WhatELSE: Shaping Narrative Spaces at Configurable Level of Abstraction for AI-bridged Interactive Storytelling

Zhuoran Lu
Autodesk Research
Toronto, Ontario, Canada
Computer Science
Purdue University
West Lafayette, Indiana, USA
lu800@purdue.edu

Qian Zhou
Autodesk Research
Toronto, Ontario, Canada
qian.zhou@autodesk.com

Yi Wang
Autodesk Research
San Francisco, California, USA
yi.wang@autodesk.com

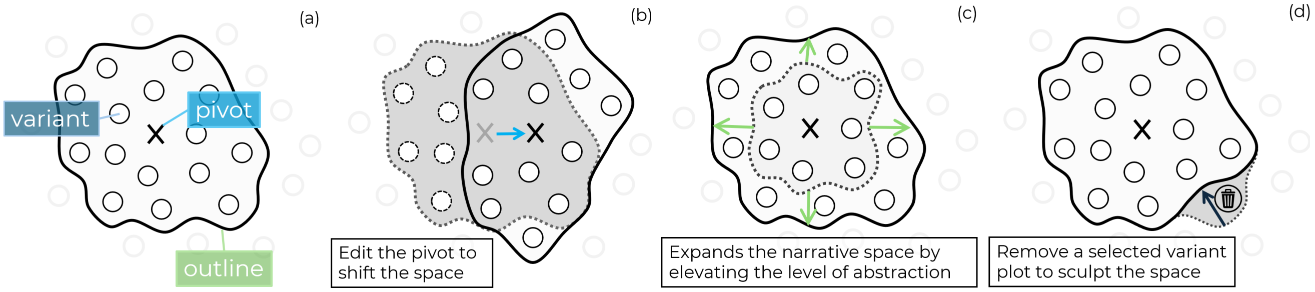

(a)
variant
pivot
outline
(b)
Edit the pivot to shift the space
(c)
Expands the narrative space by elevating the level of abstraction
(d)
Remove a selected variant plot to sculpt the space

Figure 1: We present WhatELSE, an interactive narrative authoring system that allows users to shape a narrative space using language abstraction. (a) We use the pivot, outline, and variants to describe the narrative space. Users can import an example story as a pivot. The system elevates the pivot into a narrative space. It generates an outline and variants to describe the space. Users can (b) edit the pivot to shift the space, (c) elevate the abstraction level to expand the space, or (d) remove variants to sculpt the space.

Abstract

Generative AI significantly enhances player agency in interactive narratives (IN) by enabling just-in-time content generation that adapts to player actions. While delegating generation to AI makes IN more interactive, it becomes challenging for authors to control the space of possible narratives - within which the final story experienced by the player emerges from their interaction with AI. In this paper, we present WhatELSE, an AI-bridged IN authoring system that creates narrative possibility spaces from example stories. WhatELSE provides three views (narrative pivot, outline, and variants) to help authors understand the narrative space and corresponding tools leveraging linguistic abstraction to control the boundaries of the narrative space. Taking innovative LLM-based narrative planning approaches, WhatELSE further unfolds the narrative space into executable game events. Through a user study (N=12) and technical evaluations, we found that WhatELSE enables authors to perceive and edit the narrative space and generates engaging interactive narratives at play-time.

CHI '25, Yokohama, Japan
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713363

CCS Concepts

• **Computing methodologies** → **Natural language processing**; • **Applied computing** → **Computer games**; • **Software and its engineering** → **Interactive games**.

Keywords

Interactive Narrative, Large Language Models, Abstraction, Narrative Space, Video Games, Generative AI

ACM Reference Format:
Zhuoran Lu, Qian Zhou, and Yi Wang. 2025. WhatELSE: Shaping Narrative Spaces at Configurable Level of Abstraction for AI-bridged Interactive Storytelling. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3706598.3713363

1 Introduction

Interactive Narrative (IN) is a form of digital storytelling experience where the player can influence a dramatic storyline through their actions [27, 63]. IN takes various forms in entertainment and education applications, with the most prominent one being branching storylines in role-playing games [61], where the author predefines the range of player actions and creates multiple storylines reflecting the consequences of different player choices. Instead of one single narrative, the author creates a *narrative space* consisting of all possible storylines a player can experience.

The advancement of Large Language Models (LLMs) has the potential to revolutionize IN by enabling the automatic generation of content based on the user's specifications [42, 44, 89]. This enables just-in-time generation of narrative content to adapt to different game world states. Instead of enumerating all possible storylines, authors can convey their broad narrative intent to the LLM as prompts, and let the model render concrete narrative instances customized by the player's in-game context [10, 36, 57, 79]. For example, Inworld Origin [2] is a narrative-driven adventure game where the player interacts with LLM-driven characters to solve a mystery. The characters respond to the player with unscripted actions and dialogs that are generated by LLMs at play-time, while still adhering to an overall narrative structure. AI-bridged IN significantly enhances player agency as the player can now influence the story in ways not restricted by predefined story branches, and also facilitates a form of emergent narrative [5, 77] where the final narrative experience can even possibly go beyond the author's anticipation.

To create a traditional IN, the author directly specifies **concrete narrative instances** conditioned on game world states, while in the case of AI-bridged IN, the author has to express their narrative intent as prompts - which requires them to write **abstract narrative specifications**. The abstract specification eventually transforms into concrete narrative instances at play-time based on the player's interaction with the game system. For example, instead of specifying a story where *"an ant fell into the water and was then saved by a dove by dropping a leaf to the water"*, the author writes in a prompt *"a story where a small creature got into an accident and was saved by another creature"* so that the exact plot can be generated based on the play-time game world states. Using abstract narrative specification to guide play-time plot progression is an effective way to compactly sketch out a narrative space and impose high-level authorial control.

However, this AI-bridged IN workflow presents challenges both to the author and the AI system, as concrete narrative instances are not accessible at the time of authoring. On one hand, authors struggle to create, perceive, and control the narrative space just by prompting [37]. Once they write prompts with abstract narrative specification, it is difficult for them to envision specific instances within the defined narrative space [35]. On the other hand, it is also difficult for LLMs to unfold the author's narrative intent into a sequence of events that are executable in an external game environment, as LLMs are not trained to simulate the causal dynamics defined by the game mechanism and are known for challenges in maintaining long-term consistency [53].

To address these challenges, we present WhatELSE, an IN authoring system that creates interactive narratives from user-provided example narratives. Instead of writing abstract prompts, authors can import narrative instances as *pivots* to create a narrative space. The system generates an *outline* by abstracting from these instances. To help authors perceive the space, the system uses a simulation process to generate concrete narrative *variants* from the outline-defined space. Together, the system uses *pivot*, *outline*, and *variants* to represent and shape the narrative space: (1) authors can shift the space by directly editing the *pivot* (Figure 1.b), (2) they can expand or constrain the narrative space by changing the *outline's* level of abstraction (Figure 1.c), (3) they can also fine-tune the narrative space by removing *variants* (Figure 1.d).

To support WhatELSE, we developed a technical pipeline that supports the bidirectional transformation between outlines and instances using LLM and narrative planning. To generate instances from an outline, we developed a novel LLM-based narrative planning method following the LLM-Modulo frameworks proposed by Kambhampati et al. [33], taking into account possible play-time world states and player behaviors. Narratives are grounded by character action sequences executable in the game environment and are iteratively generated and reviewed. We leverage an external simulated game environment to guide the validation and revision of plot generation to ensure that the causal dynamics in the game environment are correctly captured by the generated plots.

This work has three main contributions: 1) an IN authoring system that allows users to shape the narrative space at different levels of abstraction using the outline and instances, 2) a technical pipeline that supports bidirectional transformation between outlines and instances using LLMs and narrative planning; and 3) findings from a user study (n=12) and a technical evaluation. Results from the user study showed that WhatELSE helped authors perceive and edit the narrative space. It also demonstrated that the created narrative space could be used to generate engaging, interactive narratives at playtime. Our technical evaluation demonstrated the effectiveness of the pipeline in supporting language abstraction and generating diverse plots that react to player actions.

2 Related Work

2.1 Authoring Interactive Narratives

Various tools have been proposed to support authoring Interactive Narratives (IN) in the past decades [25, 26, 67]. Many of these tools are designed to give better control and management over branching storylines, enabling rich player actions while maintaining authorial control [22]. IN Authoring tools usually organize the narrative space in explicit branching structure [13] to enable intuitive understanding, including flowchart-like structures [22], state machines [24, 80], and collections of modular story chunks conditioned on game world states ("storylet") [38]. Every possible narrative instance is manually authored, which ensures that the author's intent is precisely preserved [63]. Much of the prior work seeks to make the authoring process more efficient through more compact representations of storylines [69]. However, despite these efforts, the complexity of defining all possible plot progressions remains a significant challenge [63]. IN authoring continues to be a time-intensive and engineering-heavy process, often falling short of expectations [26, 71].

Recent efforts have moved towards play-time narrative generation, reducing the need for extensive manual authoring [21, 23]. Authors set high-level narrative requirements, allowing for automated procedural story generation that responds to player actions [64]. While this approach saves manual effort, authors need to express their authorial intent through means other than high-level specifications. To address this challenge, our system proposes various levels of abstraction grounded in the theory of narrative structure [50], allowing authors to express design requirements from beat-level concrete details to story-level flexible goals.

2.2 Narrative Generation

A prominent approach to generating plots is through symbolic narrative planning [4, 41, 51, 66, 87, 96], which explicitly models the story domain and simulates the causal dynamics of possible plot events to guarantee the causal soundness of generated plots in the simulated context. The author describes a desired world state at the end of plot execution as a "narrative goal", and the narrative planning algorithm needs to generate a sequence of state transitions (events) that leads the world state to the narrative goal - called "narrative plans" [41].

Symbolic narrative planning requires a hand-crafted knowledge base using formal logical language that defines preconditions and effects of a predefined action set within a story [58]. Given the extensive engineering work required to construct this knowledge base, the generated plots offer limited complexity and scale.

The advancement of generative AI also leads to LLM-based methods for plot generation. LLMs can be used to design various narrative elements of the game [39, 40, 43], such as character design [15, 16, 19, 48], world setting [31, 60, 70], scenes and plots [4, 17, 54, 95]. They can also be used at playtime to facilitate just-in-time narrative content generation, such as character dialogs [3]. Recent work has also shown that LLMs can be used to drive play-time character behaviors [14, 56, 86], leading to plots naturally emerging from the characters. Unlike symbolic narrative planning, which restricts the expression of authorial intent to formal logical languages, LLM-based methods allow users to write flexible abstract narrative specifications to guide plot generation.

However, LLM-based approaches pose challenges to controllability due to their black-box nature. This lack of control remains a major obstacle for authors looking to adopt LLMs in their workflow [6]. LLM-driven character simulation methods [14, 56, 86] pose even greater challenges for authorial control due to the emergent behaviors of the LLM-powered characters, leaving a "herding cat" problem when generating narratives [88]. To overcome the challenge, effort has been made to incorporate a symbolic representation of events [4], introduce sketching input beyond text [17], and shift AI to an advisor role [67].

In this work, we aim to improve the controllability of narrative generation by providing authors with various writing assistance with a configurable level of abstraction in the outline, instance, and sentence level to specify the desired narrative goals and character behaviors. Our system combines symbolic planning, LLMs, and character simulation to generate narratives with causal soundness and emergent behaviors for flexible plot progression.

2.3 Narrative Space

The concept of narrative space refers to the range of stories that a system can generate [65], possibly conditioned by constraints or requirements from the author. A narrative space can be characterized in different ways. For example, traditional interactive fiction defines its narrative space with explicit branching storylines. The symbolic narrative planning system usually defines its narrative space based on the story domain [58], i.e., the possible events that can happen in a story world derived from the characters, locations, objects, and character action schema.

The narrative space of a generative system based on LLMs is characterized by the LLM model itself and the prompt creation mechanism. Unlike narrative spaces defined by formal representations like branching diagrams and state machines [63], narrative spaces based on natural language have soft boundaries due to the inherent fuzziness of natural language semantics. The open-endedness of LLMs makes such narrative spaces more difficult for the author to perceive. Existing work has explored semantic abstraction of sentences called "events" [4] and "loglines" [53], as a unit of a story that summarizes its central dramatic conflict. Inspired by prior work, we introduce the notion of "outline" as the abstract specification that defines the narrative space for generating IN. This "outline" has a configurable level of abstraction, allowing the author to adjust the granularity of their authorial control.

The concept of narrative space is closely related to the notion of design space [7, 75] or conceptual space [90]. They all refer to a metaphorical space of possibilities - constructed as ideas, designs, solutions, etc. Previous works have been mostly focusing on exploring a design space in search of specific designs to generate candidates from requirements to establish the space [46, 65], traverse from an existing artifact (pivot) to its alternatives (variants) [49, 68], and compare these alternatives in multiple dimensions [73, 76, 97]. Inspired by prior work, we propose the pivot and variants view to describe the narrative space.

Our work differs from prior work by considering the narrative space as the final artifact, rather than an intermediate step in creating a narrative artifact. IN authors create this space for players to explore, making it important for authors to fully understand the narrative space. Creating such space requires balancing between authorial control and emergence [34, 63, 85]. On one hand, the space of possible plots needs to be constrained by the author's narrative intent. On the other hand, the space needs to be sufficiently under-constrained so that the player's action and interactions with characters can be reflected in plots that reside in the space. Balancing these two potentially conflicting objectives is the main challenge in AI-bridged IN authoring.

3 Challenges of AI-Bridged Interactive Narrative Authoring

Interactive Narrative allows players to influence storylines through their actions, with authors creating a narrative space of possible storylines. During the authoring process, authors predefine the range of player actions and creates multiple storylines reflecting the consequences of different player choices. AI-bridged IN generates just-in-time narrative content that adapts to different game world states, freeing authors from enumerating storylines. However, shifting from traditional IN to AI-bridged IN presents challenges for authors in expressing, perceiving, and controlling the narrative space. Authors often struggle to articulate their implicit narrative intents in high-level prompts [53] and may underexpress their intent to AI systems [37]. While novice authors might start with a concrete example [52, 81], a single narrative instance can be both overly detailed and insufficient, as it includes unnecessary specifics and lacks broader context [37]. Therefore, neither concrete instances nor abstract specifications alone are ideal for defining a narrative space. Instead, the ability to configure the level of abstraction is necessary to support AI-bridged IN authoring.

On the other hand, once a narrative space is defined via prompts, the author has limited insight into the player experience, as players are responded with unscripted character actions and dialogs generated by LLMs at play-time. It is difficult to identify and prevent the deviations beyond the the author's narrative intent. Therefore, it is important to provide valuable information on how different types of player could react to instances, through which authors could preview the narrative instances as they get unfolded in the player experience [36]. Furthermore, transforming prompts into meaningful game plots is not trivial. It requires effective narrative planning to generate causally sound event sequences. Central to this success is modeling the logical causal progression of the game plot [66]. However, LLMs are not natively planners in creating causal progression and have been found to cause hallucinations without external verifier to validate the coherency and executability of the generated plan [33]. Motivated by the challenges unresolved in AI-bridged IN, we developed the following design goals to guide the design of system:

DG1: Enable users to perceive the narrative space. The narrative space in AI-bridged IN contains various possible storylines, which are generated at play-time based on player actions. Authors might struggle to envision what types of variations would be possible. The system should provide representations that can help authors to explore and understand the narrative space.

DG2: Support configurable level of abstraction in editing narrative space. Concrete instances can be overly detailed, while abstract specifications can be too vague. Supporting users to adjust the level of abstraction helps them balance between details and abstraction, which allows the narrative instances to emerge from player interactions, while still adhering to the narrative structure.

DG3: Generate meaningful game events that react to player actions at play-time. An engaging player experience requires the generated plots to represent logical causal progression that follows the game mechanism. The proposed system should support simulating casual dynamics and generate meaningful narrative content based on player actions.

4 WhatELSE: System Design and Implementation

In this section, we present the interface and features of WhatELSE, describe its technical pipeline to facilitate the transformation between narrative instances and narrative outlines, and demonstrate its workflow with an example user story.

4.1 Narrative Space Editor Interface

WhatELSE system assists the user in creating a narrative space. The user can upload narrative examples in text file(s). In addition, the user uses a sentence to describe a story's moral (e.g., *"kindness is never wasted"*). The system uses the story input to construct an initial version of the narrative space. The user can edit this narrative space using the interface. WhatELSE features three views for the user to perceive the narrative space: **Pivot View**, **Outline View**, and **Variants View** (Figure 2).

Pivot View The pivot view shows a pivot narrative instance. A *pivot (narrative instance)* is a user-defined narrative instance, considered as a representative instance in the narrative space. By default, the user's input is automatically marked as the pivot. The pivot serves as a point of reference as the user edits the narrative space.

Outline View An *outline* is an abstract specification of a sequence of events defining the narrative space. Similarly to "loglines" [53], it specifically describes the general structure of the narrative with a sequence of high-level events - e.g., *"A small creature runs into an accident. It was then saved by another creature"*. The outline describes the narrative space from a broader perspective by capturing the commonality across all the narrative instances in the narrative space. It represents the most abstract manifestation of the author's narrative intent, thus defining the boundary of the narrative space.

Variants View A *variant (narrative instance)* is a narrative instance residing in the current narrative space. A variant instantiates the outline with a sequence of concrete events - e.g., *"An ant fell into water. A dove dropped a leaf next to the ant. The ant climbed on the leaf. The ant was saved."* Each abstract event in the outline is expanded to multiple concrete events in a variant.

The variants are displayed in an interactive scatter plot along two dimensions to help users understand the shape of the narrative space: 1) the **authorial intent** dimension, measured by the distance between the moral expressed by the variant and by the pivot (ranging from 0 to 1)[1], and 2) the **emergence** dimension, measured by how much the plot progress in the variant deviates from the pivot (ranging from 0 to 1). These two dimensions are inspired by the "authorial intent" dimension in Riedl's taxonomy of IN approaches [62, 63], as well as the notion of "emergence" [83] and "interactivity" [72] from prior IN research. Users can configure the number of variants to be generated for visualization. Users can click on any variant in the visualization to display its detailed content, allowing them to compare it with the pivot. Users can also use a scroll bar to visualize the plot progression as they develop across different stages, allowing them to perceive how the narrative variants evolve over time and deviate from the pivot.

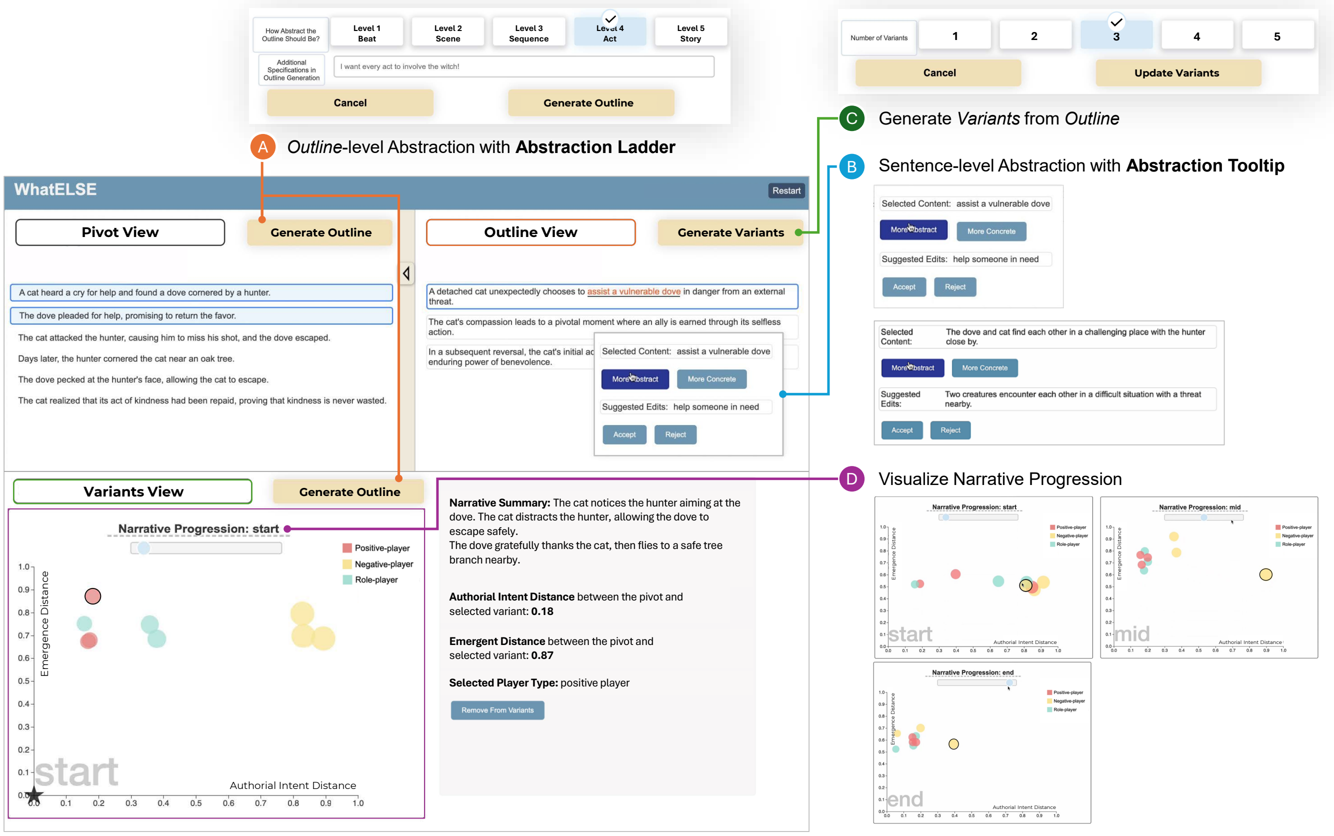
These three views provide different perspectives for users to inspect the narrative space. We also provide editing tools at each view to support shaping the narrative space in different ways.

4.1.1 *Support Editing the Narrative Space.*

The system provides editing tools at outline and instance level.

Outline Editing Users can constrain or relax the boundary of the narrative space by adjusting the outline's level of abstraction. The more abstract the outline is, the less constrained the narrative space is. For example, *"a small creature got into an accident"* is more abstract than *"the ant fell into water"*, enabling more possible narrative instances to be generated. The former removes the constraint on *"the small creature"* being *"the ant"*, and the *"accident"* being *"falling into water"*. A less constrained narrative space allows stronger player agency but follows a looser authorial structure. Outline editing allows the user to tune the narrative space to reach a desired balance between authorial structure and player agency. The system provides two tools to support the abstraction editing.

[1] This distance is evaluated by prompting the LLM to assess how well the moral is conveyed



Additional Specifications in Outline Generation: I want every act to involve the witch!

A cat heard a cry for help and found a dove cornered by a hunter.

The dove pleaded for help, promising to return the favor.

The cat attacked the hunter, causing him to miss his shot, and the dove escaped.

Days later, the hunter cornered the cat near an oak tree.

The dove pecked at the hunter's face, allowing the cat to escape.

The cat realized that its act of kindness had been repaid, proving that kindness is never wasted.

A detached cat unexpectedly chooses to assist a vulnerable dove in danger from an external threat.

The cat's compassion leads to a pivotal moment where an ally is earned through its selfless action.

In a subsequent reversal, the cat's initial ac enduring power of benevolence.

Selected Content: assist a vulnerable dove

Suggested Edits: help someone in need

Selected Content: The dove and cat find each other in a challenging place with the hunter close by.

Suggested Edits: Two creatures encounter each other in a difficult situation with a threat nearby.

Narrative Summary: The cat notices the hunter aiming at the dove. The cat distracts the hunter, allowing the dove to escape safely.
The dove gratefully thanks the cat, then flies to a safe tree branch nearby.

Authorial Intent Distance between the pivot and selected variant: **0.18**

Emergent Distance between the pivot and selected variant: **0.87**

Selected Player Type: positive player

Figure 2: An illustration of the Narrative Space Editor interface, including the pivot, outline, and variants view. Users can (A) generate outline from pivot or variants with an abstraction ladder to configure the abstraction level. They can (B) fine-tune sentence or word-level abstraction using an abstraction tooltip. They can also (C) generate variants from outline specifying the number of variants in the variants view. They can use (D) narrative progression slider to visualize the variants' dynamic distance from the pivot (star).

- **Abstraction Ladder (Figure. 2.A)** The abstraction ladder helps the user to shift the global level of abstraction across the events in the outline. Inspired by theories of narrative structure [50, 74], this ladder covers a spectrum of abstraction levels (beat, scene, sequence, act, and story level). An outline at the beat level is similar to a narrative instance, while an outline at the story level summarizes the plot into a one-line overview. Between the two ends, each level of abstraction is progressively more abstract than the previous level. For instance, a scene-level outline provides detailed descriptions of specific scenes, including characters, actions, objects, etc: *"The kind dove takes a leaf to reach the ant and drags it out of a water bubble."* An act-level outline offers a highly summarized view of the narrative, focusing on the turning points: *"A character saves their friend from danger."*
- **Abstraction Tooltip (Figure. 2.B)** The abstraction tooltip allows the user to adjust the sentence, phrase, or word-level abstraction in a more fine-grained manner. Practically, when users select a text snippet in their outline plots, the tooltip appears, offering two options: "More Abstract" and "More Concrete". By clicking the button, users receive suggested edits that replace the selected content with a more abstract or more concrete phrase. While the abstraction ladder provides global control over the entire outline, the tooltip enables more fine-grained adjustments at the word or phrase level. The suggestion of making the selected content more abstract or more concrete is based on the taxonomy in linguistics [30]. For example, *"character-animal-small animal-cat-tabby cat"* constructs a linguistic hierarchy. Given a selected text snippet *"cat"*, requesting a more abstract suggestion would yield its superordinate term *"small animal"* or *"animal"*, while a more concrete suggestion would provide its subordinate *"tabby cat"*.

Once the user is satisfied with the outline, they can click the "Generate Variants" button to generate narrative variants in the Variant View. Section 4.2.2 describes the technical pipeline for generating narrative instances from outline.

Instance Editing Users can fine-tune the narrative space by editing the instance-level content in Pivot and Variant View. They can select a variant to set or unset it as the pivot. They can also remove a variant from the narrative space or add it back. Finally, they can directly edit the text in the instances. They can click the "Generate Outline" button to update the outline based on their edited variants.

For example, a user who does not want to include certain player type may choose to remove all variants by that player type and update the outline. Section 4.2.2 describes different player types in the player proxy model.

4.2 Technical Pipeline

This section describes our technical pipeline supporting the features described in the above section, focusing on the transformation between narrative outline and narrative instances. Specifically, we employ the GPT-4o [55] for the implementation of our system.

4.2.1 Transforming Narrative Instances to Outline. We use an LLM prompting pipeline to generate outlines from narrative instances (Figure 3.1). This pipeline first prompts the LLM with domain knowledge in drama writing, providing the context of the story domain and the narrative instances. The pipeline then prompts the LLM to summarize the commonalities across these narrative instances, generating outlines at different abstraction levels based on story structure principles [50]. Finally, the system selects an outline according to the user's chosen level of abstraction.

4.2.2 Transforming Outline to Narrative Instances. To generate meaningful events that can react to player actions (DG3), we go beyond text generation and integrate an LLM-based narrative planning approach with character simulation and player proxy models. Our method extends *StoryVerse* [85] with player interactivity and behavior modeling. Generating narrative instances from outline is essentially simulating an interactive story generation process, where player actions may be generated by computational proxies of players, and the story generated grounded in the causal changes of game world states in accordance with the game mechanism.

To explain this process, we assume a *Game Environment* (Figure 3.4) is given, which contains the *Story Domain* and maintains the *World State*. The *World State* consists of a collection of variables that hold relevant values for the game mechanics, such as the characters' attributes (e.g., health points), current locations, and relationship scores, as well as their memories from the simulation. The *Story Domain* contains a set of characters, locations, and an action schema that specifies executable actions in the game system. These actions are implemented as executable function calls that modify the variables of *World State* accordingly. For example, executing the action `kill(X)` will result in character `X`'s state to become dead.

The main game loop starts by sending an event from the outline to the Interactive Narrative Compiler (Figure 3.2) to instantiate a sequence of character actions (Figure 3.3). The Game Environment (Figure 3.4) executes the actions and updates the world states resulting from the generated character actions. Once the Game Environment executes the actions, the player (or a simulated player) can interfere with the game by changing the world states, such as saving a character (Figure 3.5). Finally, the Game Environment sends the updated world states and outline back to IN Compiler for the next iteration. The process loops over the events in the outline plot, and stops when it exhausts all the events.

Plot Generator Given an event in the outline, the system generates a sequence of character actions that act out the event. It takes into account the current game world state as a result of all previous plot executions and player actions. An LLM is prompted to generate a sequence of character actions that act out the event. The prompt includes the following information from the game environment:

- the list of characters and their descriptions;
- the action schema;
- current world state (including character's memory).

This process is very similar to narrative planning which generates a sequence of state transitions that leads to a narrative goal. Compared to classic symbolic narrative planning, our narrative goal may be fuzzier - sometimes it may not be rigidly characterizable by world states. For example, the narrative goal could be *"everyone likes Bob"*, which corresponds to world states semantically in a fuzzy way. This narrative goal can also be any arbitrary statements describing the desired outcome, constraining not only the endings but also the transitions, for example, *"someone was careless and got into an accident"*. Therefore, we use an LLM-based method instead of existing symbolic narrative planning methods for transforming outlines into concrete plots.

Plot Reviewer Similar to symbolic planning problems, the plot generation requires causal soundness. This means the character actions must be valid state transitions according to the game's causal rules. We thus adopt an LLM-based planning method following the LLM-Modulo framework [33], with a game environment simulating plans generated by LLMs and providing external critiques. To review the generated plan, an LLM is prompted to provide feedback regarding the quality and feasibility of the action sequence to improve it:

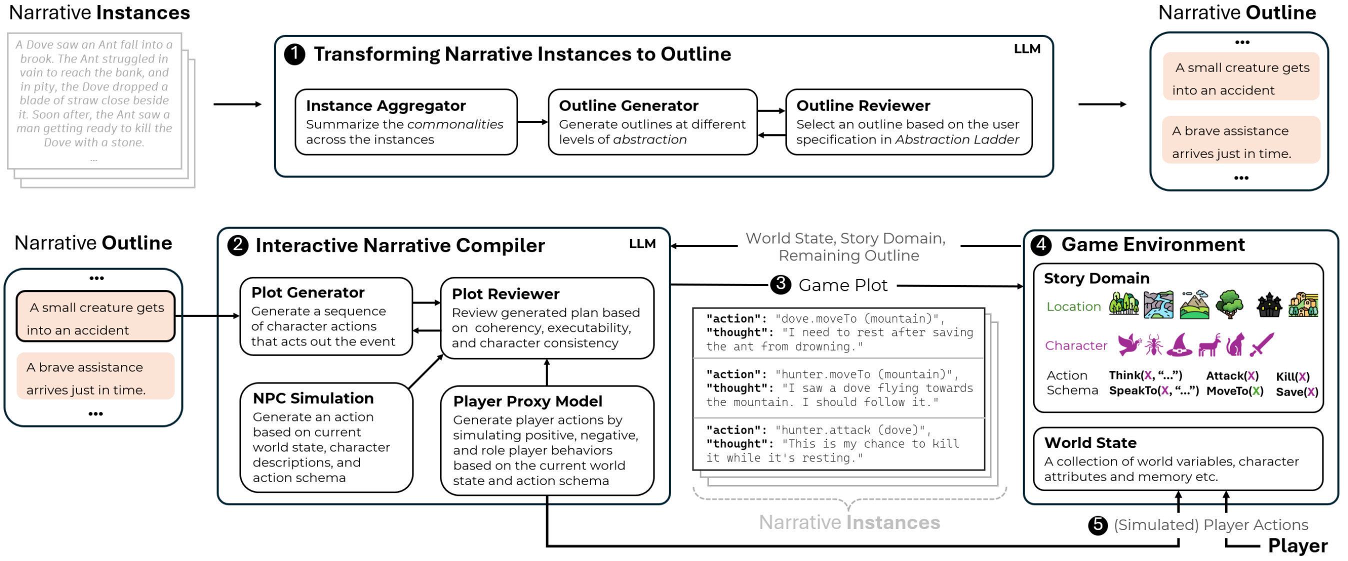
- **Overall Coherency Evaluation** Feedback is obtained by prompting an LLM to comment on the overall coherency of the generated plot and make suggestions for improvement.
- **Character Simulation Evaluation** For every action in the sequence, we prompt an LLM to play the role of the subject of the action. Given the current world state including the character's memory, we ask the LLM if the motivation for the character to perform the action has been established. We include the explanation to this question in the feedback if the motivation has not been established.

In addition, we leverage a simulated Game Environment for external evaluation. The system evaluates the **Causal soundness** of the generated action sequence and reports the observations on the success/failure of the execution. The combined feedback is added to the prompt for the next round of generation.

For example, the event *"a small creature gets into an accident"* could be turned into a sequence of character actions shown in Figure 3.3. Note that the events in the outline plot are at a higher abstraction level. This means the same event can be transformed into multiple character action sequences, leaving room for the exact plot to adapt to different world states [2]. Once the final sequence of character actions is generated, it will be executed by the *Game Environment* to update the world state.

The Plot Generator and Reviewer create a sequence of character actions to act out the event. In between these events, the player or

[2] In the above example, if the dove is dead at the time of plot execution, then a different character action sequence will be generated - one possibility is that the ant fell into the water.


Narrative Instances
A Dove saw an Ant fall into a brook. The Ant struggled in vain to reach the bank, and in pity, the Dove dropped a blade of straw close beside it. Soon after, the Ant saw a man getting ready to kill the Dove with a stone.
...
1 Transforming Narrative Instances to Outline
LLM
Instance Aggregator
Summarize the commonalities across the instances
Outline Generator
Generate outlines at different levels of abstraction
Outline Reviewer
Select an outline based on the user specification in Abstraction Ladder
Narrative Outline
A small creature gets into an accident
A brave assistance arrives just in time.
Narrative Outline
A small creature gets into an accident
A brave assistance arrives just in time.
2 Interactive Narrative Compiler
LLM
Plot Generator
Generate a sequence of character actions that acts out the event
Plot Reviewer
Review generated plan based on coherency, executability, and character consistency
NPC Simulation
Generate an action based on current world state, character descriptions, and action schema
Player Proxy Model
Generate player actions by simulating positive, negative, and role player behaviors based on the current world state and action schema
World State, Story Domain, Remaining Outline
3 Game Plot
Narrative Instances
4 Game Environment
Story Domain
Location
Character
Action Schema
Think(X, "...")
SpeakTo(X, "...")
Attack(X)
MoveTo(X)
Kill(X)
Save(X)
World State
A collection of world variables, character attributes and memory etc.
5 (Simulated) Player Actions
Player

```
"action": "dove.moveTo (mountain)",
"thought": "I need to rest after saving
the ant from drowning."

"action": "hunter.moveTo (mountain)",
"thought": "I saw a dove flying towards
the mountain. I should follow it."

"action": "hunter.attack (dove)",
"thought": "This is my chance to kill
it while it's resting."
```

Figure 3: An overview of the technical pipeline of WHATELSE. (1) The system transforms narrative instances to an outline using the LLM to summarize their commonalities, generate outlines at different levels of abstraction, and review the outline based on user specifications in the Abstraction Ladder. (2) The Interactive Narrative Compiler unfolds the outline into (3) a sequence of character actions to act out the events in the outline. (4) The Game Environment executes the actions and updates the world states. (5) The player (or a simulated player) can interfere with the game by changing the world states. Finally, the Game Environment sends the updated world states and outline back to the compiler for the next iteration.

NPCs take free actions. These actions are driven by the LLM. The player actions are determined either by a real player's input or a simulated Player Proxy Model (Figure 3.5).

Player Proxy Model When generating narrative variants, player actions are provided by an LLM-based player proxy model which operates based on player behavior classification derived from previous studies in digital games [92, 94]. Our player simulation incorporates three key player behavior models:

- **Positive Players** in digital games contribute positively by following the intended game objectives and exhibit helping behaviors [11, 82].
- **Negative Players** are the killers identified in classic player modeling [28, 47]. They often exhibit aggressive behavior that disrupts the experience of others, particularly when they seek to dominate or harm others destructively.
- **Role Players** prioritize narrative immersion and character development by mimicking the actions their character would take in the gaming world [59].

Using these player models, we simulate a potential plot that could emerge from the interaction between game characters and simulated players within the narrative space defined by the outline. In this way, the system generates a diverse set of narrative instances in the variants view.

Non-Player Character Simulation An LLM is prompted to role-play as each of the NPCs and generate an action for this character. The prompt includes the following information:

- the action schema;
- the list of characters and their descriptions;
- current world state (including character's memory);

Note that the character actions are not directly determined by the outline. However, it is affected by the current world state and, therefore, indirectly influenced by the executed events in the outline.

Using this pipeline, WHATELSE creates a gameplay experience by unfolding the outline into narrative instances. The system generates the game plot for each event in the outline as a series of character actions. The player then inputs actions within the action schema, which influence the progression of the subsequent plot. The system runs executable actions to update the world state. After each round of player action, the system unfolds the next events until exhausting all the events in the outline, in this way, creating an interactive narrative experience.

4.3 Example Workflow

Below we present an example workflow to demonstrate some of the features described above. Alice, a novice text-adventure game designer, wants to create a game based on the setting of a novel she enjoys. Alice opens WHATELSE, along with a game engine preloaded with a story domain based on the novel.

4.3.1 Encode Authorial Intent in Narrative Space. Alice starts with a rough draft of the story and a moral she wants to convey: *"Kindness is never wasted"*. Using WHATELSE, she uploads her initial story into the system (Figure 4.a). The story is displayed in the pivot view, showing a sequence of events; while an initial outline appears on the right, summarizing the key turning points (Figure 4.b). Alice adds details to the pivot to refine her story. Once satisfied, she clicks the Generate Outline button to update the outline based on her edits. She chooses the "act level" and specifies, *"The hunter has to appear in every act"*. Alice hovers to see how each event in the outline is mapped to the entries in the pivot plot. She continues exploring different levels to find the ideal level of abstraction.

Alice finds one of the events (*"The peaceful life is threatened by an unexpected danger from the hunter"*) to be too restrictive for the hunter to cause the danger. She uses the abstraction tooltip to replace the phrase *"the hunter"* with *"a character"* to leave room for variations in the game. Alice looks at the outline and is unsure what players might experience. So she clicks the Generate Variants button. The interface displays a scatter plot of potential narrative instances. Alice scrolls through different plot stages of these instances — from start to end — she notices that some instances continuously express the moral, while others only reveal it toward the end, both of which she considers acceptable. However, she also spots a cluster of instances that fail to express the moral by the end of the narrative. Curiously, she clicks on a dot representing one of these instances and reviews its details.

Alice reads the instance and realizes the issue is in the event that she had previously set as *"a character"*, which was too loosely defined, allowing the system to choose an undesirable character. To address this issue, she changed it back to *"a human character with power"*, allowing the system to choose a character reasonable for the second event.

Later, Alice notices a set of three variants where one of the events unfolds as, *"the dove speaks with the hunter, leading the hunter to notice and then chase the dove"*. Alice finds this version more compelling than her pivot plot. She removes other variants, only leaving these three narrative variants in the view. Satisfied with these variants, she clicks the Generate Outline button to create a new outline that summarizes their commonalities. She then returns to the outline editor, using the abstraction tools to iteratively edit the outline, until it aligns with the story's moral and represents a narrative space that incorporates the interesting variations.

4.3.2 Unfold the Narrative Space For interactivity. With the narrative space defined by the outline, Alice can experience the narrative instances unfolding in a turn-based text adventure game. She goes to the interactivity page. The system loads the story domain that includes a set of characters, locations, and action schema (Figure 4.d). The first sequence of the game plot is generated: a hunter is looking for food and finds a deer to hunt (Figure 4.c).

Alice, playing as the dove, chooses her next moves from a pin pad (Figure 4.f). She can bravely stop the hunter by giving out her food (Figure 4.e1). Alternatively, she could ask other characters for help (Figure 4.e2). The system compares the player's action with the narrative outline. If the player chooses to save the deer on their own (Figure 4.e1), the event of *"brave assistance"* is fulfilled by the player action. If the player chooses other actions, the system will create an event where another character demonstrates *"brave assistance"* (the witch in Figure 4) to fulfill the event. The system generates subsequent character behaviors based on the player action.

This turn-based interaction continues, with Alice alternating between reviewing generated game plots, observing character simulations, and experiencing the generated game play as a player. Since she wrote a total of five events in her outline, the game play proceeds for five rounds, until all the events she planned have been played out. Since the game plots generated are fully structured, Alice can directly export the output of the narrative compiler as a finite state machine into the game engine where she can visualize the characters and locations.

4.3.3 Additional Use Cases. In addition to Alice's case as a text-adventure game designer, WhatELSE can also serve as a powerful tool for a wide range of users. Game masters, mod developers, and fan creators across different domains can leverage its capabilities. For example, dungeon masters in tabletop role-playing games can use the Narrative Space Editor to outline gameplay scenarios before sessions and employ the Interactive Narrative Compiler to dynamically determine outcomes of player actions during gameplay. Fan creators [9] can efficiently transform their favorite novels, movies, or other media into interactive narratives, using the WhatELSE to structure and unfold new, personalized storylines based on the original story domain. Beyond entertainment, educators can utilize WhatELSE to design interactive learning experiences, such as gamified learning tutorials or interactive training modules.

5 User Study

We conducted a user study with 12 participants to evaluate WhatELSE. The goals of the study were to (1) understand their usage and perception of the Narrative Space Editor, (2) validate the effectiveness of Interactive Narrative Compiler in preserving authorial intent, and (3) explore the benefits and drawbacks of the AI-bridged IN creation workflow.

5.1 Participants

We recruited 12 participants (6 male, 6 female, between 20 to 34 years old, average 27 years) from within our institution by posting on an internal network channel. The demographic information of the participants was provided in the supplementary material. Participants were asked to self-report their experience in using generative AI (e.g., ChatGPT), digital games, and interactive narratives. Each was compensated with a $75 gift card. All participants reported at least moderate experience in using generative AI. More than half of the participants used to experience interactive narratives. Five participants reported spending more than two hours a week playing digital games. Nine participants reported having no experience creating interactive narratives, while the other three reported having very limited experience.

5.2 Task

The task is to create a playable game plot using the systems provided. Given a story domain and a narrative example with a story moral, participants were asked to generate an outline to guide the creation of a game plot that expresses the same moral as the given narrative instance. The task uses the *"Fairytale Forest"* story domain (Figure 5.3). It contains six characters (ant, dove, hunter, witch, cat, and deer), five locations (mountain, forest, village, brook, and witch house), and an action schema with six actions ("move to", "speak to", "kill", "attack", "think", "save") that defines what characters can perform. Based on the story domain, we adapted two stories widely used in previous studies as narrative examples [1, 78]. These stories

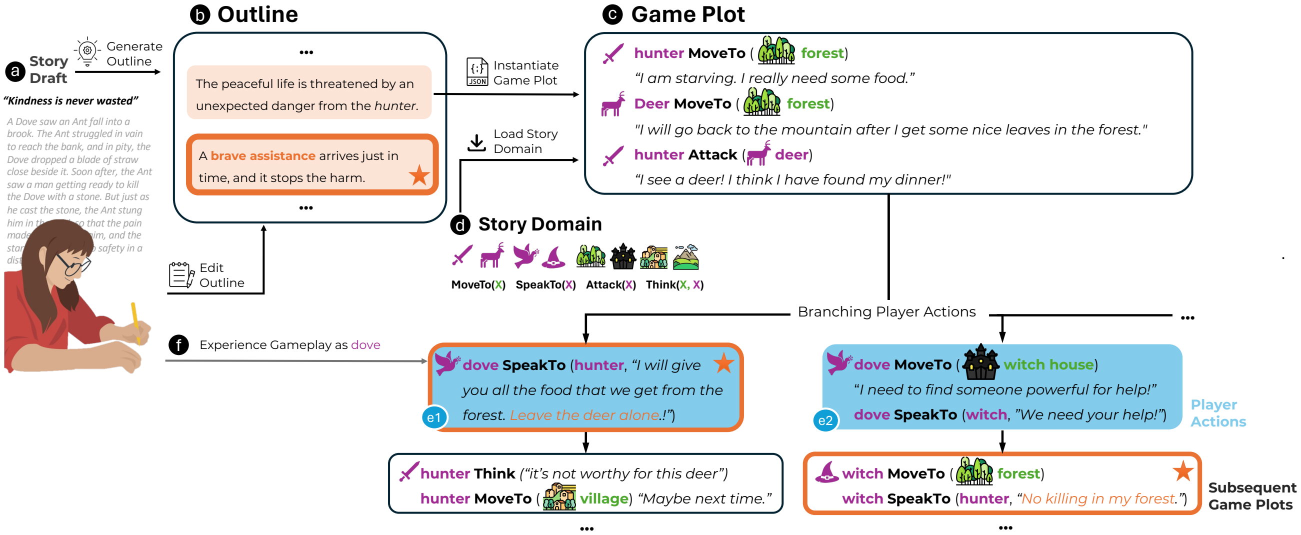

a Story Draft
Generate Outline
"Kindness is never wasted"
A Dove saw an Ant fall into a brook. The Ant struggled in vain to reach the bank, and in pity, the Dove dropped a blade of straw close beside it. Soon after, the Ant saw a man getting ready to kill the Dove with a stone. But just as he cast the stone, the Ant stung him in th so that the pain made im, and the star safety in a dis
b Outline
...
The peaceful life is threatened by an unexpected danger from the hunter.
A brave assistance arrives just in time, and it stops the harm. ★
...
Edit Outline
Instantiate Game Plot
Load Story Domain
c Game Plot
hunter MoveTo (forest)
"I am starving. I really need some food."
Deer MoveTo (forest)
"I will go back to the mountain after I get some nice leaves in the forest."
hunter Attack (deer)
"I see a deer! I think I have found my dinner!"
d Story Domain
MoveTo(X) SpeakTo(X) Attack(X) Think(X, X)
Branching Player Actions
...
f Experience Gameplay as dove
e1 dove SpeakTo (hunter, "I will give you all the food that we get from the forest. Leave the deer alone.!") ★
e2 dove MoveTo (witch house)
"I need to find someone powerful for help!"
dove SpeakTo (witch, "We need your help!")
Player Actions
hunter Think ("it's not worthy for this deer")
hunter MoveTo (village) "Maybe next time."
...
witch MoveTo (forest) ★
witch SpeakTo (hunter, "No killing in my forest.")
...
Subsequent Game Plots

Figure 4: An example workflow that shows (a) an author uploads a story draft in WhatELSE to (b) generate an outline. The system unfolds the outline into (c) an executable game plot with (d) a pre-loaded story domain, which supports branching storylines based on the player actions. If the player chooses to (e1) save the deer from the hunter, this action fulfills the "brave assistance" event in the outline defined by the author (shown as the orange star). If the player chooses to (e2) ask another character (e.g. a witch) for help, the witch will instead save the deer, demonstrating "brave assistance" to fulfill the event. Alternatively, if the player does not choose to save the deer at all, the system will choose a character from the story domain to save the deer as a demonstration of "brave assistance". This example shows how the game plot is dynamically adjusted based on the player actions to fulfill the outline. (f) The author can play the game plot to better understand the player experience.

feature clear plots centered on emerging danger and the characters' decisions to help each other. The two narratives convey two distinct morals: *"Kindness is never wasted"* and *"Kindness is not always rewarded"*. Details of story domain and examples can be found in the supplementary material. We chose this story domain and narrative examples because they are simple and easy to follow, making it suitable for novice users with limited experience in IN. We used the pre-defined story moral as a representation of the authorial intent.

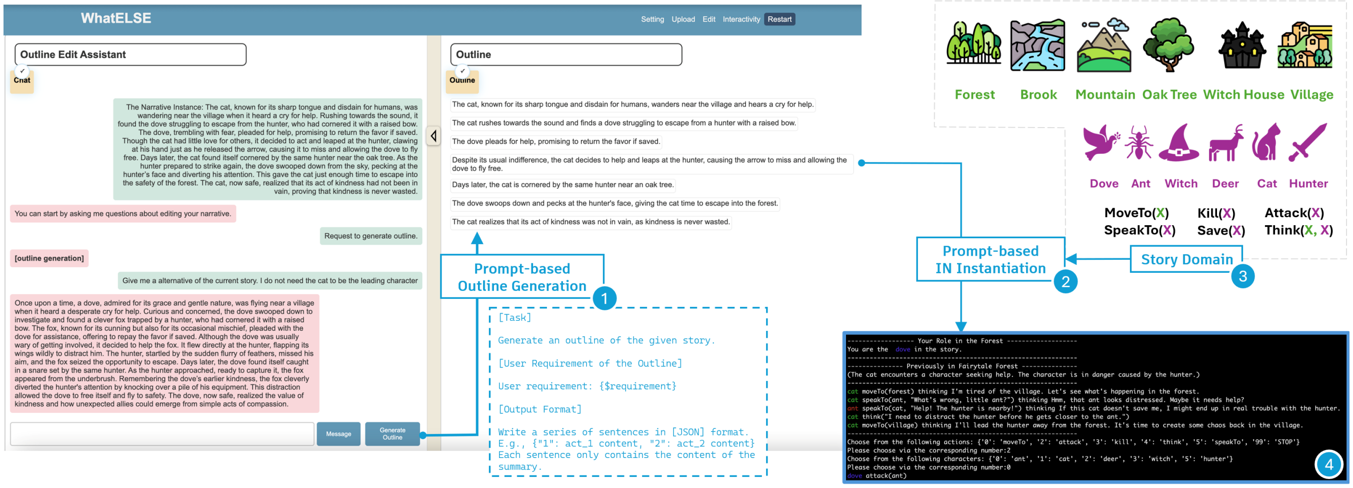
To complete the task, participants are encouraged to meet these criteria for their final outline: (1) The outline should consist of 3–4 sentences. (2) Each sentence must have at least one abstract element, allowing multiple events to fit the description. (3) The outline should convey the same moral as the provided narrative instance. (4) Participants are encouraged to make creative edits to the narrative instance, ensuring that their authorial intent is reflected in the outline. These criteria guide participants by providing clear objectives and controlling the process.

5.3 Conditions

We compared WhatELSE with a baseline system that uses a conversational LLM-powered assistant in a counterbalanced within-subject design. This comparison aimed to evaluate the effectiveness of WhatELSE in supporting two features: (1) generating outlines from a narrative instance and (2) instantiating narratives from the outline (IN Compiler). Specifically, neither of the features is supported by traditional IN authoring tools such as Twine [22]. They are also not natively supported by a plain ChatGPT-like interface. Therefore, we implemented a baseline system with these two features empowered by LLMs for a fair comparison. Participants could load a narrative instance and use the chatbox with a send button to prompt natural language queries. By pressing the "Generate Outline" button, an auto-generated outline would be displayed on the right. This outline generation used a prompt-based approach that loaded the chatbox history as requirements and outputted the outline in JSON format (Figure 5.1). Participants can also directly edit the generated outline. To instantiate narratives from the outline, we implemented a prompt-based plot generation approach (Figure 5.2, prompt in Appendix) similar to prior work [84], with preloaded knowledge of the story domain on characters, locations and action schema (Figure 5.3). During runtime, the LLM acts as a text adventure game engine, dynamically generating game plots based on the outline and responding to player input in the baseline system (Figure 5.4).

5.4 Procedure

All 12 studies were hosted in person. After obtaining consent, an experimenter introduced the background of IN, including the concepts of narrative instance, narrative space, and playable game plots. Participants were oriented to complete the task by the experimenter, who provided instructions, answered questions, and offered help throughout the study. They were informed of two systems. Each system has two stages: creating an outline and playing the game plot generated from it.



The Narrative Instance: The cat, known for its sharp tongue and disdain for humans, was wandering near the village when it heard a cry for help. Rushing towards the sound, it found the dove struggling to escape from the hunter, who had cornered it with a raised bow. The dove, trembling with fear, pleaded for help, promising to return the favor if saved. Though the cat had little love for others, it decided to act and leaped at the hunter, clawing at his hand just as he released the arrow, causing it to miss and allowing the dove to fly free. Days later, the cat found itself cornered by the same hunter near the oak tree. As the hunter prepared to strike again, the dove swooped down from the sky, pecking at the hunter's face and diverting his attention. This gave the cat just enough time to escape into the safety of the forest. The cat, now safe, realized that its act of kindness had not been in vain, proving that kindness is never wasted.

You can start by asking me questions about editing your narrative.

Request to generate outline.

[outline generation]

Give me a alternative of the current story. I do not need the cat to be the leading character

Once upon a time, a dove, admired for its grace and gentle nature, was flying near a village when it heard a desperate cry for help. Curious and concerned, the dove swooped down to investigate and found a clever fox trapped by a hunter, who had cornered it with a raised bow. The fox, known for its cunning but also for its occasional mischief, pleaded with the dove for assistance, offering to repay the favor if saved. Although the dove was usually wary of getting involved, it decided to help the fox. It flew directly at the hunter, flapping its wings wildly to distract him. The hunter, startled by the sudden flurry of feathers, missed his aim, and the fox seized the opportunity to escape. Days later, the dove found itself caught in a snare set by the same hunter. As the hunter approached, ready to capture it, the fox appeared from the underbrush. Remembering the dove's earlier kindness, the fox cleverly diverted the hunter's attention by knocking over a pile of his equipment. This distraction allowed the dove to free itself and fly to safety. The dove, now safe, realized the value of kindness and how unexpected allies could emerge from simple acts of compassion.

The cat, known for its sharp tongue and disdain for humans, wanders near the village and hears a cry for help.

The cat rushes towards the sound and finds a dove struggling to escape from a hunter with a raised bow.

The dove pleads for help, promising to return the favor if saved.

Despite its usual indifference, the cat decides to help and leaps at the hunter, causing the arrow to miss and allowing the dove to fly free.

Days later, the cat is cornered by the same hunter near an oak tree.

The dove swoops down and pecks at the hunter's face, giving the cat time to escape into the forest.

The cat realizes that its act of kindness was not in vain, as kindness is never wasted.

```
[Task]
Generate an outline of the given story.
[User Requirement of the Outline]
User requirement: {$requirement}
[Output Format]
Write a series of sentences in [JSON] format.
E.g., {"1": act_1 content, "2": act_2 content}
Each sentence only contains the content of the
summary.
```

```
----------------- Your Role in the Forest ------------------
You are the  dove in the story.
------------------------------------------------------------
-------------- Previously in Fairytale Forest --------------
(The cat encounters a character seeking help. The character is in danger caused by the hunter.)
------------------------------------------------------------
cat moveTo(forest) thinking I'm tired of the village. Let's see what's happening in the forest.
cat speakTo(ant, "What's wrong, little ant?") thinking Hmm, that ant looks distressed. Maybe it needs help?
ant speakTo(cat, "Help! The hunter is nearby!") thinking If this cat doesn't save me, I might end up in real trouble with the hunter.
cat think("I need to distract the hunter before he gets closer to the ant.")
cat moveTo(village) thinking I'll lead the hunter away from the forest. It's time to create some chaos back in the village.
------------------------------------------------------------
Choose from the following actions: {'0': 'moveTo', '2': 'attack', '3': 'kill', '4': 'think', '5': 'speakTo', '99': 'STOP'}
Please choose via the corresponding number:2
Choose from the following characters: {'0': 'ant', '1': 'cat', '2': 'deer', '3': 'witch', '5': 'hunter'}
Please choose via the corresponding number:0
dove attack(ant)
```

Figure 5: An illustration of the baseline system that uses a prompt-based approach for (1) the outline generation and (2) IN instantiation with (3) the pre-loaded Fairytale Forest story domain. (4) During runtime, LLM acts as a text adventure game engine, dynamically generating game plots based on the outline and responding to player input in the baseline system.

In each condition, participants had five minutes to prepare a narrative instance with a story moral. An example was provided as a starting point. To prime them for a creator's mindset, participants were encouraged to edit the given story by highlighting interesting scenes or adding details, ensuring their edits still aligned with the story moral. Upon editing the story, participants received a walk-through tutorial on the assigned system's features. Then, they had five minutes to practice and familiarize themselves with the tool. After the tutorial, participants used the assigned system to create a narrative outline for 15 minutes. They were encouraged to iteratively refine their outline until it met the task criteria. Upon completion, participants were asked to fill out a narrative editing questionnaire (Figure 6.a) rating their experience with the system.

After filling out the questionnaire, participants played a turn-based interactive narrative game as the Dove in the Fairytale Forest. In each turn, they could choose their actions, with the plot developments guided by the outline they created. They then filled out a game plot questionnaire (Figure 6.b) rating their experience with the generated plots. Finally, we conducted a 10-minute post-study semi-structured interview about their preferences for the two systems. The study took about 70 minutes in total.

5.5 Measurement and Analysis

We collected answers from a semi-structured interview and two questionnaires (narrative editing and game plot questionnaire). We also recorded their interaction activity in the interface. The narrative editing questionnaire focused on assessing the authoring experience. It has 11 questions, including 7 questions rating the controllability, expressivity, and perceived overall experience of the system, as well as 4 questions from the NASA Task Load Index (TLX) [29] which measured participants' perceived workload while using the system. The game plot questionnaire focused on evaluating the player experience and their perceived quality of the generated game plots. This questionnaire included 9 questions rating controllability, expressivity, and the overall quality of the game plots. To examine the potential differences between conditions, we conducted the Wilcoxon signed rank test on the questionnaire results. For the qualitative data collected in the interview, we transcribed audio from the interview and analyzed the results using thematic clustering [8].

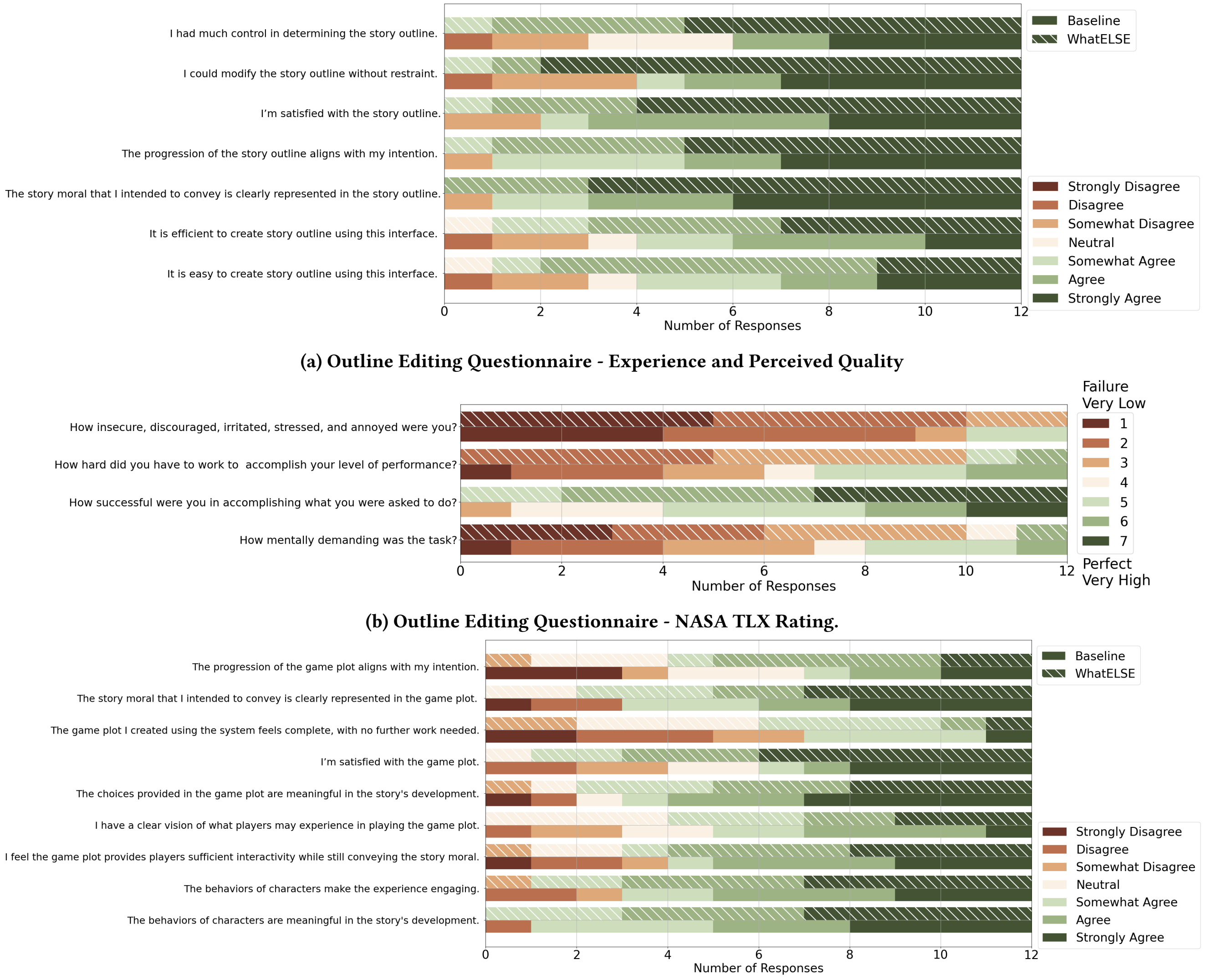
5.6 Results

We present questionnaire results (Figure 6), observation of participants' interactions[3] (Figure 7), and their interview responses to understand how WhatELSE helps people shape the narrative space and further unfold it into executable game events. Numbers in brackets indicate Median (*Med*).

WhatELSE supports the creation of a narrative outline through authorial controls. Participants reported to have more control in generating the outline using WhatELSE (*Med*: 7) than baseline (*Med*: 5, $p = 0.048$). They also reported that they were less restrained in creation using WhatELSE (*Med*: 7) than the baseline (*Med*: 6, $p = 0.049$). In the interview, most participants (9 out of 12) commented that the abstraction tools were helpful. As P4 noted, the ability to select the level of abstraction *"allows users to have more control via clicking buttons."* Similarly, P7 highlighted that the Abstraction Tooltip enhanced the creation process by providing *"an easier interface to change anything you got."* Overall, participants were more satisfied with the outline generated using WhatELSE (*Med*: 7) compared to the baseline (*Med*: 6, $p = 0.047$). They felt more successful in completing the task of creating outlines with WhatELSE (*Med*: 6) compared to the baseline (*Med*: 5, $p = 0.025$).

WhatELSE helps participants perceive the Narrative Space. We found participants alternated between the narrative outline and narrative instance to perceive the narrative space. Nine participants

[3] A recording error resulted in the loss of two participant's interaction recordings.



(a) Outline Editing Questionnaire - Experience and Perceived Quality

(b) Outline Editing Questionnaire - NASA TLX Rating.

(c) Game Plot Questionnaire

Figure 6: Questionnaire results comparing WhatELSE with the baseline.

alternated between Variant and Outline View to edit the narrative space, and two participants alternated between Pivot and Outline View (Figure 7). While the Outline View gives a rough idea of the narrative space, participants found the Variant View offered more detailed information. For example, P2 found the Outline View useful but also commented it *"very summarized though"*. They then switched to the Variant View for concrete examples, which provided them with *"interesting variations"* (P2) that they found *"very helpful for inspiration"*. We observed multiple participants having similar surprising reactions to the variants, indicating that people may *underestimate* the scale of narrative space by working on the outline. Participants also used the Variants View to prevent deviation from their authorial intent. For example, P1 frequently checked the variants for their distance to the given story moral, stating, *"I pretty much completely relied on that, right? OK, if anyone here (character) isn't following my vision for the story, then, you know, I can easily know what the problem is."* This indicates that the visualization helped him quickly identify unexpected elements in the narrative space.

WhatELSE preserves the authorial intent in the gameplay. While playing the game generated from the outline, participants reported that the progression of the game plot aligns better with the story moral using WhatELSE (*Med*: 6.0) compared to the baseline (*Med*: 4.0, $p = 0.020$). For instance, P10 expressed that in the baseline condition, the story moral in the game felt *"some kind of lost"*, indicating a misalignment with their expectations. This contrasts with P10's experience in WhatELSE, where the moral was better preserved within the plot. P5's experience also showcases the effectiveness of the Interactive Narrative Compiler in preserving the authorial intent. P5 created an outline to express *"kindness is not always rewarded"*. In the first round of gameplay, they killed both a weak character (an ant) and a threatening hunter instead of saving

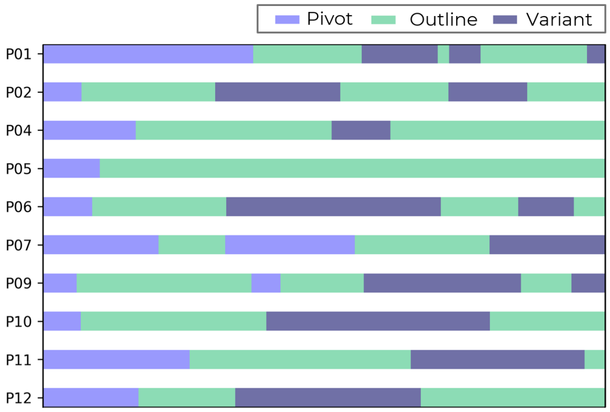


Figure 7: Percentage of participants' interaction time in each view using WhatELSE to create an executable interactive narrative.

the ant. In the second round, the system adapted by having other characters (a deer and a cat) try to stop P5's killing behaviors, but P5 killed them too. In the final round, the system output a monologue that forced P5 to reflect on their actions, conveying the lesson that *"kindness is not always rewarded"*, with kindness coming from NPCs rather than from the players. P5 praised WhatELSE for its adaptability and for effectively conveying the intended moral, describing the system as *"smart"* for how it *"tied my actions and packaged it to be its original directive"*, which was the story moral.

WhatELSE provides engaging game events reacting to player actions. Overall, participants were more satisfied with the game events generated by WhatELSE (*Med*: 6.5) over the baseline (*Med*: 4.5, $p = 0.014$). They felt that WhatELSE responded significantly to their actions, enhancing their sense of agency. For instance, in the baseline, P6 killed a character, but due to a lack of narrative planning, the supposedly dead character reappeared after several rounds. However, in WhatELSE-generated games, their actions had a greater impact; when they tried to "kill" a character, the game responded appropriately. They also found the game plot to be more complete and they will spend less further work in editing narrative space (*Med*: 4.5) compared to the baseline (*Med*: 3.0, $p = 0.027$).

6 Technical Evaluation

The user study shows that WhatELSE effectively helps users create engaging interactive narratives by enhancing both authorial control and player engagement through efficient narrative space editing. To validate the technical pipeline driving the transformation between narrative outline and instances, we conducted technical evaluations focusing on effectiveness in 1) generating the outline from instances and 2) generating instances from an outline.

6.1 From Narrative Instances to Narrative Outline

WhatELSE provides an abstraction ladder with different levels of abstraction to generate an outline from instances using a prompting pipeline. To examine the effectiveness of the pipeline, we employ two lexical-level measures, *concreteness rate*, and *imageability score*. Both measures are adapted from large-scale crowdsourced annotations in previous studies that have been widely used in linguistic evaluations [12, 91]. Both scores are lexicon-based, with each word assigned an averaged score from a batch of crowdsourced annotations; for each outline, we calculate the average score across all words in this outline, excluding stop words. Intuitively, a higher concreteness rate indicates that the wording is more concrete and specific, corresponding to a lower level of abstraction. A lower imageability score suggests that the wording allows greater room for interpretation, corresponding to a higher level of abstraction.

We generated outlines for a sample of 100 stories from the Fairytale dataset [93] at three different levels of abstraction (scene, sequence, and act level). We reported the concreteness rate and imageability score of the generated outlines in Table 1. The results show that our method effectively produces outlines at three distinct levels of abstraction, with significant differences in the lexical measures between each pair of abstraction levels. This demonstrates the system's ability to define narrative spaces with varying degrees of constraint.

Measurement	Abstraction Level		
	Scene Level	Sequence Level	Act Level
Concreteness Rate	3.56 ± 0.05	3.09 ± 0.06	2.95 ± 0.06
Imageability Score	497.33 ± 22.82	453.64 ± 32.42	438.91 ± 34.17

Table 1: Measured abstraction level of the generated outline plot using the proposed prompt pipeline employed in Abstraction Ladder, to generate outline with distinct levels of abstraction from narrative examples. A lower concreteness rate and imageability score indicate the text is more abstract at the lexical level.

6.2 From Narrative Outline to Narrative Instances

We compared the plot quality generated using our approach with the baseline prompting-based approach (Figure 5) from two aspects: *plot diversity* and *player impact*.

Plot Diversity Plot diversity refers to the ability to generate a wide range of different plots within the narrative space. It indicates the level of interactivity and player agency supported by the plot generation method, as it showcases the system's ability to offer varied storylines within the narrative space described by the outline.

To quantitatively assess diversity among a set of N plots generated within the narrative space, we calculate the averaged distance between each plot and the other $N - 1$ plots in the set. We then compute the average distance for each plot relative to the others. For distance calculation, we use the ROUGE score [45], a reference-based evaluation metric that measures text similarity. Adapting from the ROUGE-1 score r_1 targeting word level and macro averaged ROUGE score r_{macro} measures similarity across multiple levels of wording, we compute the word-level distance $d_1 = 1 - r_1$ and the macro distance $d_{macro} = (1 - r_{macro})$ between plots, respectively. As the ROUGE score indicates similarity between text, a higher averaged distance indicates the greater diversity of plots.

Measurement	Human-generated Outline		LLM-generated Outline	
	d_1	d_{macro}	d_1	d_{macro}
Proposed IN Compiler	0.65 ± 0.01	0.78 ± 0.01	0.64 ± 0.01	0.77 ± 0.01
Baseline Prompt-based IN Compiler	0.51 ± 0.02	0.63 ± 0.02	0.61 ± 0.02	0.73 ± 0.02

Table 2: Measured plot distance by averaged ROUGE-1 distance d_1 and the macro-averaged ROUGE distance d_{macro} among the plots generated by baseline and our approach. Results show that WhatELSE IN Compiler leads to a larger averaged distance among the plots, and thus a greater diversity of plots within the narrative space.

Method	Measurement			
	d_1	d_{macro}	World-state Change (Neg)	Character Involvement (Pos)
Proposed IN Compiler	0.59 ± 0.01	0.74 ± 0.001	1.00 ± 0.0	1.70 ± 0.37
Baseline Prompt-based IN Compiler	0.53 ± 0.01	0.67 ± 0.01	0.85 ± 0.08	1.65 ± 0.41

Table 3: Measured impact of player action on game plot progression, by (1) averaged pairwise ROUGE-1 distance d_1 and the macro-averaged ROUGE distance d_{macro} between pairs of the game plots driven by contrasting player actions, and (2) averaged world state change rate driven by the negative player action and the averaged character involvement driven by the positive action. Results show that the contrasting player actions make the proposed approach generate plots with larger pairwise distances. Additionally, in WhatELSE playtime, player actions lead to more stable world-state change and better character involvement than the baseline.

For comparison, we first collected two sets of outlines based on the *"Fairytale Forest"* story domain (Figure 5). The first set, consisting of 12 outlines, was generated by participants in the user study, each tied to one of two specific morals. Additionally, we developed a set of 50 outlines by prompting an LLM, focusing on various morals within the story domain. We simplify each outline by taking only the first act, resulting in 12 human-generated and 100 LLM-generated single-act outlines. These outlines were then used to guide plot generation without involving player actions, using the proposed narrative planning based approach and the baseline prompt-based approach.

We generated a set of 20 plots with each outline and then calculated the averaged distances d_1 and d_{macro} among each set of plots. As shown in Table 2, our approach generated more diverse plots within the same narrative space, indicating more variety of storylines and stronger player agency.

Player Impact We use the term player impact to refer to the extent to which players' actions meaningfully influence the progression of the plot. A higher player impact indicates that the narrative is more responsive to player actions, leading to different outcomes and providing a more personalized experience.

Given a sequence of events (S), the following metrics measure the difference between the subsequent events (S') in response to players taking different actions after the leading sequence S.

- **Subsequent Plot Divergence** We execute a pair of contrasting player actions after S, and compare how the following plot progression diverges semantically based on the player's different actions. The contrasting actions are attacking/killing a character (negative action) and seeking help for a character in danger (positive action). This comparison is performed by calculating two types of ROUGE distances. Specifically, the distance is computed pairwise between the two plots generated after the positive and negative actions. A higher ROUGE distance indicates a greater divergence between the two plots driven by contrasting player actions, thus reflecting their higher impact on the plot progression.
- **Perceived World State Change** The metric assesses the perceived alternation of world states caused by the player's actionn in a specific scenario. We execute a player action of killing a character, and count the frequency of the killed character's reappearance in the subsequent plots.
- **Player Character Involvement** The metric examines whether the player's action increases the player character's involvement in the subsequent events in the plot. We calculate the frequency with which the player's character appears in the plots that follow the positive action of helping a character. This indicates the extent to which the player's actions influence their engagement in the narrative.

We use the *"Fairytale Forest"* story domain (Figure 5) and a fixed story outline containing two acts for evaluation. The player character is set as the dove. To initialize, we generate the plot for the first act in the outline as S and set the world state accordingly. We then execute the designed player actions, and then use our method and the baseline method to generate a batch of 20 plots following each player action to compute the above metrics.

As Table 3 shows, our approach generates significantly more diverse plots following the player's contrasting actions. Moreover, our approach generates plot with better perceived world state change and character involvement following player's actions. Overall, we found that WhatELSE integrates player actions with a higher impact in the narrative generation process.

7 Discussion

The rise of generative models and prompt engineering has significantly impacted various fields and domains. Our user study results suggest that even laypeople can effectively use LLMs to create executable game plots for interactive narratives with appropriate interactive support. Therefore, it is valuable to continue exploring new interaction designs that assist novice users in creating interactive narratives. In this section, we discuss the implications derived from our system design and user studies, as well as the limitations and potential directions for future work.

7.1 Revisiting the Design Goals

We revisit our design goals presented in Section 3 and reflect on the degree to which they have been achieved. We also discuss the opportunities and challenges for designing future AI-bridged IN authoring systems.

DG1: Enable users to perceive the narrative space WhatELSE provides three views for the user to perceive the narrative space: pivot, outline, and variants view. The pivot view displays a user-defined narrative instance as the representative example in the narrative space. In the study, we observed most participants (9/12) did not change the pivot upon generating the outline. Instead, they alternated between variants and outline view to edit the narrative space. Particularly, they found variants view useful in providing *"inspiration"* (P2) or surprising instances, indicating people may underestimate the size of narrative space when authoring IN. Participants also used variant views to prevent instances from deviating from their authorial intent. The variants view shows the distance of each variant to the pivot, allowing users like P1 to quickly identify outliers and remove unexpected instances in their narrative space. Currently, the variants view displays variants based on authorial intent distance and emergent behavior distance from the pivot. It would be interesting to let users customize these dimensions and dynamically visualize the narrative space, similar to Luminate [76].

DG2: Support configurable level of abstraction in editing narrative space Our tools give users the flexibility to adjust the level of detail in their authoring of the narrative space. With the abstraction ladder, they can configure the overall outline, while the abstraction tooltip lets them fine-tune sentences or individual words. This granular control helps users create the narrative space more effectively. The user study reflected this with higher ratings on user control (Figure 6.a) and positive interview responses. Additionally, the technical evaluation results in Table 1 demonstrate the abstraction ladder's effectiveness in managing different levels of abstraction. In the user study, participants with narrative writing experience easily understood the abstraction ladder levels, referring to a level as *"story beats"* (P2). Other participants needed more exploration to find their desired level of abstraction. While the abstraction ladder is designed based on narrative structure, future work should explore making the levels more intuitive for people unfamiliar with narrative writing.

DG3: Generate meaningful game events that react to player actions at play-time We adopted an LLM-based narrative planning method that generates causally sound character actions that act out the event defined in the outline. In the study, participants were more satisfied with the game events generated by WhatELSE compared to the baseline (Figure 6.c). They found that the choices they made in the game had a more realistic impact on the characters. Furthermore, the results of the technical evaluation in Table 3 demonstrate that WhatELSE can generate diverse plots that respond to the player's contrasting actions. Like other LLM-based narrative generation, the quality of generated plots is limited by the challenge of preserving long-term dependency and coherence [53]. Future work should explore approaches such as increasing the LLM's context window [32] and adopting a hierarchical generation approach [53].

7.2 Intuitive and Analytic Thinking in Authoring Narrative Space

WhatELSE supports both intuitive and analytical thinking when authoring IN. Authoring IN with instances aligns with people's natural writing process. It is also straightforward to judge the moral expression in a narrative instance. When authors begin with uploading a draft story, their edits and creations on narrative instances reflect intuitive storytelling, where specific events guide their thinking about what could happen next.

More analytical thinking occurs when authors abstract the outline to shape the narrative space. The act of replacing specific characters or events with abstract descriptions, such as changing *"the hunter"* to *"a human character with power"*, is a deliberate attempt. This process of using language abstraction to incorporate greater diversity requires considerations and planning. By gradually refining the outline through abstraction, authors weigh different narrative potentials and ensure that the result still conveys the desired moral.

WhatELSE enables creators to jointly leverage both modes of thinking in their creation. By starting with concrete narrative instances with clear perception, creators can quickly ground their ideas and establish the foundation for the narrative space to shape. Through the process of abstraction, they deliberately decide operations to shape the narrative space. This transition between intuitive and analytical thinking may serve as a plausible explanation for WhatELSE's advantage for authors to maintain control over the narrative space while allowing for creative flexibility.

7.3 Mixed-initiative Design for LLM Applications

Participants' feedback highlights that, compared to fully free-form conversational assistants, WhatELSE is more helpful for novices in creating their games. As P12 noted, *"I can explore what the different versions (abstraction levels) look like"*, and P6 added, *"(WhatELSE) allows you to explore more."* These comments point to a deeper reason behind the effectiveness of WhatELSE. Though conversational-based chatbots seem to provide infinite options to choose from in creation, they are overwhelming to novices. P1 has been acutely aware of this challenge and left an opinion: *"Maybe someone who is an experienced writer could be like.. I want to structure the plot this way. But as me, a person who isn't a writer, it's hard to really know what to do (with the baseline)"*.

In contrast, the mixed-initiative design of WhatELSE primes users to explore more in sculpting the narrative space in their iterative trials by implicitly guiding them in a structured workflow of outline generation, of which P14 named as *"indicators of what I'm trying to do"*. P2, who has moderate experience in narrative writing, also noticed the design and said, *"It's very close to how I think when I write a story, those levels, those metrics"*.

7.4 Strategic Integration of WHATELSE in Interactive Narrative Creation

A number of IN authoring tools have been developed, including branching-based systems like Twine [22] and event-node graph frameworks [18]. WHATELSE is designed to complement, rather than compete with, these existing tools. We aim to enable WHATELSE to augment existing tools by addressing the challenges widely applied to interactive narrative creation so that a more comprehensive workflow can be supported.

WHATELSE can be used in combination with traditional authoring tools in several ways. For example, the variants generated by WHATELSE can be converted into event diagrams or storyline branches. Therefore, the shaped narrative space via the narrative space editor can be exported to traditional interactive narrative authoring tools for fine-grained manual editing. Similarly, the fully structured representations of the narrative space can be exported to the interactive narrative compiler, as a clearer guide used in the plot execution mode.

Furthermore, our system aims to address the broader challenges of AI-bridged creation of interactive narratives. Within this paradigm, WHATELSE provides creators with the tools to shape, refine, and manage the narrative space. This concept is closely connected to the design space explored in other creative domains; thus, studies that investigate the design exploration and management in design space can be further adapted to this specific application for narrative space editing [20, 76], with their proposed techniques providing another view of the narrative space.

7.5 Branching Outline with Multiple Pivots

It is possible to combine the traditional branching storyline graph representation with our outline representation, to obtain branching outline graphs where each node represents an abstract event. Figure 8 shows such a branching outline, where each path in this graph corresponds to multiple concrete storylines.

One potential extension of this work is to generalize our workflow and system to support creating such branching outline. The more general system should allow the user to provide multiple narrative examples as input to initialize the narrative space, and multiple instances in the narrative space should be able to serve as pivots. For instance, the user can provide three example narratives as pivots to indicate three types of storylines (Figure 8).

Branching outline representation expands the expressivity of both our outline representation and traditional branching graph representation. Our system has shown a linear outline can be unfolded to instantiate the same story structure in multiple concrete forms. A branching outline representation can potentially support organizing multiple story structures together for expressing more complex themes. However, it also raises new challenges for both technical pipeline and interface design, for example, how to computationally construct branching outlines from multiple example narratives, and how to assist the user in better perceiving connection between different narrative instances from the same and different abstract branches, especially when the boundary between different abstract branches can be blurred due to the fuzziness of the trigger conditions expressed with abstract language.

8 Limitation and Future Work

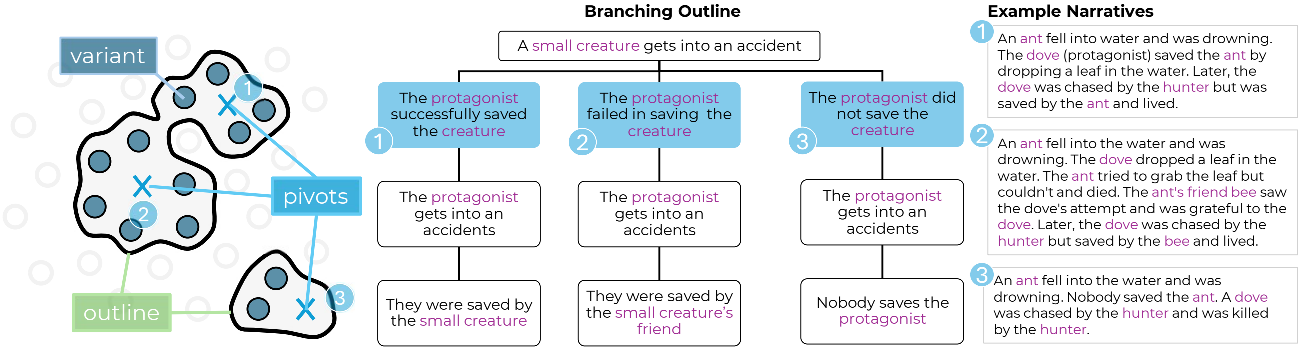
We describe the limitations of our work to clearly define the scope of our findings and inspire future research directions.

Study Design. We evaluated our system through an in-person study with 12 participants who have minimal IN authoring experience and moderate generative AI experience. The study task used a simple story domain with a template story to minimize learning burdens. In real-world scenarios, IN creation typically involves a larger narrative space with more complex plots. Therefore, studies that use more complex creation tasks and have a more diverse participant background, such as including people with limited AI experience or professional IN writers, would give us a deeper understanding of the usability, effectiveness, and generalizability of our work. Additionally, WHATELSE has not been directly compared with traditional IN authoring tools. While a full comparison of WHATELSE and several traditional tools across an entire IN authoring process may not be an overkill, experiments targeting specific perspectives, such as branching capability, can better clarify the unique contribution or limitation of WHATELSE.

Player Experience. Although this work focuses primarily on the author's experience, the player experience is also important. While the results of the technical evaluation show that WHATELSE generates diverse plots that respond to simulated player's actions, future studies the collect subjective player feedback, such as their engagement and enjoyment through crowdsourced assessment of text-adventure games generated using WHATELSE, would help us better understand this system.

Interface Design. The system provides the pivot, variant, and outline view to help perceive the narrative space. Within these three views, more intuitive feedback could be introduced such as providing quantifiable measures of abstraction and concreteness in the outline, similar to the metrics of authorial intent and emergence for variants. While the outline view depicts the boundary of the narrative space, the outline itself is a free-form natural language query. By incorporating classic views of IN, such as a branching diagram, we could explore using a semi-structured outline that combines the flexibility of natural language with the clarity of structured representations. Additionally, introducing an intermediary component between the mutual transformation between narrative instances and the outline, such as an event diagram, would create a "trinity" of instance, semi-structure, and outline, offering more control and granularity in narrative creation.

Export the Game Plot. One of the key advantages of our approach is that game plot is generated through narrative planning, making it fully structured and controllable. It is possible to deploy the generated game plot into a real game engine so that each event in the plot corresponds to an in-game function call. Future work will leverage this advantage to use an existing game engine to serve as an intermediate between WHATELSE and game players, where both the game plots and player actions will operate the game engine, embodying the generated game plot in a real gaming experience.


variant
pivots
outline
1
2
3
Branching Outline
A small creature gets into an accident
The protagonist successfully saved the creature
The protagonist failed in saving the creature
The protagonist did not save the creature
The protagonist gets into an accidents
The protagonist gets into an accidents
The protagonist gets into an accidents
They were saved by the small creature
They were saved by the small creature's friend
Nobody saves the protagonist
Example Narratives

1 An ant fell into water and was drowning. The dove (protagonist) saved the ant by dropping a leaf in the water. Later, the dove was chased by the hunter but was saved by the ant and lived.

2 An ant fell into the water and was drowning. The dove dropped a leaf in the water. The ant tried to grab the leaf but couldn't and died. The ant's friend bee saw the dove's attempt and was grateful to the dove. Later, the dove was chased by the hunter but saved by the bee and lived.

3 An ant fell into the water and was drowning. Nobody saved the ant. A dove was chased by the hunter and was killed by the hunter.

Figure 8: Illustration of a narrative space generated by a branching outline with three pivots to indicate three types of storylines.

9 Conclusion

Generative AI advances interactive narrative creation by enabling just-in-time content generation that adapts to player choices. However, this increased interactivity makes it difficult for authors to control the narrative space. In this paper, we introduced WhatELSE, an interactive narrative authoring system that tackles this challenge via a mutual transformation between narrative instances and narrative space. Through its three views—narrative instance, outline, and variants—WhatELSE empowers authors to perceive and shape narrative boundaries using linguistic abstraction. By leveraging an LLM-based simulation, WhatELSE further unfolds narrative spaces into executable gameplots. Our user study (N=12) and technical evaluations showed that WhatELSE enables the creation of structured yet flexible gameplots, making it an effective tool for interactive narrative creation. We believe our work contributes to advancing creators' balance of their authorial intent and player interactivity in AI-bridged interactive narrative creation.

Acknowledgments

We are grateful to the anonymous reviewers who provided many helpful comments. We also thank our colleagues, David Ledo, Fraser Anderson and Hilmar Koch for feedback and suggestions in preparing the manuscripts and figures. Any opinions, findings, conclusions, or recommendations expressed here are those of the authors alone.

References

[1] Nik Rafidah Nik Muhamad Affendi, Awang Azman Awang Pawi, and Rahimah Hamdan. 2018. The Elements of Thinking in Archipelago's Fables from the Perspective of Children in the 21st Century. *Advanced Science Letters* 24, 7 (2018), 4767–4769.

[2] Inworld AI. 2023. Inworld Origins.

[3] Nader Akoury, Qian Yang, and Mohit Iyyer. 2023. A Framework for Exploring Player Perceptions of LLM-Generated Dialogue in Commercial Video Games. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[4] Prithviraj Ammanabrolu, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara J Martin, and Mark O Riedl. 2020. Story realization: Expanding plot events into sentences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 7375–7382.

[5] Ruth Aylett. 1999. Narrative in virtual environments-towards emergent narrative. In *Proceedings of the AAAI fall symposium on narrative intelligence*. USA, 83–86.

[6] 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*. 1209–1227.

[7] Michael Mose Biskjaer, Peter Dalsgaard, and Kim Halskov. 2014. A constraint-based understanding of design spaces. In *Proceedings of the 2014 conference on Designing interactive systems*. 453–462.

[8] Ann Blandford, Dominic Furniss, and Stephann Makri. 2016. *Qualitative HCI research: Going behind the scenes*. Morgan & Claypool Publishers.

[9] Paul Booth. 2009. Narractivity and the narrative database: Media-based wikis as interactive fan fiction. *Narrative Inquiry* 19, 2 (2009), 372–392.

[10] Aleksandr Boriskin and Daria Galimzianova. 2024. The LSG Challenge Workshop at INLG 2024: Prompting Techniques for Crafting Extended Narratives with LLMs. In *Proceedings of the 17th International Natural Language Generation Conference: Generation Challenges*. 118–122.

[11] Barbaros Bostan. 2009. Player motivations: A psychological perspective. *Computers in Entertainment (CIE)* 7, 2 (2009), 1–26.

[12] Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior research methods* 46 (2014), 904–911.

[13] Q Chen, S Cao, J Wang, et al. 2022. How Does Automation Shape the Process of Narrative Visualization. *A Survey on Tools [J]* (2022).

[14] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848* (2023).

[15] Chun-Wei Chiang, Zhuoran Lu, Zhuoyan Li, and Ming Yin. 2024. Enhancing AI-Assisted Group Decision Making through LLM-Powered Devil's Advocate. In *Proceedings of the 29th International Conference on Intelligent User Interfaces*. 103–119.

[16] Frederik Roland Christiansen, Linus Nørgaard Hollensberg, Niko Bach Jensen, Kristian Julsgaard, Kristian Nyborg Jespersen, and Ivan Nikolov. 2024. Exploring Presence in Interactions with LLM-Driven NPCs: A Comparative Study of Speech Recognition and Dialogue Options. In *Proceedings of the 30th ACM Symposium on Virtual Reality Software and Technology*. 1–11.

[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*. 1–19.

[18] John Joon Young Chung and Max Kreminski. 2024. Patchview: LLM-Powered Worldbuilding with Generative Dust and Magnet Visualization. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–19.

[19] Samuel Rhys Cox and Wei Tsang Ooi. 2023. Conversational Interactions with NPCs in LLM-Driven Gaming: Guidelines from a Content Analysis of Player Feedback. In *International Workshop on Chatbot Research and Design*. Springer, 167–184.

[20] Mohamed Ez-Zaouia, Iza Marfisi-Schottman, Maysa Oueslati, Cendrine Mercier, Aous Karoui, and Sébastien George. 2022. A design space of educational authoring tools for augmented reality. In *International Conference on Games and Learning Alliance*. Springer, 258–268.

[21] Jonas Freiknecht and Wolfgang Effelsberg. 2020. Procedural generation of interactive stories using language models. In *Proceedings of the 15th International Conference on the Foundations of Digital Games*. 1–8.

[22] Jane Friedhoff. 2013. Untangling twine: A platform study. In *Proceedings of DiGRA 2013 Conference*.

[23] Jacob Garbe, Max Kreminski, Ben Samuel, Noah Wardrip-Fruin, and Michael Mateas. 2019. StoryAssembler: an engine for generating dynamic choice-driven narratives. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*. 1–10.

[24] Craig Paul Green, Lars Erik Holmquist, and Steve Gibson. 2020. Towards the Emergent Theatre: A Novel Approach for Creating Live Emergent Narratives Using Finite State Machines. In *International Conference on Interactive Digital Storytelling*. Springer, 92–101.

[25] Daniel Green, Charlie Hargood, and Fred Charles. 2018. Define "authoring tool": a survey of interactive narrative authoring tools. (2018).

[26] Daniel Green, Charlie Hargood, and Fred Charles. 2021. Use of tools: UX principles for interactive narrative authoring tools. *Journal on Computing and Cultural Heritage (JOCCH)* 14, 3 (2021), 1–25.

[27] Melanie C Green and Keenan M Jenkins. 2014. Interactive narratives: Processes and outcomes in user-directed stories. *Journal of Communication* 64, 3 (2014), 479–500.

[28] Juho Hamari and Janne Tuunanen. 2014. Player types: A meta-synthesis. (2014).

[29] SG Hart. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Human mental workload/Elsevier* (1988).

[30] John R Hayes and Linda S Flower. 1983. A Cognitive Model of the Writing Process in Adults. Final Report. (1983).

[31] Meiqing Jin, Manvi Kaul, Shriya Ramakrishanan, Hardik Jain, Samarth Chandrawat, Ishita Agarwal, Tianyi Zhang, Andrew Zhu, and Chris Callison-Burch. [n. d.]. WorldWeaver: Procedural World Generation for Text Adventure Games using Large Language Models. ([n. d.]).

[32] Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and applications of large language models. *arXiv preprint arXiv:2307.10169* (2023).

[33] Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Kaya Stechly, Mudit Verma, Siddhant Bhambri, Lucas Saldyt, and Anil Murthy. 2024. LLMs Can't Plan, But Can Help Planning in LLM-Modulo Frameworks. *arXiv preprint arXiv:2402.01817* (2024).

[34] Eunsuk Kang, Ethan Jackson, and Wolfram Schulte. 2011. An approach for effective design space exploration. In *Foundations of Computer Software. Modeling, Development, and Verification of Adaptive Systems: 16th Monterey Workshop 2010, Redmond, WA, USA, March 31-April 2, 2010, Revised Selected Papers 16*. Springer, 33–54.

[35] 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 CHI Conference on Human Factors in Computing Systems*. 1–16.

[36] Yeo Jin Kim, Alex Goslen, Jonathan Rowe, Bradford Mott, and James Lester. 2023. Language Model-Based Player Goal Recognition in Open World Digital Games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 19. 75–85.

[37] Max Kreminski, John Joon Young Chung, and Melanie Dickinson. 2024. Intent Elicitation in Mixed-Initiative Co-Creativity.. In *IUI Workshops*.

[38] Max Kreminski and Noah Wardrip-Fruin. 2018. Sketching a map of the storylets design space. In *Interactive Storytelling: 11th International Conference on Interactive Digital Storytelling, ICIDS 2018, Dublin, Ireland, December 5–8, 2018, Proceedings 11*. Springer, 160–164.

[39] Michael Lankes and Andreas Stockl. 2023. AI-Powered Game Design: Experts Employing ChatGPT in the Game Design Process. *The Eurasia Proceedings of Science Technology Engineering and Mathematics* 24 (2023), 1–9.

[40] Pier Luca Lanzi and Daniele Loiacono. 2023. Chatgpt and other large language models as evolutionary engines for online interactive collaborative game design. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 1383–1390.

[41] Michael Lebowitz. 1985. Story-telling as planning and learning. *Poetics* 14, 6 (1985), 483–502.

[42] Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. Pre-trained language models for text generation: A survey. *Comput. Surveys* 56, 9 (2024), 1–39.

[43] Xiaoxu Li, Xiao You, Siyuan Chen, Pittawat Taveekitworachai, and Ruck Thawonmas. 2023. Analyzing audience comments: Improving interactive narrative with ChatGPT. In *International Conference on Interactive Digital Storytelling*. Springer, 220–228.

[44] Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. [n. d.]. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[45] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[46] Richard Lin, Rohit Ramesh, Parth Nitin Pandhare, Kai Jun Tay, Prabal Dutta, Bjoern Hartmann, and Ankur Mehta. 2024. Design Space Exploration for Board-level Circuits: Exploring Alternatives in Component-based Design. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–14.

[47] Joachim Majors and Matilda Ståhl. 2021. Some Killer Feedback: A Case Study on Volunteering as Playtesters and Player Typologies. In *European Conference on Games Based Learning*. Academic Conferences International Limited, 516–XXI.

[48] Alessandro Marincioni, Myriana Miltiadous, Katerina Zacharia, Rick Heemskerk, Georgios Doukeris, Mike Preuss, and Giulio Barbero. 2024. The Effect of LLM-Based NPC Emotional States on Player Emotions: An Analysis of Interactive Game Play. In *2024 IEEE Conference on Games (CoG)*. IEEE, 1–6.

[49] Justin Matejka, Michael Glueck, Erin Bradner, Ali Hashemi, Tovi Grossman, and George Fitzmaurice. 2018. Dream lens: Exploration and visualization of large-scale generative design datasets. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–12.

[50] Robert McKee. 1997. Story: Substance, structure, style and the principles of screenwriting. 1997. *Kent, Great Britain: Methuen* (1997).

[51] James R Meehan. 1977. TALE-SPIN, An Interactive Program that Writes Stories.. In *Ijcai*, Vol. 77. 91–98.

[52] Alexia Micallef and Philip M Newton. 2024. The Use of Concrete Examples Enhances the Learning of Abstract Concepts: A Replication Study. *Teaching of Psychology* 51, 1 (2024), 22–29.

[53] 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*. 1–34.

[54] Muhammad U Nasir, Steven James, and Julian Togelius. 2024. Word2World: Generating Stories and Worlds through Large Language Models. *arXiv preprint arXiv:2405.06686* (2024).

[55] OpenAI. 2023. ChatGPT. https://openai.com/chatgpt. Version 4.0.

[56] 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*. 1–22.

[57] Xiangyu Peng, Jessica Quaye, Sudha Rao, Weijia Xu, Portia Botchway, Chris Brockett, Nebojsa Jojic, Gabriel DesGarennes, Ken Lobb, Michael Xu, et al. 2024. Player-driven emergence in llm-driven game narrative. In *2024 IEEE Conference on Games (CoG)*. IEEE, 1–8.

[58] Steven Poulakos, Mubbasir Kapadia, Guido M Maiga, Fabio Zünd, Markus Gross, and Robert W Sumner. 2016. Evaluating accessible graphical interfaces for building story worlds. In *Interactive Storytelling: 9th International Conference on Interactive Digital Storytelling, ICIDS 2016, Proceedings 9*. Springer, 184–196.

[59] Anna Samira Praetorius and Daniel Görlich. 2020. How avatars influence user behavior: A review on the proteus effect in virtual environments and video games. In *Proceedings of the 15th International Conference on the Foundations of Digital Games*. 1–9.

[60] Jay Ratican and James Hutson. 2024. Adaptive Worlds: Generative AI in Game Design and Future of Gaming, and Interactive Media. *ISRG Journal of Arts, Humanities and Social Sciences* 2, 5 (2024).

[61] M.O. Riedl and R.M. Young. 2006. From linear story generation to branching story graphs. *IEEE Computer Graphics and Applications* 26, 3 (2006), 23–31. doi:10.1109/MCG.2006.56

[62] Mark O Riedl. 2009. Incorporating Authorial Intent into Generative Narrative Systems.. In *AAAI spring symposium: intelligent narrative technologies II*. 91–94.

[63] Mark Owen Riedl and Vadim Bulitko. 2013. Interactive narrative: An intelligent systems approach. *Ai Magazine* 34, 1 (2013), 67–67.

[64] Mark O Riedl and Robert Michael Young. 2006. From linear story generation to branching story graphs. *IEEE Computer Graphics and Applications* 26, 3 (2006), 23–31.

[65] Mark O Riedl and R Michael Young. 2006. Story planning as exploratory creativity: Techniques for expanding the narrative search space. *New Generation Computing* 24, 3 (2006), 303–323.

[66] Mark O Riedl and Robert Michael Young. 2010. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research* 39 (2010), 217–268.

[67] Melissa Roemmele and Andrew S Gordon. 2015. Creative help: A story writing assistant. In *Interactive Storytelling: 8th International Conference on Interactive Digital Storytelling, ICIDS 2015, Copenhagen, Denmark, November 30-December 4, 2015, Proceedings 8*. Springer, 81–92.

[68] Adriana Schulz, Jie Xu, Bo Zhu, Changxi Zheng, Eitan Grinspun, and Wojciech Matusik. 2017. Interactive design space exploration and optimization for CAD models. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–14.

[69] Adrian Scigajlo. 2023. *Generation of Potential Narratives in Interactive Fiction*. Ph. D. Dissertation. McMaster University.

[70] Ada-Rhodes Short. 2024. Designing Fictional Worlds for Play Through Large Language Models. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 88377. American Society of Mechanical Engineers, V03BT03A051.

[71] Ulrike Spierling and Nicolas Szilas. 2009. Authoring issues beyond tools. In *Interactive Storytelling: Second Joint International Conference on Interactive Digital Storytelling, ICIDS 2009, Guimarães, Portugal, December 9-11, 2009. Proceedings 2*. Springer, 50–61.

[72] Sarah Stang. 2019. This action will have consequences": Interactivity and player agency. *Game studies* 19, 1 (2019).

[73] Gary M Stump, Mike Yukish, Timothy W Simpson, and E Nathan Harris. 2003. Design space visualization and its application to a design by shopping paradigm. In *International design engineering technical conferences and computers and information in engineering conference*, Vol. 37009. 795–804.

[74] John Louis Styan. 1960. *The elements of drama*. Cambridge University Press.

[75] 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 preprint arXiv:2310.12953* (2023).

[76] Sangho Suh, Meng Chen, Bryan Min, Toby Jia-Jun Li, and Haijun Xia. 2024. Luminate: Structured Generation and Exploration of Design Space with Large Language Models for Human-AI Co-Creation. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–26.

[77] Neil Suttie, Sandy Louchart, Ruth Aylett, and Theodore Lim. 2013. Theoretical considerations towards authoring emergent narrative. In *Interactive Storytelling: 6th International Conference, ICIDS 2013, Istanbul, Turkey, November 6-9, 2013, Proceedings 6*. Springer, 205–216.

[78] Iis Suwartini and Fitri Merawati. 2019. The culture of helping indonesia in the children's story book as media for character education. In *ICLLE 2019: Proceedings of the 2nd International Conference on Language, Literature and Education, ICLLE 2019, 22-23 August, Padang, West Sumatra, Indonesia*. European Alliance for Innovation, 299.

[79] Penny Sweetser. 2024. Large Language Models and Video Games: A Preliminary Scoping Review. *arXiv preprint arXiv:2403.02613* (2024).

[80] MF Syahputra, A Arippa, RF Rahmat, and Ulfi Andayani. 2019. Historical theme game using finite state machine for actor behaviour. In *Journal of Physics: Conference Series*, Vol. 1235. IOP Publishing, 012122.

[81] Marc Tomlinson and Bradley C Love. 2006. Learning abstract relations through analogy to concrete exemplars. In *Proceedings of the 28th annual conference of the Cognitive Science Society*. Lawrence Erlbaum Associates Mahwah, NJ, 2269–2274.

[82] John A Velez and David R Ewoldsen. 2013. Helping behaviors during video game play. *Journal of Media Psychology* (2013).

[83] Richard Walsh. 2011. Emergent narrative in interactive media. *Narrative* 19, 1 (2011), 72–85.

[84] Nick Walton. 2019. AI Dungeon 2. https://aidungeon.io/. Accessed: 2024-09-07.

[85] Yi Wang, Qian Zhou, and David Ledo. 2024. StoryVerse: Towards co-authoring dynamic plot with LLM-based character simulation via narrative planning. In *Proceedings of the 19th International Conference on the Foundations of Digital Games*. 1–4.

[86] Zhilin Wang, Yu Ying Chiu, and Yu Cheung Chiu. 2023. Humanoid Agents: Platform for Simulating Human-like Generative Agents. arXiv:2310.05418 [cs.CL]

[87] Stephen Ware and R Young. 2011. Cpocl: A narrative planner supporting conflict. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 7. 97–102.

[88] Stephen G Ware and Cory Siler. 2021. Sabre: A narrative planner supporting intention and deep theory of mind. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 17. 99–106.

[89] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.

[90] Geraint A Wiggins. 2006. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-based systems* 19, 7 (2006), 449–458.

[91] Michael Wilson. 1988. MRC psycholinguistic database: Machine-usable dictionary, version 2.00. *Behavior research methods, instruments, & computers* 20, 1 (1988), 6–10.

[92] Narnia C Worth and Angela S Book. 2015. Dimensions of video game behavior and their relationships with personality. *Computers in Human Behavior* 50 (2015), 132–140.

[93] Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Jia-Jun Li, Nora Bradford, Branda Sun, et al. 2022. Fantastic Questions and Where to Find Them: FairytaleQA–An Authentic Dataset for Narrative Comprehension. *arXiv preprint arXiv:2203.13947* (2022).

[94] GN Yannakakis, P Spronck, Daniele Loiacono, E André, et al. 2013. Player Modeling. In *Artificial and Computational Intelligence in Games*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 45–59.

[95] Qing Ru Yong and Alex Mitchell. 2023. From Playing the Story to Gaming the System: Repeat Experiences of a Large Language Model-Based Interactive Story. In *International Conference on Interactive Digital Storytelling*. Springer, 395–409.

[96] R Michael Young, Stephen G Ware, Brad A Cassell, and Justus Robertson. 2013. Plans and planning in narrative generation: a review of plan-based approaches to the generation of story, discourse and interactivity in narratives. *Sprache und Datenverarbeitung, Special Issue on Formal and Computational Models of Narrative* 37, 1-2 (2013), 41–64.

[97] Loutfouz Zaman, Wolfgang Stuerzlinger, Christian Neugebauer, Rob Woodbury, Maher Elkhaldi, Naghmi Shireen, and Michael Terry. 2015. Gem-ni: A system for creating and managing alternatives in generative design. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. 1201–1210.