Chilton. 2023. Social Dynamics of AI Support in Creative Writing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 245, 15 pages. https://doi.org/10.1145/3544548.3580782

[23] Yuan Gong, Youxin Pang, Xiaodong Cun, Menghan Xia, Yingqing He, Haoxin Chen, Longyue Wang, Yong Zhang, Xintao Wang, Ying Shan, and Yujiu Yang. 2023. Interactive Story Visualization with Multiple Characters. In *SIGGRAPH Asia 2023 Conference Papers* (Sydney, NSW, Australia) *(SA '23)*. Association for Computing Machinery, New York, NY, USA, Article 101, 10 pages. https://doi.org/10.1145/3610548.3618184

[24] T. Hergenrader. 2018. *Collaborative Worldbuilding for Writers and Gamers.* Bloomsbury Academic. https://books.google.co.kr/books?id=z-_7swEACAAJ

[25] Md Naimul Hoque, Bhavya Ghai, Kari Kraus, and Niklas Elmqvist. 2023. Portrayal: Leveraging NLP and Visualization for Analyzing Fictional Characters. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference* (Pittsburgh, PA, USA) *(DIS '23)*. Association for Computing Machinery, New York, NY, USA, 74–94. https://doi.org/10.1145/3563657.3596000

[26] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.* 55, 12, Article 248 (mar 2023), 38 pages. https://doi.org/10.1145/3571730

[27] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of Experts. arXiv:2401.04088 [cs.LG]

[28] Peiling Jiang, Jude Rayan, Steven P. Dow, and Haijun Xia. 2023. Graphologue: Exploring Large Language Model Responses with Interactive Diagrams. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 3, 20 pages. https://doi.org/10.1145/3586183.3606737

[29] Taewook Kim, Hyomin Han, Eytan Adar, Matthew Kay, and John Joon Young Chung. 2024. Authors' Values and Attitudes Towards AI-bridged Scalable Personalization of Creative Language Arts. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA.

[30] Tae Soo Kim, Yoonjoo Lee, Minsuk Chang, and Juho Kim. 2023. Cells, Generators, and Lenses: Design Framework for Object-Oriented Interaction with Large Language Models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 4, 18 pages. https://doi.org/10.1145/3586183.3606833

[31] Max Kreminski, Melanie Dickinson, Noah Wardrip-Fruin, and Michael Mateas. 2022. Loose Ends: A Mixed-Initiative Creative Interface for Playful Storytelling. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 18, 1 (Oct. 2022), 120–128. https://doi.org/10.1609/aiide.v18i1.21955

[32] Max Kreminski and Chris Martens. 2022. Unmet creativity support needs in computationally supported creative writing. In *Proceedings of the First Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2022)*. 74–82.

[33] Max Kreminski and Michael Mateas. 2021. Reflective Creators. In *International Conference on Computational Creativity*. 309–318.

[34] Mina Lee, Katy Ilonka Gero, John Joon Young Chung, Simon Buckingham Shum, Vipul Raheja, Hua Shen, Subhashini Venugopalan, Thiemo Wambsganss, David Zhou, Emad A. Alghamdi, Tal August, Avinash Bhat, Madiha Zahrah Choksi, Senjuti Dutta, Jin L.C. Guo, Md Naimul Hoque, Yewon Kim, Simon Knight, Seyed Parsa Neshaei, Antonette Shibani, Disha Shrivastava, Lila Shroff, Agnia Sergeyuk, Jessi Stark, Sarah Sterman, Sitong Wang, Antoine Bosselut, Daniel Buschek, Joseph Chee Chang, Sherol Chen, Max Kreminski, Joonsuk Park, Roy Pea, Eugenia Ha Rim Rho, Zejiang Shen, and Pao Siangliulue. 2024. A Design Space for Intelligent and Interactive Writing Assistants. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA.

[35] Mina Lee, Percy Liang, and Qian Yang. 2022. CoAuthor: Designing a Human-AI Collaborative Writing Dataset for Exploring Language Model Capabilities. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 388, 19 pages. https://doi.org/10.1145/3491102.3502030

[36] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What Makes Good In-Context Examples for GPT-3?. In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, Eneko Agirre, Marianna Apidianaki, and Ivan Vulić (Eds.). Association for Computational Linguistics, Dublin, Ireland and Online, 100–114. https://doi.org/10.18653/v1/2022.deelio-1.10

[37] I. Livingstone. 1986. *Dicing with Dragons: An Introduction to Role-Playing Games*. Penguin Group USA, Incorporated. https://books.google.co.kr/books?id=uBqXPwAACAAJ

[38] Ryan Louie, Andy Coenen, Cheng Zhi Huang, Michael Terry, and Carrie J. Cai. 2020. Novice-AI Music Co-Creation via AI-Steering Tools for Deep Generative Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376739

[39] Krzysztof M Maj. 2015. Transmedial world-building in fictional narratives. *IMAGE. Zeitschrift für interdisziplinäre Bildwissenschaft* 11, 2 (2015), 83–96.

[40] Piotr Mirowski, Kory W. Mathewson, Jaylen Pittman, and Richard Evans. 2023. Co-Writing Screenplays and Theatre Scripts with Language Models: Evaluation by Industry Professionals. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 355, 34 pages. https://doi.org/10.1145/3544548.3581225

[41] Tamara Munzner. 2014. *Visualization analysis and design.* CRC press.

[42] Donald A. Norman. 2002. *The design of everyday things.* Basic Books, [New York]. http://www.amazon.de/The-Design-Everyday-Things-Norman/dp/0465067107/ref=wl_it_dp_o_pC_S_nC?ie=UTF8&colid=151193SNGKJT9&coliid=I262V9ZRW8HR2C

[43] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

[44] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 2, 22 pages. https://doi.org/10.1145/3586183.3606763

[45] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning To Retrieve Prompts for In-Context Learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (Eds.). Association for Computational Linguistics, Seattle, United States, 2655–2671. https://doi.org/10.18653/v1/2022.naacl-main.191

[46] Natalia Samutina. 2016. Fan fiction as world-building: Transformative reception in crossover writing. *Continuum* 30, 4 (2016), 433–450.

[47] Pao Siangliulue, Joel Chan, Steven P. Dow, and Krzysztof Z. Gajos. 2016. IdeaHound: Improving Large-scale Collaborative Ideation with Crowd-Powered Real-time Semantic Modeling. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 609–624. https://doi.org/10.1145/2984511.2984578

[48] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard (Eds.). Association for Computational Linguistics, Seattle, Washington, USA, 1631–1642. https://aclanthology.org/D13-1170

[49] Hari Subramonyam, Roy Pea, Christopher Lawrence Pondoc, Maneesh Agrawala, and Colleen Seifert. 2024. Bridging the Gulf of Envisioning: Cognitive Challenges in Prompt Based Interactions with LLMs. (2024).

[50] Sangho Suh, Meng Chen, Bryan Min, Toby Jia-Jun Li, and Haijun Xia. 2023. Structured Generation and Exploration of Design Space with Large Language Models for Human-AI Co-Creation. arXiv:2310.12953 [cs.HC]

[51] Sangho Suh, Bryan Min, Srishti Palani, and Haijun Xia. 2023. Sensecape: Enabling Multilevel Exploration and Sensemaking with Large Language Models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 1, 18 pages. https://doi.org/10.1145/3586183.3606756

[52] Michael Terry, Chinmay Kulkarni, Martin Wattenberg, Lucas Dixon, and Meredith Ringel Morris. 2023. AI Alignment in the Design of Interactive AI: Specification Alignment, Process Alignment, and Evaluation Support. arXiv:2311.00710 [cs.HC]

[53] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288 [cs.CL]

[54] Theia Vogel. 2024. repeng. https://github.com/vgel/repeng/

[55] Nick Walton. 2019. AI Dungeon 2. https://aidungeon.cc/

[56] Tiannan Wang, Jiamin Chen, Qingrui Jia, Shuai Wang, Ruoyu Fang, Huilin Wang, Zhaowei Gao, Chunzhao Xie, Chuou Xu, Jihong Dai, Yibin Liu, Jialong Wu, Shengwei Ding, Long Li, Zhiwei Huang, Xinle Deng, Teng Yu, Gangan Ma, Han Xiao, Zixin Chen, Danjun Xiang, Yunxia Wang, Yuanyuan Zhu, Yi Xiao, Jing Wang, Yiru Wang, Siran Ding, Jiayang Huang, Jiayi Xu, Yilihamu Tayier, Zhenyu Hu, Yuan Gao, Chengfeng Zheng, Yueshu Ye, Yihang Li, Lei Wan, Xinyue Jiang, Yujie Wang, Siyu Cheng, Zhule Song, Xiangru Tang, Xiaohua Xu, Ningyu Zhang, Huajun Chen, Yuchen Eleanor Jiang, and Wangchunshu Zhou. 2024. Weaver: Foundation Models for Creative Writing. arXiv:2401.17268 [cs.CL]

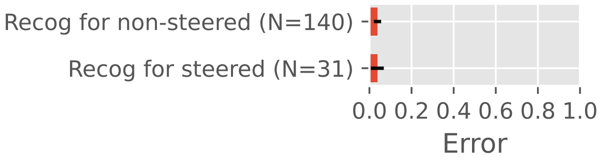


Figure 19: Errors in recognizing concept weights for elements placed in the visualization, when elements are generated (1) without and (2) with steering inputs.

[57] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 24824–24837. https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf

[58] Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 385, 22 pages. https://doi.org/10.1145/3491102.3517582

[59] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-Grained Human Feedback Gives Better Rewards for Language Model Training. *arXiv preprint arXiv:2306.01693* (2023).

[60] Ji Soo Yi, Rachel Melton, John Stasko, and Julie A. Jacko. 2005. Dust & Magnet: Multivariate Information Visualization Using a Magnet Metaphor. *Information Visualization* 4, 4 (oct 2005), 239–256. https://doi.org/10.1057/palgrave.ivs.9500099

[61] Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story Writing With Large Language Models. In *27th International Conference on Intelligent User Interfaces* (Helsinki, Finland) *(IUI '22)*. Association for Computing Machinery, New York, NY, USA, 841–852. https://doi.org/10.1145/3490099.3511105

[62] Biqiao Zhang, Georg Essl, and Emily Mower Provost. 2017. Predicting the distribution of emotion perception: capturing inter-rater variability. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (Glasgow, UK) *(ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 51–59. https://doi.org/10.1145/3136755.3136792

[63] Andrew Zhu, Lara Martin, Andrew Head, and Chris Callison-Burch. 2023. CALYPSO: LLMs as Dungeon Masters' Assistants. In *Proceedings of the Nineteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Salt Lake City) *(AIIDE '23)*. AAAI Press, Article 39, 11 pages. https://doi.org/10.1609/aiide.v19i1.27534

[64] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. 2023. Representation Engineering: A Top-Down Approach to AI Transparency. arXiv:2310.01405 [cs.LG]

A SENSEMAKING QUESTIONS

We list sensemaking questions used in the user study in Table 3.

B COMPARISON BETWEEN ELEMENTS GENERATED WITH OR WITHOUT VISUAL STEERING

We compare how Patchview generates elements differently with and without visual steering inputs. In Figure 19, we show how the recognition errors vary, and observe that they are similarly low regardless of whether visual steering input was used. With Welch's t-test, we found no significant difference between these two groups ($t(51.86) = 0.058, p > 0.5$).

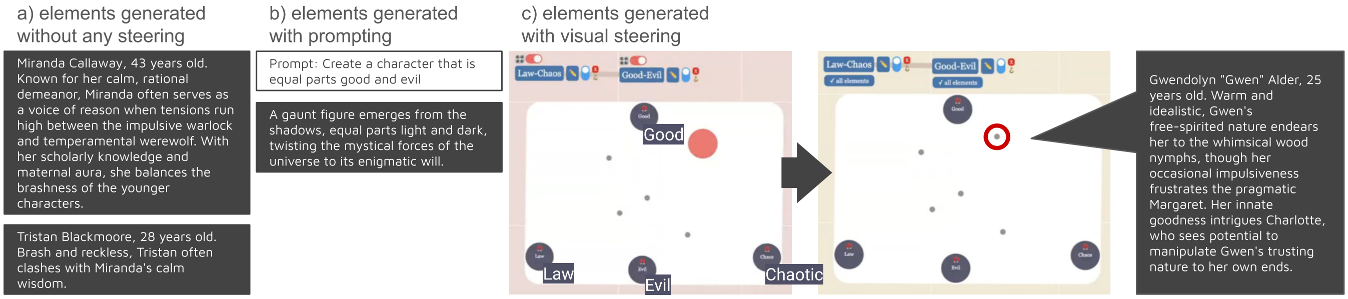
Figure 20 shows world elements generated by P1 without any steering (i.e., using only the element generation button); with a natural language prompt; and with visual steering. As shown in the case of "Tristan Blackmoore," not using any steering could result in an under-specified element description. Natural language prompts could help with steering generation, but as shown in Figure 20b, expressing nuanced intentions could be tough and not all prompts resulted in detailed and expressive descriptions of elements. Visual steering (Figure 20c) could be a complement to this, allowing users to express nuanced intentions with simple placement of a visual marker.

C EXAMPLE WORLD

We share an additional partial example of a user-created world from the user study in Figure 21.

Table 3: Questions used in the sensemaking tasks of the user study. In the type, L stands for landscape understanding questions and C stands for the comparison questions.

World	Type	Question	Options	Answer
1	L	Which faction is linked to the smallest number of characters in this world?	The Faerie Fleet (a mysterious group of tiny winged humanoids that pilot delicate yet powerful ships grown from seeds)	
			The Iron Brigade (a regiment of steampunk automatons that pilot bulky ironclad warships)	O
			The Skysharks (a clan of winged reptilian mercenaries that fly agile bioships grown from eggs)	
1	C	Choose the character that is least associated with Skysharks.	Cogwhistle is a 112-year old brass automaton who serves as an elite commander in the Iron Brigade. With a clockwork mind and pneumatic limbs, Cogwhistle is utterly devoted to his steam-driven brethren yet feels a flickering fascination with the graceful faeries that contrasts his mechanical nature.	
			Frostwind is a 31-year old winged velociraptor mercenary who serves as Razortooth's trusted lieutenant in the Skysharks. Hatched from a faerie-spliced egg, he has some fae ancestry that gives him an icy demeanor and talent for aerial combat. Frostwind is coldly loyal to Razortooth yet feels a faint kinship with Silverblossom.	
			Razortooth is a 37-year old winged velociraptor mercenary who leads the Skysharks clan. He is larger and more cunning than the rest of his kind, and is utterly ruthless in battle. His personal bioship Razors Edge is the fastest and most maneuverable ship in the clan.	
			Silvercog is a 17-year-old faerie automaton who escaped the Iron Brigade to join the Skysharks. Forged from faerie dust and brass, she has a precise clockwork mind yet yearns for the grace and freedom of her fae ancestors. Though mistrusted by Razortooth, Silvercog bonds with Silverslice over their shared outcast status and conflicted origins.	O
2	L	Which dimension is associated with the greatest number of characters?	Good-Law	O
			Good-Chaotic	
			Evil-Law	
			Evil-Chaotic	
2	C	Which character is most chaotic?	Sir Galahad Pureheart, age 45, is a devoted paladin who lives by a strict code of honor, righteousness and duty. Unwavering in his beliefs, he shows no mercy to those he views as evil or chaotic, though his actions are driven by a desire to protect the innocent and punish wrongdoers. His rigid worldview often puts him at odds with more free-spirited allies.	
			Captain Jade Stormcloud, age 32, is a brash but big-hearted pirate who lives life to the fullest. Though she chafes at rules and restrictions, her strong moral compass keeps her from taking her freedom too far. She would find common ground with Sir Galahad in fighting evil, but her flexible worldview would help temper his rigidity.	O
			Lord Vladimir Skullreaper, age 67, is a cruel tyrant who rules his lands with an iron fist. Public executions are commonplace under his absolute authority, as he shows no mercy to those who dare question his laws.	
			Brother Lucian Greymane, age 37, is a battle-hardened templar who tirelessly wages war against the forces of darkness. Though devoted to his holy crusade, hints of disillusionment sometimes pierce his staunch faith and code of honor. His zeal for righteousness is tempered with shades of world-weariness and moral ambiguity. While righteous at heart, he is no stranger to employing harsh methods when he deems the ends justify them. He would find kinship with Galahad but also empathize with Stormcloud's flexibility in fighting evil.	



a) elements generated without any steering

Miranda Callaway, 43 years old. Known for her calm, rational demeanor, Miranda often serves as a voice of reason when tensions run high between the impulsive warlock and temperamental werewolf. With her scholarly knowledge and maternal aura, she balances the brashness of the younger characters.

Tristan Blackmoore, 28 years old. Brash and reckless, Tristan often clashes with Miranda's calm wisdom.

b) elements generated with prompting

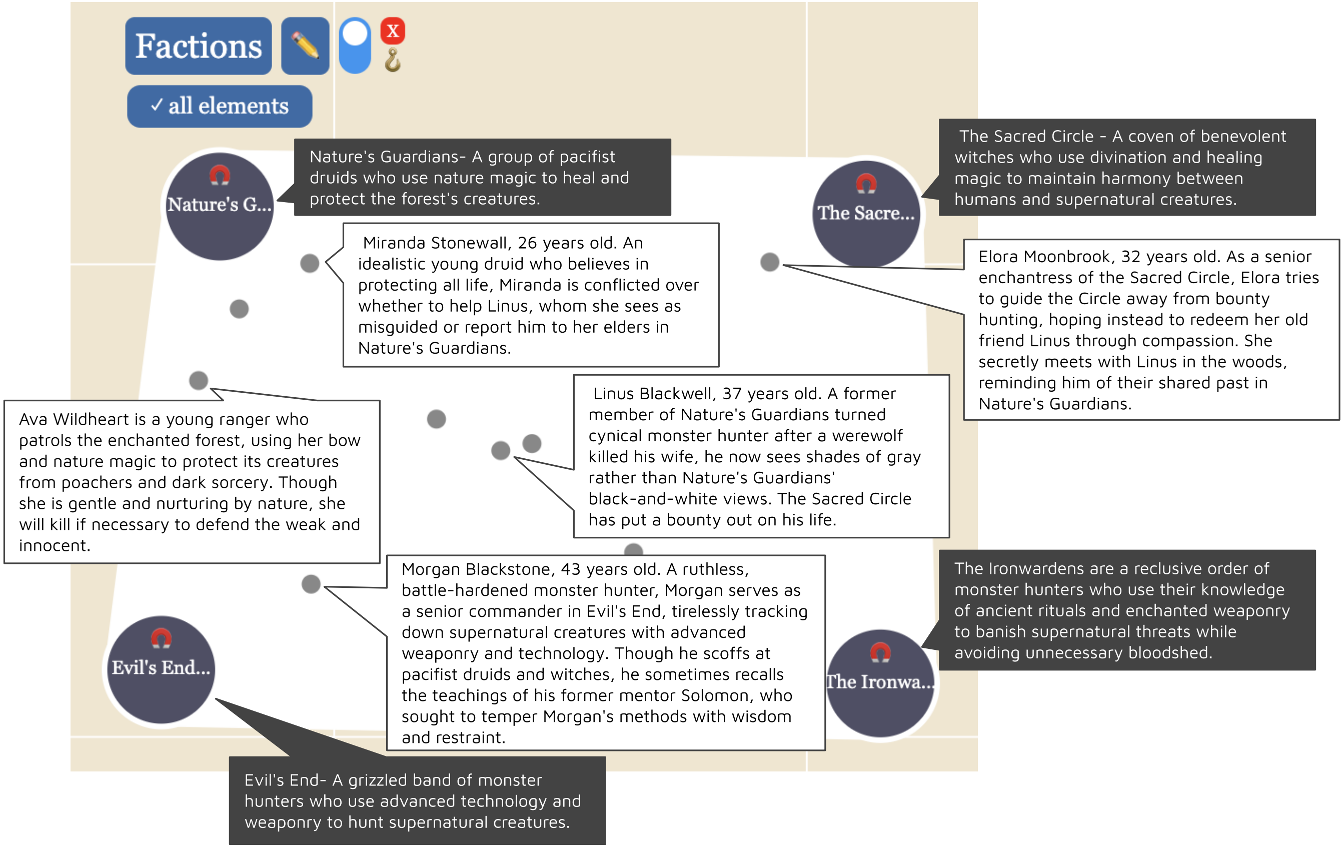
Prompt: Create a character that is equal parts good and evil

A gaunt figure emerges from the shadows, equal parts light and dark, twisting the mystical forces of the universe to its enigmatic will.

c) elements generated with visual steering

Gwendolyn "Gwen" Alder, 25 years old. Warm and idealistic, Gwen's free-spirited nature endears her to the whimsical wood nymphs, though her occasional impulsiveness frustrates the pragmatic Margaret. Her innate goodness intrigues Charlotte, who sees potential to manipulate Gwen's trusting nature to her own ends.

Figure 20: Elements generated by P1, (a) without any steering, (b) with natural language prompting, and (c) with visual steering.



Nature's Guardians- A group of pacifist druids who use nature magic to heal and protect the forest's creatures.

The Sacred Circle - A coven of benevolent witches who use divination and healing magic to maintain harmony between humans and supernatural creatures.

Miranda Stonewall, 26 years old. An idealistic young druid who believes in protecting all life, Miranda is conflicted over whether to help Linus, whom she sees as misguided or report him to her elders in Nature's Guardians.

Elora Moonbrook, 32 years old. As a senior enchantress of the Sacred Circle, Elora tries to guide the Circle away from bounty hunting, hoping instead to redeem her old friend Linus through compassion. She secretly meets with Linus in the woods, reminding him of their shared past in Nature's Guardians.

Ava Wildheart is a young ranger who patrols the enchanted forest, using her bow and nature magic to protect its creatures from poachers and dark sorcery. Though she is gentle and nurturing by nature, she will kill if necessary to defend the weak and innocent.

Linus Blackwell, 37 years old. A former member of Nature's Guardians turned cynical monster hunter after a werewolf killed his wife, he now sees shades of gray rather than Nature's Guardians' black-and-white views. The Sacred Circle has put a bounty out on his life.

Morgan Blackstone, 43 years old. A ruthless, battle-hardened monster hunter, Morgan serves as a senior commander in Evil's End, tirelessly tracking down supernatural creatures with advanced weaponry and technology. Though he scoffs at pacifist druids and witches, he sometimes recalls the teachings of his former mentor Solomon, who sought to temper Morgan's methods with wisdom and restraint.

The Ironwardens are a reclusive order of monster hunters who use their knowledge of ancient rituals and enchanted weaponry to banish supernatural threats while avoiding unnecessary bloodshed.

Evil's End- A grizzled band of monster hunters who use advanced technology and weaponry to hunt supernatural creatures.

Figure 21: A view created by P2.