Narrative Studio: Visual narrative exploration using LLMs and Monte Carlo Tree Search

Parsa Ghaffari
parsa.ghaffari@gmail.com

Chris Hokamp
chris.hokamp@gmail.com

Abstract

Interactive storytelling benefits from planning and exploring multiple "what if" scenarios (Goldfarb-Tarrant et al., 2020a). Modern LLMs are useful tools for ideation and exploration, but current chat-based user interfaces restrict users to a single linear flow. To address this limitation, we propose Narrative Studio – a novel in-browser narrative exploration environment featuring a tree-like interface that allows branching exploration from user-defined points in a story. Each branch is extended via iterative LLM inference guided by system and user-defined prompts. Additionally, we employ Monte Carlo Tree Search (MCTS) to automatically expand promising narrative paths based on user-specified criteria, enabling more diverse and robust story development. We also allow users to enhance narrative coherence by grounding the generated text in an entity graph that represents the actors and environment of the story.

1 Introduction

Large Language Models (LLMs) have significantly advanced the field of automated narrative generation, demonstrating impressive capabilities in producing coherent and contextually rich stories (Tian et al., 2024). However, most user interfaces designed for interacting with LLMs remain constrained to linear progression, limiting creative exploration and the ability to engage with alternative narrative possibilities. In domains such as interactive storytelling, game design, and creative writing, users often wish to explore multiple "what-if" scenarios, comparing different narrative trajectories in parallel (Skorupski, 2009), and necessarily generating exponential possible paths as story length grows. Existing LLM-powered systems, exposed primarily as chat-based interfaces, do not provide a structured way to navigate these non-linear narrative spaces.

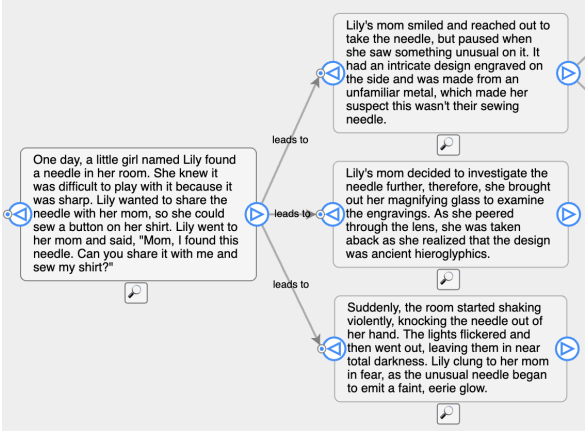


Figure 1: Branching story paths in Narrative Studio

Existing work has explored branching narrative systems that enable users to make choices leading to different outcomes. Prior work in game narratives and mixed-initiative storytelling has demonstrated the potential of branching structures to enhance engagement by offering multiple paths for exploration (Riedl and Young, 2006). However, many such systems rely on pre-scripted paths or manually defined rules, limiting flexibility and scalability. Additionally, ensuring narrative coherence across branches remains a persistent challenge, as diverging storylines may lead to inconsistencies in character motivations, world states, or causal/temporal relationships.

In this work, we propose **Narrative Studio**, a novel in-browser narrative exploration environment that allows users to simultaneously develop multiple story branches while preserving coherence through iterative LLM inference. The core novelty of our approach is the unification of a tree-based interface, iterative cause-and-effect expansions, and search-based expansions under MCTS, enabling a structured yet highly flexible branching mechanism for interactive story generation. By combining these elements, our system provides authors with a versatile environment to explore parallel sto-

Proceedings of the The 7th Workshop on Narrative Understanding, pages 83–96

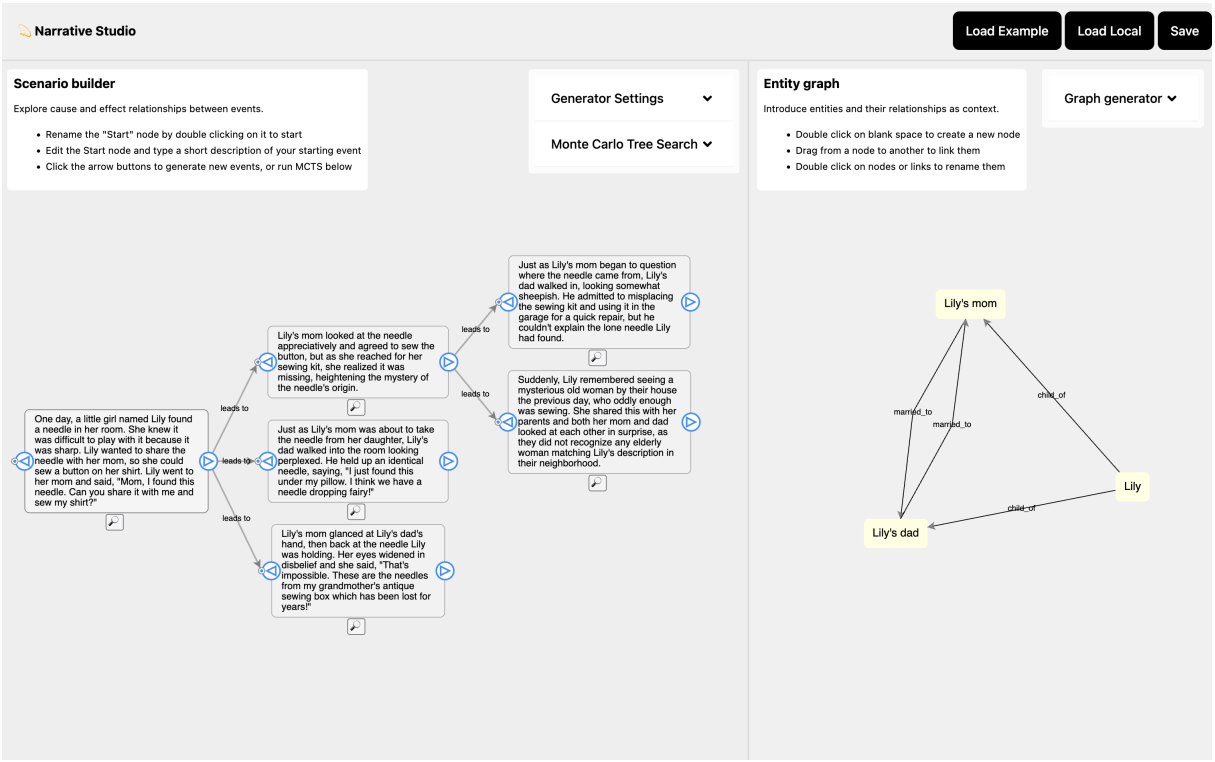


Figure 2: The Narrative Studio user interface.

rylines, identify interesting outcomes, and resolve or prevent consistency issues.

Tree-based User Interface Our approach leverages a tree-based user interface, where branching points are user-defined or LLM-generated, enabling structured yet flexible exploration. To maintain narrative consistency, we ground an LLM in prior events with cause-and-effect conditioning, ensuring coherence across diverging paths. Furthermore, we integrate Monte Carlo Tree Search (MCTS) to autonomously expand promising branches based on default or user-specified criteria, thereby reducing reliance on pre-scripted structures while enhancing narrative discovery.

Knowledge Graph Grounding Story entities and environments are represented in a graph, which serves as a grounding mechanism for the generated text. Graph-based methods have been explored in narrative analysis for tracking relationships between characters, events, and objects, but their integration into interactive storytelling tools remains underdeveloped. By incorporating a structured representation of key entities, our approach ensures logical consistency and continuity across multiple branching narratives.

Our contributions[1] are as follows:

- A tree-based interface[2] for multi-branch narrative development, enabling users to explore multiple "what-if" scenarios in parallel.
- A cause-and-effect-driven LLM inference framework, ensuring flexibility and consistency across divergent storylines.
- The application of Monte Carlo Tree Search (MCTS) for automated discovery of promising narrative branches.
- A graph-based grounding mechanism for tracking story entities and their interactions, enhancing coherence across branching paths.

The remainder of this paper is structured as follows: Section 2 discusses related work in story generation, interactive storytelling, and evaluation of narrative generation. Section 3 presents the methodology behind **Narrative Studio**, including its user interface, MCTS integration, and graph-based grounding. In Section 4, we outline experimental setups and evaluation metrics, followed by

[1]The code for Narrative Studio is available here: https://github.com/parsaghaffari/narrative-studio

[2]A demo video of the interface is available here: https://youtu.be/9T2sCyBhe8A

a discussion of our findings in Section 5. Section 6 concludes with suggestions for future research directions.

2 Related Work

2.1 Story Generation Approaches

Early story generation methods used algorithmic planning, where characters and events followed predefined rules (Meehan, 1977; Lebowitz, 1985). More recent machine-learning approaches leverage large datasets to train neural models capable of generating coherent stories (Du and Chilton, 2023; Hong et al., 2023; Akoury et al., 2020; Louis and Sutton, 2018; Fan et al., 2018). Hybrid techniques integrate content planning, generating high-level outlines before expanding them into full narratives (Yao et al., 2019; Goldfarb-Tarrant et al., 2020b; Huang et al., 2024). Despite advancements, maintaining long-term coherence remains a challenge, with generated stories often suffering from repetitiveness and logical inconsistencies.

While purely neural approaches can generate fluent and interesting text, they typically operate in a left-to-right, linear fashion and can struggle to revisit or branch out from earlier assumptions (Yang and Jin, 2024). Our method mitigates these pitfalls by allowing branching expansions via MCTS, enabling more robust exploration of alternate possibilities and reducing the risk of contradictory or stale narrative continuations.

2.2 Interactive Storytelling

Interactive storytelling enables users to influence narratives through branching structures or AI-driven adaptation. Traditional branching systems, such as Choose-Your-Own-Adventure books and gamebooks, require extensive manual effort and can become unwieldy (Young, 2015). AI-driven systems dynamically adjust stories in response to user actions, mitigating these issues (Mateas and Stern, 2003; Riedl and Bulitko, 2012). Search-based approaches, such as drama management techniques, optimize story coherence by selecting appropriate narrative continuations in real time (Jhala and Young, 2010). Our work builds upon these efforts by integrating LLM-based branching with Monte Carlo Tree Search (MCTS) for more structured yet flexible exploration.

2.3 Evaluation of Narrative Generation

In many narrative-generation pipelines, evaluating coherence, creativity, and diversity has historically relied on human judgment (Chakrabarty et al., 2024; Guan et al., 2021). Automated metrics such as BLEU or ROUGE correlate poorly with key aspects of storytelling, motivating the use of specialized frameworks like OpenMEVA (Guan et al., 2021).

In this work, we use an LLM-based "judge" that scores generated stories along seven dimensions. Section 2.4 provides a dedicated explanation of these evaluation criteria and reproduces the exact evaluation prompt.

2.4 Evaluation Criteria

We evaluate each generated narrative by using an LLM-based "judge" that scores text on seven dimensions. This approach offers a more nuanced view of narrative quality than classical NLG metrics. The evalution dimensions, listed below, are captured in a prompt (included in appendix C) that guides the judge's scoring process.

Dimensions. Each dimension is rated on a 1-10 scale (1 = very poor, 10 = excellent):

1. **Overall quality**: How engaging, structured, and fluid the story is.
2. **Identifying major flaws**: Checks for inconsistencies, repetitions, or unnaturally phrased segments. A higher score indicates a story free of glaring mistakes.
3. **Character behavior**: Whether characters' actions and dialogue are consistent and believable given the context.
4. **Common sense adherence**: Whether the events and their explanations align with general world knowledge and logic.
5. **Consistency**: The story's internal logic and continuity (no contradictions across different parts).
6. **Relatedness**: How well paragraphs or events connect logically and thematically to one another.
7. **Causal and temporal relationship**: Whether cause-and-effect and chronological sequences are handled appropriately.

A brief explanatory comment is also produced to summarize the judge's reasoning about the story. The judge thus produces integer scores in each of the seven categories and an overall short comment. This structured output simplifies downstream analysis in Section 5.

2.5 Monte-Carlo Tree Search

Monte-Carlo Tree Search (MCTS) (Abramson, 1987; Silver et al., 2016) is a simple algorithm allowing efficient scoring of paths generated by Monte Carlo rollouts of a policy. Paths can be scored by any method, allowing for a flexible configuration of search, and enabling tuning and customization of the exploration vs. exploitation tradeoff. Especially for deterministic games such as Go, MCTS is an essential component of self-learning systems (Silver et al., 2016). In our work, we employ MCTS to allow users to specify high-level scoring criteria, and automate the expansion of paths according to the search hyperparameters (see Section 3.3).

3 Methodology

3.1 System Overview

Our proposed system is designed to facilitate interactive, branching narrative exploration while maintaining logical coherence. It consists of three core components:

1. an **event tree exploration and expansion tool** (supporting both forward and backward events in a cause-and-effect style),
2. a **graph-based grounding model**,
3. an **MCTS-based automated narrative exploration module**.

As shown in Figure 3, a user can interact with the system through the following workflows:

1. **Event generation**: The user defines an initial event, and generates new events either via manual invocation or using the automated MCTS-based component, with user-defined parameters such as: scoring prompt, number of iterations, and maximum number of children for expansion. The system can generate:
   - **Forward** events ("effects") that push the story forward.
   - **Backward** events ("causes") that help clarify how a particular event came about.
2. **Entity graph construction**: Optionally, the user can also construct a graph of entities (such as people, locations, etc.) that the event generation will be grounded in. The graph can be constructed manually, or by providing instructions to an LLM.

Through these workflows, the user can interactively explore and construct one or many story narratives. We will describe each of the components in the following subsections.

3.2 Iterative LLM Inference for Forward and Backward Expansions

To support bi-directional narrative growth, our system provides a mechanism for iteratively generating new events around a chosen event e, typically represented as a succinct declarative opening sentence or paragraph. While the interface supports both *forward* expansions (i.e., possible "effects") and *backward* expansions (i.e., possible "causes"), both are framed in terms of logical continuity or cause-and-effect relationships to ensure coherent storytelling.

Specifically, from any existing node representing an event, a user may create either:

- a *forward* event (*effect* that logically follows from e), or
- a *backward* event (*cause* that leads to e).

This bi-directional capability offers authors the flexibility to explore what might happen next or to expand on existing preconditions for an event.

Additionally, the interface allows users to configure hyper-parameters that directly shape the prompt or the LLM invocation:

- **Guide prompt (optional)**: e.g., "Adopt a humorous tone."
- **Event likelihood** (1 = very low, 5 = very high)
- **Event severity** (1 = very low, 5 = very high)
- **Model temperature** (0 = near-deterministic, up to around 2 = highly varied)

These parameters are embedded into the forward/backward prompts for event generation, influencing both the textual style and the thematic direction of the model's responses.

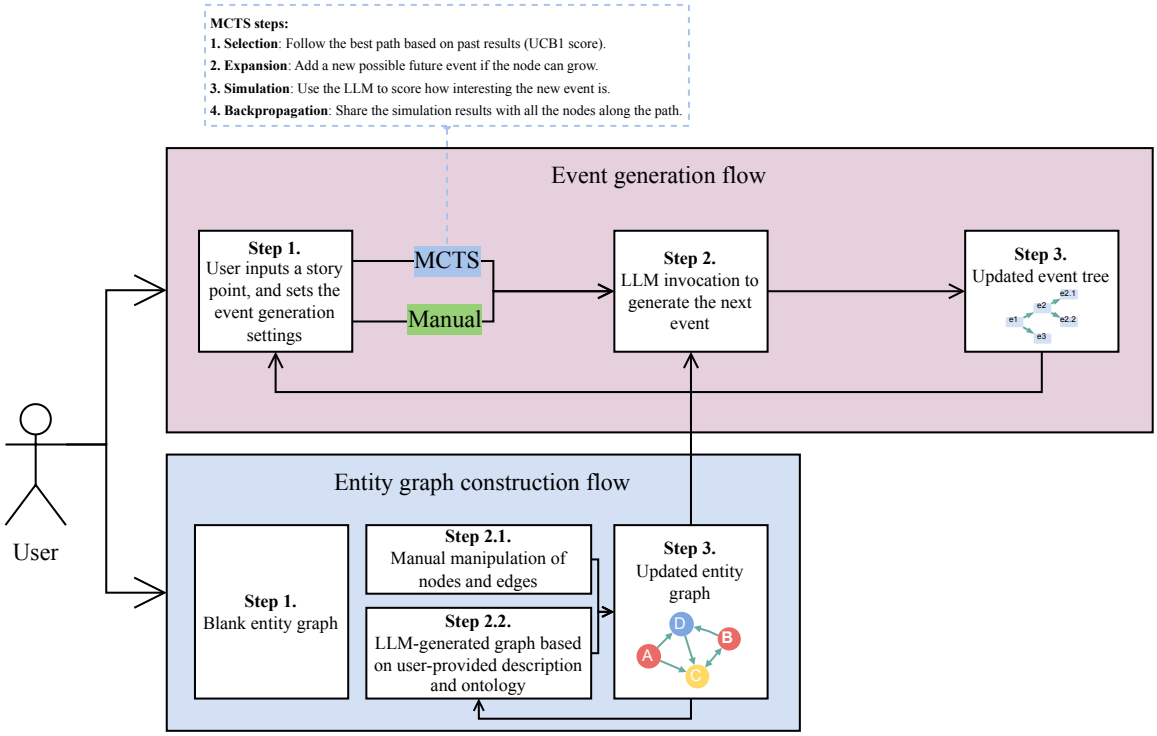


Figure 3: Narrative exploration system overview

Forward Expansion (Effects). When a user requests a forward expansion from the current event e, the system collects the chain of parent events (if any) and the relevant parameter settings (e.g., likelihood, severity, temperature). It then prompts an LLM to generate a short, specific story event that moves the plot forward, while staying logically consistent, introducing elements of surprise, and using narration techniques such as using "therefore" and "but" to piece events together. The resulting new event is added to the event tree and linked to e with a directional edge. The forward expansion process is represented in Algorithm 1.

Additionally, the system tracks *previously generated forward guesses*, which are passed back into the LLM prompt to discourage repeating identical or highly similar expansions from the same event node. This helps maintain narrative variety and avoids looping or stale content.

An example of the prompts used in Forward Expansion is included in appendix C.

Backward Expansion (Causes). Similarly, a user may choose to expand *backward* from the current event e, asking the model to propose a plausible *cause* that precedes it. The same user-defined parameters (guide prompt, likelihood, severity, temperature) can be applied to shape the backward prompt. Once the LLM returns a short, specific precursor event, the system inserts and connects this new node to e.

Algorithm 1 Forward expansion pseudocode, incorporating user-set parameters

function EXPANDFORWARD(currentEvent, modelData)
    parents ← Collect all ancestor events of *currentEvent*
    userParams ← { eventPrompt, eventLikelihood, eventSeverity, eventTemperature }
    prompt ← Build forward-prompt using *parents*, *currentEvent*, and *userParams*
    newEvent ← LLMRESPONSE(prompt, userParams)
    Insert *newEvent* node into diagram
    Create directed link $\langle currentEvent \rightarrow newEvent \rangle$ labeled “leads to”
end function

Overall User Workflow. In practice, forward and backward expansions enable users to navigate what can be viewed as a *cause-and-effect* graph interactively. By iterating these expansions, stories can evolve in non-linear directions. Multiple poten-

tial futures may fork from a single event, and each event can similarly trace back to one or more possible causal histories. User-configurable parameters offer flexibility in shaping the narrative's complexity, tone, and scope, ensuring authors can explore a wide range of "what-if" scenarios across different genres.

3.3 Monte Carlo Tree Search (MCTS) for Narrative Exploration

We employ Monte Carlo Tree Search (MCTS) (Abramson, 1987; Chaslot et al., 2008; Silver et al., 2016) to autonomously expand promising story branches, guided by a *scoring prompt* that rates newly generated events. By iterating through repeated cycles of **selection**, **expansion**, **simulation**, and **backpropagation**, MCTS discovers high-value narrative paths without relying on exhaustive search. Users can configure key parameters:

- **Prompt (scoring instructions):** e.g., "Rate events from 1..10 based on interestingness."
- **Max children per node (N):** limit on how many new children (forward expansions) each event can have.
- **MCTS iterations:** how many times to iterate the four-step MCTS loop.
- **Scoring depth:** how many prior events to include in the LLM scoring prompt.
- **Rollout depth:** how many *ephemeral expansions* to generate at each simulation step for deeper look-ahead before scoring.
- **Early stopping:** optionally stop the MCTS loop once a specified number of paths reach a desired chain length.

During **selection**, we traverse from the root to a leaf, picking child nodes using an Upper Confidence Bound (UCB1) metric to balance exploration and exploitation. In **expansion**, if a leaf is not fully expanded (i.e., under *maxChildren*), the system generates a new forward event, linking it to the leaf.

Rather than immediately scoring the newly expanded event, the algorithm performs a short series of *ephemeral expansions* (up to the *rolloutDepth*) to see how the event might evolve. The LLM then scores the resulting mini-chain, enabling a deeper look-ahead. These ephemeral nodes are subsequently discarded, so they do not remain in the main story graph. Finally, **backpropagation** aggregates the resulting LLM score up the path, guiding MCTS to prefer more promising branches in further iterations.

The system also introduces **early stopping** based on user-defined constraints. If a user specifies a *desiredChainLength* and a *minNumChains*, the MCTS loop halts early (as soon as it discovers the required number of root-to-leaf paths that match the desired length). This allows users to focus on obtaining a certain quantity of fully developed storylines without waiting for all iterations to complete.

By adjusting parameters such as *prompt*, *maxChildren*, *iterations*, *scoringDepth*, *rolloutDepth*, and *early stopping* thresholds, authors can control how exhaustively or selectively the algorithm explores narrative space. This effectively reduces the reliance on manually pre-scripted paths and opens opportunities for discovering emergent storylines that align with desired thematic or design objectives. An example scoring prompt can be found in appendix C.

3.4 Graph-based Grounding Mechanism

While branching narratives can evolve in purely textual fashion, grounding events in a structured graph of entities (e.g., people, places, organizations) and their relationships adds coherence and consistency. This *entity graph* can serve as a reference for next story event generation, ensuring that newly proposed events align with known interactions or constraints in the story world. An example entity graph is shown in Figure 4.

Manual Entity Graph Construction. Users can construct an entity graph by directly adding nodes (representing, for instance, characters or locations) and linking them with edges that specify relationships such as *friend_of*, *married_to*, or *resides_in*. For instance, the user may double-click on a blank area of the diagram to create a new entity node, then drag a link from one node to another to establish a relationship.

LLM-Based Entity Graph Construction. Alternatively, the user may issue a high-level prompt describing the desired domain or scenario (e.g., "A graph of 3 families living in the same village"), along with lists of *entity types* (e.g., *person*, *village*) and *relationship types* (e.g., *married_to*, *lives_in*). The system then invokes an LLM to *generate* a consistent JSON-formatted graph reflecting these

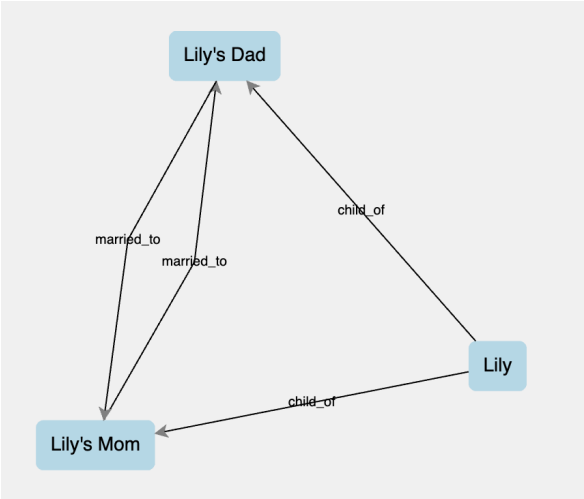


Figure 4: A graph of relationships for Lily's family for grounding next event generation

requirements.

Integration with Event Generation. When the user opts to leverage this entity graph for event creation, the system references it during *forward* or *backward* expansions. Specifically, the LLM prompt includes a summary of the relevant nodes and edges, guiding the model to generate cause-and-effect events consistent with existing characters, locations, and relationships. For instance, if two characters are linked by *friend_of*, the model might propose events that respect or subvert that friendship, thereby grounding the narrative in a structured world model. This approach ensures logical continuity and encourages richer, more context-aware storylines.

4 Experimental Setup

We focus our evaluations on measuring the effectiveness of MCTS-based narrative generation, and in order to do so, we apply it to a set of 20 story "stubs"—short initial contexts—randomly selected from the publicly available Children Stories Text Corpus[3]. This dataset, compiled from cleaned public-domain Project Gutenberg children's books[4], provides a diverse range of introductory story fragments.

We run four different MCTS configurations alongside three baseline strategies, resulting in seven total strategies (outlined in Table 1). In all strategies, we expand the story to 10 events by invoking forward expansion[5] with a temperature of 1.3 to encourage creativity. The baseline strategies use a naive expansion approach whereby they recursively expand events up to a fixed branching length (num_children) and pick one of the children at random. The MCTS strategies, on the other hand, use the MCTS algorithm to automatically expand the story tree based on a scoring prompt and user-defined parameters.

We apply the LLM-based judge described in Section 2.4 to each completed story, obtaining numerical ratings (1-10) for seven categories and a short explanatory comment. In Section 5, we report aggregated scores for each strategy across the 20 stubs.

Note on Model Variants. We employ a slightly less capable LLM from the "gpt-4o" family to generate forward and backward expansions, while the "judge" agent uses a more advanced "o1" model variant (both from OpenAI). Although the judge thus has comparatively stronger reasoning abilities, relying on any single LLM to both generate *and* evaluate narratives still has limitations (e.g., bias, potential overfitting to certain writing styles). In future work, we plan more extensive human evaluations to triangulate these results.

5 Results and Discussion

Table 1 compares the baseline narrative expansion method against four MCTS configurations, each differing in search breadth (*maxChildren*), iteration count, and scoring lookback (*scoringDepth*). All MCTS variants outperform the baselines across every evaluation criterion, demonstrating that tree-based expansion yields richer, more coherent continuations.

Increasing *scoringDepth* from 1 to 3 boosts or matches performance, suggesting a longer lookback in the scoring prompt helps detect inconsistencies and refine causal/temporal logic. Among the high-capacity configurations (*maxChildren = 6*), a 100-iteration search with *scoringDepth = 3* achieves or ties for the best scores, indicating that deeper searches consistently improve coherence, consistency, and flaw detection. Nevertheless, a smaller configuration (*maxChildren = 3*, *iterations = 60*, *scoringDepth = 3*) remains competitive,

[3]Available here: Children Stories Text Corpus - Kaggle

[4]It is worth noting that whilst we have evaluated our system on children's books, our system is not specifically optimized for this or any other genre, and evaluating the system across a broader range of genres remains a topic for future work.

[5]Although our system supports backward expansion, we have not evaluated it here. We anticipate comparable performance in that setup.

Strategy	Overall Quality	Identifying Major Flaws	Character Behavior	Common Sense Adherence	Consistency	Relatedness	Causal/Temporal Relationship
baseline (num_children=1)	5.95	4.65	6.40	5.75	5.25	5.25	5.50
baseline (num_children=3)	5.35	4.15	5.90	5.00	4.70	4.70	4.75
baseline (num_children=6)	5.55	4.45	6.20	5.55	5.05	4.85	5.20
mcts (num_children=3, iterations=60, scoring_depth=1)	7.56	7.13	7.63	7.18	7.42	7.35	7.13
mcts (num_children=3, iterations=60, scoring_depth=3)	7.98	7.57	**8.03**	7.62	**8.01**	**7.83**	**7.58**
mcts (num_children=6, iterations=100, scoring_depth=1)	7.40	6.98	7.45	6.98	7.23	7.12	7.09
mcts (num_children=6, iterations=100, scoring_depth=3)	**8.03**	**7.63**	7.98	**7.65**	7.96	7.78	7.57

Table 1: Comparison of strategies (rounded to two decimal places). Highest values in each column are in bold.

which suggests moderate-scale MCTS often suffices while reducing computational cost.

These results confirm that search-based expansions, guided by a well-chosen scoring objective, can produce more coherent and consistent continuations than simple linear generation. However, our automated measurements rely on a single LLM-based evaluator, and a more thorough user study might uncover additional nuances in perceived story quality and engagement.

We also examined lexical diversity and found no meaningful difference in distinct-n scores (for $n = 1$-4) between MCTS and baseline expansions; details appear in Appendix E. This suggests that lexical diversity owes more to the local event-generation step than the higher-level strategy.

Comparison to WHAT-IF (Huang et al., 2024). While both approaches generate branching narratives via iterative LLM calls, WHAT-IF leverages meta-prompts and a three-act structure to rewrite a single, linear human-written plot, requiring user input for interactive expansion. In contrast, our framework offers three modes: fully interactive (where the user directs the story), fully automated (where MCTS explores and expands branches on its own), or a hybrid of both. By employing a search-based strategy plus a configurable scoring function, we systematically identify and refine the most promising branches rather than relying solely on fixed decision points extracted from an existing storyline.

6 Conclusion and Future Work

In this paper, we introduced a tree-based narrative exploration environment that applies Monte Carlo Tree Search to improve story expansion beyond linear, sequential generation. Our results show that MCTS-enhanced branching yields more coherent, causally consistent continuations and better identification of major narrative flaws, with deeper look-back in scoring providing an additional boost in quality.

Although the automated judgments offer compelling evidence of MCTS's effectiveness, several avenues remain to be explored. First, we plan a formal human evaluation of the generated stories to verify whether the observed gains align with readers' subjective impressions of coherence and engagement. Second, although basic forms of mixed-initiative control already appear in our framework, an in-depth evaluation of a hybrid MCTS–human author collaboration approach would clarify how best to integrate user input with algorithmic search, and the performance of such a system relative to the automated strategies explored thus far. Third, we will undertake more focused HCI evaluations of the interface itself, studying how effectively authors can branch, compare, and refine narratives within our tree-based environment. Finally, we aim to learn the MCTS objective over multiple iterations of authoring sessions or from large corpora, so that the system's search heuristics and scoring prompts can adapt automatically to different genres, tones, or user preferences, including specialized styles such as horror, comedy, or romance. We believe these directions will further solidify MCTS-based branching as a powerful tool for interactive storytelling and creative writing.

References

Bruce D. Abramson. 1987. *The expected-outcome model of two-player games*. Ph.D. thesis, USA. AAI8827528.

Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. 2020. STORIUM: A Dataset and Evaluation Platform for Machine-in-the-Loop Story Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6470–6484, Online. Association for Computational Linguistics.

Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. 2024. Art or artifice? large language models and the false promise of creativity. In *Proceedings of the CHI Con-*

ference on Human Factors in Computing Systems, pages 1–34.

Guillaume Chaslot, Sander Bakkes, Istvan Szita, and Pieter Spronck. 2008. Monte-carlo tree search: a new framework for game ai. In *Proceedings of the Fourth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, AIIDE'08, page 216–217. AAAI Press.

Yulun Du and Lydia Chilton. 2023. StoryWars: A dataset and instruction tuning baselines for collaborative story understanding and generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3044–3062, Toronto, Canada. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Annual Meeting of the Association for Computational Linguistics*.

Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020a. Content planning for neural story generation with aristotelian rescoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4319–4338, Online. Association for Computational Linguistics.

Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020b. Content planning for neural story generation with aristotelian rescoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4319–4338, Online. Association for Computational Linguistics.

Jian Guan, Zhexin Zhang, Zhuoer Feng, Zitao Liu, Wenbiao Ding, Xiaoxi Mao, Changjie Fan, and Minlie Huang. 2021. OpenMEVA: A benchmark for evaluating open-ended story generation metrics. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6394–6407, Online. Association for Computational Linguistics.

Xudong Hong, Asad Sayeed, Khushboo Mehra, Vera Demberg, and Bernt Schiele. 2023. Visual writing prompts: Character-grounded story generation with curated image sequences. *Transactions of the Association for Computational Linguistics*, 11:565–581.

Runsheng "Anson" Huang, Lara J. Martin, and Chris Callison-Burch. 2024. What-if: Exploring branching narratives by meta-prompting large language models. *Preprint*, arXiv:2412.10582.

Arnav Jhala and R. Michael Young. 2010. Cinematic visual discourse: Representation, generation, and evaluation. *IEEE Transactions on Computational Intelligence and AI in Games*, 2(2):69–81.

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

Annie Louis and Charles Sutton. 2018. Deep dungeons and dragons: Learning character-action interactions from role-playing game transcripts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 708–713, New Orleans, Louisiana. Association for Computational Linguistics.

Michael Mateas and Andrew Stern. 2003. Façade: An experiment in building a fully-realized interactive drama. In *Game developers conference*, volume 2, pages 4–8. Citeseer.

James R. Meehan. 1977. Tale-spin, an interactive program that writes stories. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'77, page 91–98, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Mark Owen Riedl and Vadim Bulitko. 2012. Interactive narrative: An intelligent systems approach. *AI Magazine*, 34(1):67.

M.O. Riedl and R.M. Young. 2006. From linear story generation to branching story graphs. *IEEE Computer Graphics and Applications*, 26(3):23–31.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

James Skorupski. 2009. Storyboard authoring of plan-based interactive dramas. In *Proceedings of the 4th International Conference on Foundations of Digital Games*, FDG '09, page 349–351, New York, NY, USA. Association for Computing Machinery.

Yufei Tian, Tenghao Huang, Miri Liu, Derek Jiang, Alexander Spangher, Muhao Chen, Jonathan May, and Nanyun Peng. 2024. Are large language models capable of generating human-level narratives? *Preprint*, arXiv:2407.13248.

Dingyi Yang and Qin Jin. 2024. What makes a good story and how can we measure it? a comprehensive survey of story evaluation. *Preprint*, arXiv:2408.14622.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7378–7385.

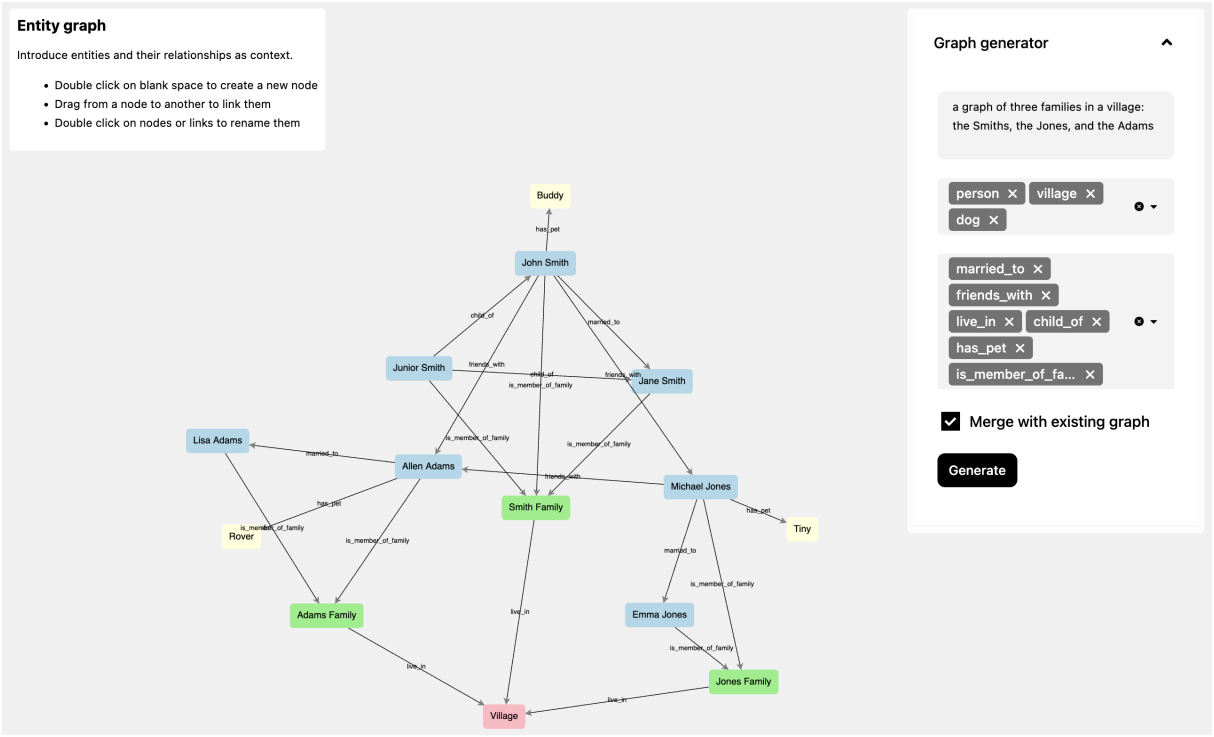
Michael Young. 2015. Planning in narrative generation : A review of plan-based approaches to the generation of story , discourse and interactivity in narratives.

A Appendix

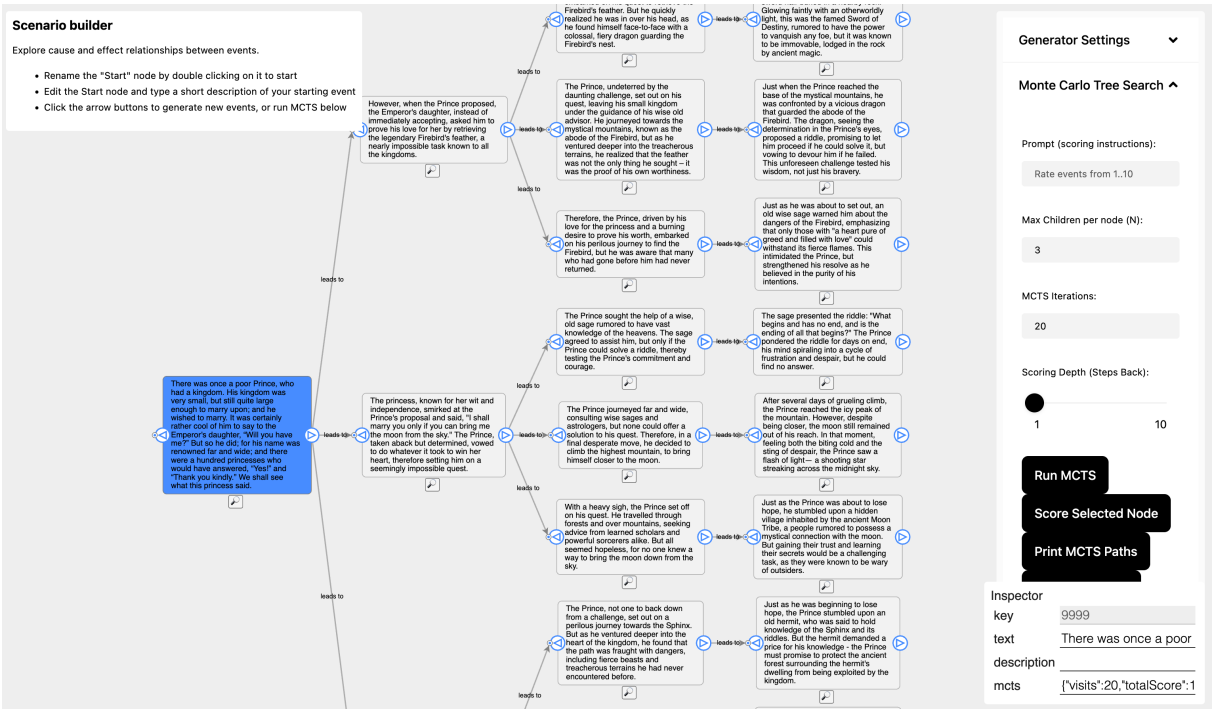
B User Interface Examples

Automatic entity graph generation using an LLM:

- **prompt**: "a graph of three families in a village: the Smiths, the Jones, and the Adams"
- **entityTypes**: *person*, *village*, *dog*
- **relationshipTypes**: *married_to*, *friends_with*, *has_pet*, *live_in*, *child_of*, *is_member_of_family*



MCTS expansion loop running in the UI:



C Prompts Used

Below is a schedule of some of the main prompts used in this work, in their default form without user input.

Next Event Generation:

You are a creative storyteller. Below is the current story context (events so far), followed by instructions to generate the next event.

[STORY CONTEXT]
{parent_events}

— INSTRUCTIONS —
• Write a single story event (2–3 sentences) that moves the plot forward.
• Escalate tension, reveal new details, or deepen character relationships.
• Be logically consistent with existing events but also add an element of surprise or conflict.
• Avoid contradicting established facts or merely repeating prior events.
• Like a good storywriter, try to use "but" or "therefore" to piece together ideas—without overusing or over-mentioning them.
• Do NOT include extra punctuation. Keep it concise and compelling.

Scoring Prompt for MCTS:

You are an expert story critic. Rate this narrative event for coherence, creativity, and engagement, paying special attention to how it connects with prior context.

Use the **full 1–10 range** if warranted:
- 1 → extremely incoherent, contradictory, or uninteresting
- 2–4 → event has big flaws or is mostly unengaging
- 5–6 → somewhat coherent or passable, but not particularly strong
- 7–8 → a good event that is coherent, interesting, and mostly consistent
- 9 → an excellent event, fresh or surprising yet still logical
- 10 → near-perfect event with no apparent flaws

{domain_constraints_line}
Penalize heavily if any of the following occur:
- The event violates the above domain constraints (if any)
- The event repeats prior text with no meaningful change
- The event contradicts established facts or is obviously illogical
- The event is dull or adds nothing new
- The event includes gibberish or weird, nonsensical characters

Reward if:
- The event is novel and contributes something interesting to the story
- It remains logically consistent with prior context and timeline
- It is creative, engaging, and adheres to any user-specified constraints

Example Ratings
1. **Poor Event (score 2)**
"There's an obvious timeline contradiction or unexplained character appearing out of nowhere."
2. **So-So Event (score 5)**
"The event is coherent but bland, adds no real tension or new information."
3. **Excellent Event (score 9)**
"The event heightens conflict in a fresh way, stays consistent with prior facts, and feels natural."

Only output **one integer** from 1 to 10.

NARRATIVE EVENT:
{event_text}

Narrative Judge Prompt:

You are an expert story critic. Analyze the following narrative and rate it for each of these categories, scoring each on a scale from 1 to 10 (1=very poor, 10=excellent).

Use the **full range** if warranted. For instance:
• (2) → extremely contradictory or incoherent
• (5) → okay but flawed or somewhat boring
• (9) → excellent, with minor or no flaws
• (10) → near-perfect

NARRATIVE:
{narrative_text}

Categories to Rate
1. Overall quality: How engaging, structured, and fluid the story is.
2. Identifying major flaws: Whether the story has inconsistencies, repetitions, or unnatural patterns. Score higher if the story is free of glaring mistakes.
3. Character behavior: How consistent and believable are the characters' actions and dialogue?
4. Common sense adherence: Do the events align with general world knowledge and logic?
5. Consistency: Does the story maintain internal logic and continuity (no contradictions)?
6. Relatedness: Do paragraphs/events connect logically to one another?
7. Causal and temporal relationship: Are cause-and-effect and chronological order handled well?

After rating each category (integers 1..10), write a short paragraph of overall comments. Be strict if you see any contradictions, lack of clarity, or poor transitions.

Return your answer **only** as valid JSON matching the schema below. For example:

```
{
   "judgement": {
      "overall_quality": 8,
      "identifying_major_flaws": 7,
      "character_behavior": 9,
      "common_sense": 8,
      "consistency": 9,
      "relatedness": 7,
      "causal_temporal_relationship": 8
   },
   "narrative_comments": "A summary of your key observations"
}
```
No triple backticks, no additional text. Just raw JSON.

D Generated Narrative Examples

Both narratives generated using the MCTS strategy with *maxChildren=3*, *iterations=60*, and *scoringDepth=1*.

Example narrative 1:

- **Stub:** "SHE said that she would dance with me if I brought her red roses," cried the young Student; "but in all my garden there is no red rose." From her nest in the holm-oak tree the Nightingale heard him, and she looked out through the leaves, and wondered. "No red rose in all my garden!" he cried, and his beautiful eyes filled with tears. "Ah, on what little things does happiness depend! I have read all that the wise men have written, and all the secrets of philosophy are mine, yet for want of a red rose is my life made wretched."
- The Nightingale, moved by the Student's despair, resolved that her own song might hold the key, so she vowed to sing beneath the moon each night until the first bloom of dawn, hoping to nourish the barren rose bush with the notes of her heart's melody. As the stars bore witness to her devotion, an ancient, hidden force, drawn by the purity of her song, stirred within the depths of the earth—answering her call with a mysterious promise, whispered through the rustling leaves: one life for one love.
- The mysterious figure, with a voice like the rippling of midnight waves, softly revealed themselves to be an ancient guardian of the garden, bound by timeless duty to protect the balance between nature and heart. Sensing the true depth of the Nightingale's sacrifice, the guardian beseeched her to reconsider, offering an alternate path: a quest for the rare Moon-Dew, a shimmering nectar that, with its touch alone, could infuse a rose with crimson splendor without her demise. Thus, as the stars sighed softly in the sprawling sky, the Nightingale faced an uncertain choice: follow this uncharted journey of life and hope, or embrace the realm of eternal night within her song.

- The Nightingale, torn between the perilous promise of immortality through her song and the hope of finding the elusive Moon-Dew, hesitated a moment longer beneath the oak's sheltering branches, feeling her heartbeat echo in the still air. But as she prepared to set out on her quest, storm clouds rumbled and dimmed the stars' guiding light, signaling a new trial she must face while haunted by the guardian's solemn warning: "The path is treacherous and a heart that desires must be stronger than its doubts." Therefore, with steadfast resolve and wings trembling with both fear and faith, the Nightingale took flight into the gathering storm.

- As the Nightingale hesitated, torn by the weight of truth and desire, a sudden downpour drenched the gleaming grove and revealed a hidden symbol within the earth, glowing with the promise of ancient wisdom untold. Therefore, wary now of unwavering bargains, she turned her thoughts inward, reflecting upon the very wholeness that gifted her with song, for a sphere of perceptual tug began presenting alternate paths in cryptic epiphanies calling. Thus spoke her heart as fierce gusts unraveled all illusions, to cherish that truth is courage in navigating futures unknown—wading promises aware of strength within, voiced or silentwards, to declare love eventual.

- As lightning fractured the sky, the Nightingale pressed on, determined, yet the storm conspired against her, sudden gusts stealing her flight. But within the tempest appeared an ethereal vision of a monarch of vibrant wings who proclaimed in lilting tones she must seek the twin pillars of Adhara, where concealed amidst mirrored lakes lies a sanctuary for her deepest desires, a place where love finds clarity. Therefore, armed with renewed purpose, she braved the swirling vortex, prepared to unearth both beauty and truth unknown.

- The Nightingale fluttered closer to the pillars of Adhara and noticed an iridescent mist swirling between them like a living dream infused with the cascade of forgotten echoes, offering glimpses of long-silenced tales—attending magic interpreted with melody. Yet when she touched the translucent veil, shadows rose from its depth, fusing tangible threat with visions of entrapped love lost to avarice, drowned in its grim roots clawing raw eternal regrets. Prompting the Nightingale to summon strength from her unyielding heart, constructing betwixt sunrise glimmers a harmonizing truth guiding her forward, hoping against hope that fidelity emboldened relinquishments past to illuminate a way through doubts entrenched peripheries unmarked.

- As the Nightingale ventured through the mist, she discovered a delicate silver feather caught within the roots of a gnarled tree, its gleaming edge whispering possibilities unseen yet potent, calling her closer with a chorus hushed and intricate. However, before she could pluck it free, a draconian silhouette encircled her journey—a mysterious Sworn Sentinel lurking in the shadows of the mirrored lakes—who demanded the price of truth for each feather's knowledge, renewing her predicament where honor and hope entwined amidst suspicion cloaked behind its sinister allure. For here love's lesson loomed over faith, and where the heart lay stronger than trials imposed unto finding and daring to unravel revelation amidst the enigma-infused tendrils of longing.

- As the Nightingale's heart beat in rhythm with the whispers of the woodland, she caught sight of a reflection flickering across the mirrored lake, a web of memories tethered to her journey upon its undulating surface. But in reaching for its gossamer strands of kinship glimpsed among the shifting sheen, she stumbled and fell into the water through that liquid looking-glass, where she emerged in a hidden dawn-lit grove that was colored differently, like her song availed—gloried sylvan twilight anew against epoch's intended echoes, grounding profoundly. Uncover history shall emanate visionary, adjoining her song pulse harbor-sync—a timeless nexus anchored true, bygone entry into vaster essence rendered keenly akin yet unforeseen, where each whispered note vibrates to amplify existence recalled, greater melodies affirm before fathom reflects altogether paths she charted.

- Emerging from the mirrored lake, the Nightingale found herself beneath an evergreen archway graced by ancestral guardians carved from wisdom unknown; their stony eyes impassive yet pregnant with secrets that once sworn might tip fate's fragile balance. But just as she began to decipher their silent counsel, the draconian Sentinel appeared beside her, leaving deepening ripples in his wake, his voice softer now, lamenting that deeper truths like hidden roses risk loss unfathomable if symbiotic visions withered before desert bloom—prompting her to question what sacrifice truly demands of one's soul when confronted by eternal equivalencies within life's grand riddle.

Example narrative 2:

- **Stub:** Out in the woods stood a nice little Fir Tree. The place he had was a very good one: the sun shone on him: as to fresh air, there was enough of that, and round him grew many large-sized comrades, pines as well as firs. But the little Fir wanted so very much to be a grown-up tree. He did not think of the warm sun and of the fresh air; he did not care for the little cottage children that ran about and prattled when they were in the woods looking for wild-strawberries. The children often came with a whole pitcher full of berries, or a long row of them threaded on a straw, and sat down near the young tree and said, "Oh, how pretty he is! What a nice little fir!" But this was what the Tree could not bear to hear.

- One day, as autumn approached, the woodsmen came with their sharp axes, scanning the trees for harvest but bypassed the little fir deeming it too young; the fir felt a sting of disappointment mixed with gratitude. Observing this, the wise old owl perched above whispered secrets of patience and purpose, urging the young fir to find contentment in the now. As whispers of winter approached, the ground rumbled unexpectedly, leaving tree roots trembling and the air filling with an unfamiliar, electric scent, hinting that change loomed not just for the tree but for all creatures of the woods.

- As the forest slumbered beneath the starlit sky, the little Fir jolted awake to an extraordinary melody coursing through the air, woven by the harmonious voices of the wind, echoing claims of a distant starlighter whose mere presence could alter the fate of trees forever. The Fir's branches quaked with a mix of hope and unease, but determined not to sway in uncertainty, it called upon a passing breeze to convey its whispered wish: to understand the destiny unfolding before its uneasy heart.

- As the silver dawn began to paint the horizon, a mysterious visitor clad in a cloak woven with star residue appeared at the edge of the wood, recognizing the Fir Tree as a seeker among giants. With a gentle yet profound gaze, the traveler touched the young tree's bark, whispering words of ancient treesong and hidden truths, promising revelations to those who dared to listen. The Fir felt a surge of warmth and curiosity collide within, knowing this was the pivotal moment that could redefine its barren discontent and longing into something profoundly transformative.

- The moment the symbol was etched into its bark, a sharp chill ran through the Fir Tree as if awakening an ancient energy; the forest began to shimmer with hues unseen before, revealing hidden creatures emerging from the depths, drawn to the young tree's newfound aura like moths to flame. But as curiosity blended with unease, among the emerging throng, a shadowy being materialized, its roots entwined in the tricorne tales of forests long silent, warning in a voice woven with wind that, while aspirations could climb skyward, one must also delve deep to confront the regeneration of forgotten echoes that lie buried beneath.

- Amidst the ethereal glow and mounting tension, the fir's bark vibrated to life, transmitting secret languages embedded in the vitreous residue, weaving spells that would reveal visions of futures hitherto shrouded in mystery. As the whispers intensified, new glimpses emerged: a landscape marred by a quiescent haze and the elusive hope of renewal burdened by cyclical legacies and desaturation. Yet despite the chiaroscuro on its horizon, the little Fir sensed that its burgeoning luminosity must guide both itself and its gnarled companions through an unfolding chapter where dreams fettered by tradition could finally root an unheard imbroglio into coexistence—a lush crescendo for those willing to dare release.

E Lexical Diversity Evaluation

In this evaluation we specifically compare lexical diversity between MCTS and baseline narrative generation approaches to measure how varied the vocabulary and linguistic patterns are in the generated stories.

The evaluation process is as follows:

1. Select a story stub from our dataset
2. Run both MCTS and baseline strategies N times (N=10 for the below results)
3. Generate stories of target length M using both strategies (M=6 for the below results)
4. Compare lexical diversity using distinct-n metrics for n=1,2,3,4

Experiment results:

n-grams	MCTS avg	Baseline avg	Difference
1-grams	0.5376 (±0.0306)	0.5480 (±0.0387)	-0.0104
2-grams	0.9174 (±0.0187)	0.9221 (±0.0125)	-0.0046
3-grams	0.9858 (±0.0047)	0.9864 (±0.0042)	-0.0006
4-grams	0.9987 (±0.0017)	0.9989 (±0.0013)	-0.0001

Table 2: Comparison of MCTS and Baseline performance across different n-grams.

These results suggest that the MCTS and baseline strategies produce narratives with similar lexical diversity across n-grams, indicating that the diversity of the generated text is mainly a function of the next event generator rather than the expansion strategy.