Reflections & Resonance: Two-Agent Partnership for Advancing LLM-based Story Annotation

Yuetian Chen, Mei Si
Rensselaer Polytechnic Institute
110 8th Street, Troy, New York 12180
{cheny63, sim}@rpi.edu

Abstract

We introduce a novel multi-agent system for automating story annotation through the generation of tailored prompts for a large language model (LLM). This system utilizes two agents: Agent A is responsible for generating prompts that identify the key information necessary for reconstructing the story, while Agent B reconstructs the story from these annotations and provides feedback to refine the initial prompts. Human evaluations and perplexity scores revealed that optimized prompts significantly enhance the model's narrative reconstruction accuracy and confidence, demonstrating that dynamic interaction between agents substantially boosts the annotation process's precision and efficiency. Utilizing this innovative approach, we created the "StorySense" corpus, containing 615 stories, meticulously annotated to facilitate comprehensive story analysis. The paper also demonstrates the practical application of our annotated dataset by drawing the story arcs of two distinct stories, showcasing the utility of the annotated information in story structure analysis and understanding.

Keywords: Narrative Analysis, Prompt Large Language Model, Story Annotation

1. Introduction

Story annotation, a critical process for dissecting and understanding the multifaceted elements of storytelling such as character arcs, themes, and emotional dynamics, necessitates an intricate understanding of linguistic and narrative structures. Historically, this has been a meticulous and resource-intensive endeavor. Nonetheless, the advent of advanced computational tools like GPT-4 (OpenAI, 2023), marks a significant leap forward. GPT-4's capability to rapidly and comprehensively analyze narrative components can potentially streamline this process, and mitigate the common limitations of manual annotation, including scalability challenges and inherent biases.

The application of LLMs in story annotation introduces its own complexities, necessitating a sophisticated blend of narrative theory and the specific functionalities of the LLMs. While narrative theories illuminate the intricacies of story structure and elements, each LLM, such as GPT-4, possesses distinct capabilities and strengths. To address these challenges, we propose a dual-agent system, inspired by the AutoGen framework (Wu et al., 2023), aimed at enhancing prompt effectiveness through collaboration. Agent A initiates the annotation with prompts based on deep narrative understanding. Agent B then evaluates these prompts through the lens of story reconstruction, offering feedback to refine the prompts in accordance with the accuracy of the reconstructed stories. This process fosters a dynamic, iterative refinement, ensuring the prompts are finely tuned for optimal story annotation, as illustrated in Figure 1.

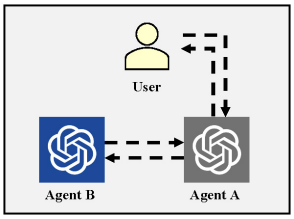

Figure 1: Iterative prompt refinement process

2. Related Work

Existing Story Annotation Datasets A variety of datasets exist for story generation and understanding, each serving different narrative processing needs. CC-Stories (Trinh and Le, 2018) emphasizes common sense reasoning with CommonCrawl documents. TVStoryGen (Chen and Gimpel, 2021) focuses on generating detailed TV show summaries, requiring knowledge of character dynamics. STORYWARS (Du and Chilton, 2023) compiles collaborative stories from diverse authors for comprehensive tasks. The SPGC (Gerlach and Font-Clos, 2020) and Children Stories Text Corpus (Edenbd, 2022), from Project Gutenberg (pro, 2023), offer large book collections, with the latter specialized in children's literature. The Shmoop Corpus (Chaudhury et al., 2019) provides summaries for a chronological narrative view. However, a gap in detailed annotations concerning character states, emotions, decisions, narrative arcs, and writing techniques still exists. This limitation hampers the datasets' utility for complete story recreation and thorough narrative analysis, thereby calling for the creation

LREC-COLING 2024, pages 13813–13818
20-25 May, 2024.

of new datasets with richer annotations.

Automated Prompt Engineering Automated Prompt Engineering (APE) boosts the efficacy of Large Language Models (LLMs) by fine-tuning prompts. Vanilla Prompt Engineering, which is crucial for APE, emphasizes the creation and optimization of prompts (Jiang et al., 2020; Wei et al., 2022; Lampinen et al., 2022), although its effectiveness is tempered by the variability inherent in human-driven processes (Webson and Pavlick, 2021). Prompt Tuning (P-tuning) proposes a more resource-efficient strategy by utilizing a compact, trainable model for generating task-specific virtual tokens (Liu et al., 2022a,b), but it can encounter difficulties with intricate tasks and requires significant data and computational power. Meanwhile, In-context learning (ICL) allows LLMs to adapt to complex tasks with minimal examples, without the need for adjusting the underlying model parameters (Dong et al., 2022).

However, traditional methods for prompt engineering often rely on manual effort, leading to time consumption and inconsistency, or require large datasets to achieve optimal outcomes. In response, our framework draws inspiration from the AutoGen framework (Wu et al., 2023). AutoGen is designed for collaborative goal achievement using multi-agent systems, optimizing LLM workflows by assigning specific roles to agents and coordinating their interactions. By applying this concept, our approach uses a dual-agent system to generate precise prompts for story annotation. This strategy streamlines the prompt engineering process and improves outcomes by harnessing the collaborative potential of multi-agent systems.

3. Methodology & Implementation

In our method, illustrated in Figure 2, we utilize a collaborative interaction between two main agents, both powered by GPT-4, named `Agent A` and `Agent B`. This process involves a dynamic and self-guided iterative feedback loop where `Agent B` provides guidance to `Agent A`, aiming to optimize the narrative annotation task.

3.1. Agent A – the Annotating Agent

During the Problem Initialization Stage, Agent A initiates a basic prompt that captures the specific requirements of the annotation task. This prompt merges guidelines for analysis with necessary formatting details and includes vital elements for deep story comprehension:

1. Story Content: This includes characters' states, intentions, motivations, and emotions, and tracks their evolution, shedding light on the narrative's dynamics.
2. Narrative Presentation: It details the narrative structure, such as whether the storytelling is chronological or non-linear, and looks into literary techniques like foreshadowing or flashbacks to enhance understanding and engagement.

Agent A's approach is designed to lay a comprehensive groundwork for annotating and understanding stories effectively. The full prompt is shown in Figure 2.

3.2. Agent B – the Critic Agent

In Stage II, `Agent B` takes charge of reconstructing a story from the given annotations from `Agent A` using the following prompt:

```
1 Use the following annotated
  predicates and specific information
  to craft a story:
2 [Generated annotations]
```

Agent B then compares the newly assembled story, termed the "Reconstructed Story," with the Ground Truth, utilizing the following guidelines for evaluation:

```
1 Here is the actual ground truth of
  the story and instructions for
  annotation.
2 [Ground Truth Story]
3 [Current Prompt]
4 Analyze the differences and explain
  what other information you need to
  improve the generated story.
```

An example of Agent B's feedback is provided in Table 1. The feedback is then used by `Agent A` to revise the prompt for the next iteration using the following instruction:

```
1 Here is the feedback I have:
2 [Annotation Feedback from Agent B]
3 let's fix it in prompt further
```

The iteration ends after a predetermined N_{iter} number of cycles.

3.3. Construct the StorySense Corpus

Using our methodology, we developed the StorySense corpus with 615 stories, ranging from Aesop's Fables' brief, moral tales to the complex narratives of "20,000 Leagues Under the Sea." These stories were sourced from public domain collections including Project Gutenberg (pro, 2023), the Shmoop Corpus (Chaudhury et al., 2019), and Aesop's Fables (Gibbs et al., 2002), ensuring a

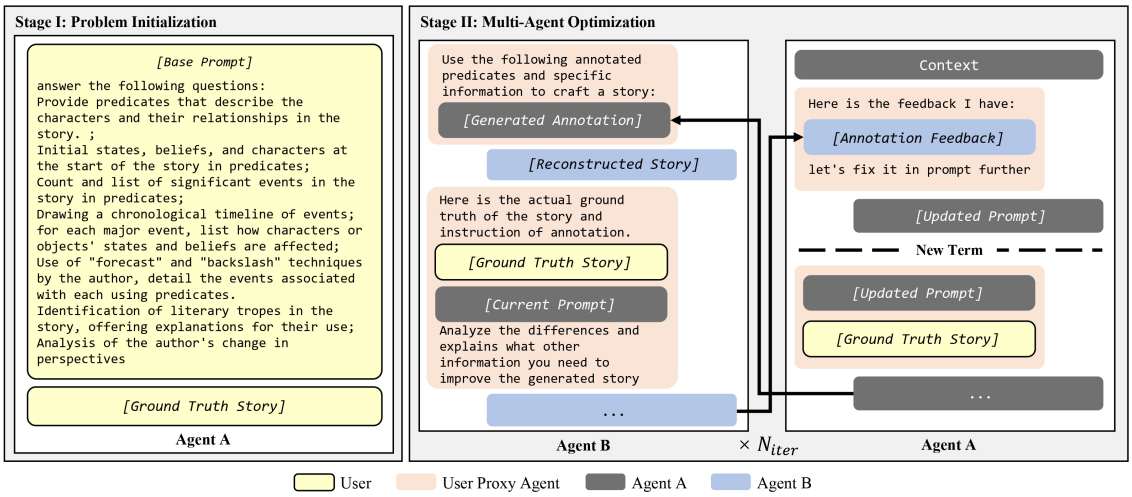

Figure 2: High-level workflow of Reflection & Resonance framework

The generated story does utilize many of the annotations you provided and crafts them into a coherent narrative. However, if you are looking for an exact match or specific elements to be included in a certain way, then the prompt needs to be more explicit.

Here's a more detailed prompt based on your requirements:
STORY CREATION PROMPT
Use the following annotated predicates and specific information to craft a narrative:
Character and Relationship Information:
- Ned Land is a harpooner onboard the ship Abraham Lincoln.
... ...
Starting Scenario and Beliefs:
- The story must begin with the Professor onboard the Abraham Lincoln, which is at sea.
... ...
Major Plot Points:
... ...
Narrative Tone and Perspective:
- The story should maintain a first-person perspective from the Professor's viewpoint.

Table 1: Agent B's feedback example

diverse narrative spectrum. We limited the story length to 200-1800 tokens to maintain uniformity. Figure 3 presents data on story lengths and character counts.

Each story in the corpus was annotated with a prompt optimized from our baseline, refined five times by Agent B ($N_{iter} = 5$). The full optimized prompt is presented in Table 1 of the supplementary material. Table 2 presents the three most frequent tags in three categories, showing their prevalence in the annotations. For instance, the tag `is_protagonist(X)` is found in 95% of the stories, highlighting that nearly all stories feature a protagonist. Conversely, tags under the Significant Events category show a wide variety, reflecting the diversity of narratives in the corpus.

Moreover, the high frequencies of certain tags within each section demonstrate the agent's robustness in identifying and describing key elements of the story, regardless of variations in the specific narratives. This adaptability is crucial for generating comprehensive annotations across a wide range of stories and genres.

4. Evaluation

The goal behind developing this new dataset is to furnish rich data for comprehensive analysis of storytelling methods and to test machine learning models on their ability to rebuild stories solely from annotations. Our evaluation approach is twofold: firstly, we examine the efficacy of our system by assessing how well Agent B provides constructive feedback to improve the quality and applicability of the information extracted. Secondly, we demonstrate the dataset's practical use by visualizing and

Category	Tag	Freq.	Example
Character and Relationship Descriptions in Predicates	`is_protagonist(X)`	95%	`is_protagonist(Alice)`
	`is_antagonist(X)`	80%	`is_antagonist(Queen)`
	`is_mentor(X, Y)`	65%	`is_mentor(Cheshire_Cat, Alice)`
Initial Conditions	fantasy world	85%	The story takes place in a whimsical fantasy world.
	curiosity	70%	The protagonist's curiosity drives the story forward.
	talking animals	60%	The story features animals with human-like qualities.
Significant Events in Predicates	`falls_into(X, Y)`	15%	`falls_into(Frodo, Mount_Doom)`
	`attends(X, Y)`	12%	`attends(Harry, Hogwarts)`
	`confronts(X, Y)`	12%	`confronts(Luke, Darth_Vader)`

Table 2: Example tags generated during annotation

Story ID	Ground Truth	Optimized Prompt		Baseline Prompt	
	Perplexity	Perplexity	Relevance Score	Perplexity	Relevance Score
1	20.03	**13.43**	**4.00**	14.74	2.92
2	15.78	**11.47**	**3.75**	12.36	3.17
3	15.69	**12.26**	**3.58**	13.82	3.25
4	20.03	**9.77**	**3.75**	11.42	3.58
5	**10.75**	13.11	**3.92**	11.05	2.42

Table 3: Relevance evaluation and perplexity scores for 5 stories

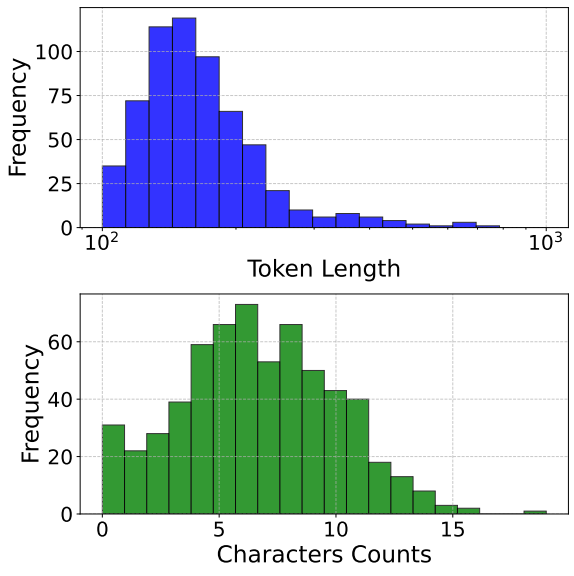

Figure 3: Story lengths and character counts

analyzing two fables, highlighting its capacity to yield deeper insights into narrative techniques and storytelling research.

4.1. Effectiveness of the Framework

To assess our framework's effectiveness in refining prompts and generating effective annotations, we chose five Aesop's Fables from our corpus, leveraging their simplicity and consistent length for a straightforward comparative analysis. The selected stories are "The Kid and the Wolf," "The Lion and the Ass," "The Bees, the Wasps, and the Hornet," "The Bat and the Weasels," and "The Wolf and the Shepherd," labeled as story IDs 1-5 in Table 3. These fables were picked for their varied themes. We compared Agent B's story reconstruction capabilities using both the optimized prompt, refined through five iterations and the original baseline prompt for each fable.

Human Evaluation To evaluate the effectiveness of our framework, we recruited 15 college students as participants to compare the reconstructed stories with the originals. Participants rated the similarity on a scale from 1 (poor match) to 5 (almost perfect reconstruction).

The results, presented in Table 3, reveal that the optimized prompts consistently achieved higher scores in capturing the original stories' essential

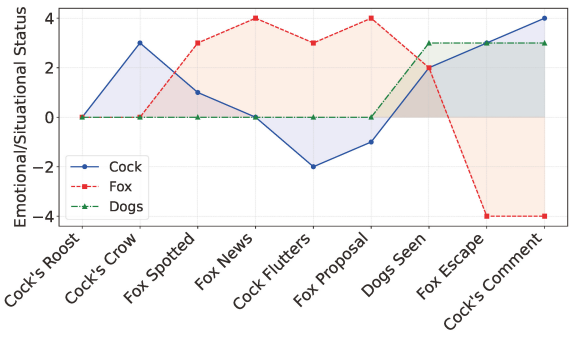

Figure 4: Timeline for "The Cock and the Fox" story

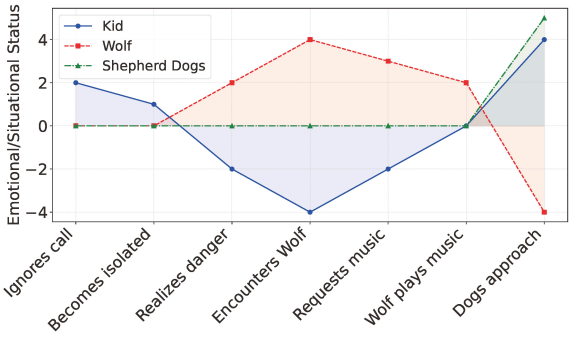

Figure 5: Timeline for "The Wolf and the Kid" story

narrative elements and structure compared to the baseline prompts. The optimized prompts received average scores between 3.75 and 4.00, suggesting a high fidelity in story reconstruction. In contrast, the baseline prompts showed more varied outcomes, indicating possible inconsistencies in their effectiveness.

Automated Evaluation We used perplexity to evaluate the language model's consistency with linguistic patterns in the reconstructed narratives (Jelinek et al., 1977). As shown in Table 3, optimized prompts generally resulted in lower perplexity scores, indicating a closer approximation to the original stories' linguistic style and narrative essence.

The improvement in perplexity with optimized prompts indicates that our method effectively guides the language model to capture key components and stylistic features of the original stories. Higher perplexity scores associated with baseline prompts reflect the model's uncertainty and the challenges of maintaining consistency without precise guidance. The ground truth column in Table 3 provides a reference point for the perplexity of continuing the original story, which typically presents higher values due to the open-ended nature of the task. The comparative analysis of perplexity scores validates our approach, demonstrating that optimized prompts enable the language model to generate text with greater confidence and focus, affirming the efficacy of our optimization method in enhancing narrative reconstruction.

4.2. Example Applications

In this section, we showcase how we use our annotations to visualize the emotional and situational arcs of characters in "The Wolf and the Kid" (Figure 5) and "The Cock and the Fox" (Figure 4), aligning with key story events. We combined character descriptions, event timelines, and character states from our annotations, then used GPT-4 to translate these into a -4 to 4 scale, indicating the intensity of emotional or situational changes. This approach allowed us to plot characters' emotional journeys and situational changes on the story timeline, visually capturing the narrative's dynamics and the characters' experiences.

Positive and negative values on the y-axis signal positive or negative emotions and situations, respectively, with different line styles representing various characters and significant events marked for context.

These visualizations underscore common themes across both stories: the victory of the seemingly weaker party through wit, the intertwined emotional paths of the main characters, and the significant influence of secondary characters. Additionally, these tales convey moral lessons on the value of quick thinking and life's inherent uncertainties.

5. Conclusion and Future Work

This study introduces a multi-agent system for story annotation and the creation of the StorySense corpus, marking significant advancements in using Large Language Models (LLMs) for narrative analysis and story reconstruction. Our work illustrates LLMs' capability to deeply understand and accurately reconstruct narratives, showcasing improved annotation precision and efficiency.

Looking ahead, our research agenda includes expanding human evaluation diversity and sample size, broadening the application of our methods across various stories, and adapting to different language models. Refining the annotation schema and exploring new dimensions of story analysis are also key. Moreover, developing specialized automated evaluation metrics will enhance our ability to assess narrative reconstruction beyond technical accuracy to include creativity and emotional depth.

By advancing these areas, we aim to further our contributions to natural language understanding and generation within storytelling, bridging LLMs' potential with the rich domain of storytelling.

6. Bibliographical References

2023. Project gutenberg. Retrieved: 2023-10-21, from https://www.gutenberg.org.

Atef Chaudhury, Makarand Tapaswi, Seung Wook Kim, and Sanja Fidler. 2019. The shmoop corpus: A dataset of stories with loosely aligned summaries. *arXiv preprint arXiv:1912.13082*.

Mingda Chen and Kevin Gimpel. 2021. Tvstorygen: A dataset for generating stories with character descriptions. *arXiv preprint arXiv:2109.08833*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Yulun Du and Lydia Chilton. 2023. Storywars: A dataset and instruction tuning baselines for collaborative story understanding and generation. *arXiv preprint arXiv:2305.08152*.

Edenbd. 2022. Children stories text corpus. https://www.kaggle.com/datasets/edenbd/children-stories-text-corpus. Accessed: 2023-10-21.

Martin Gerlach and Francesc Font-Clos. 2020. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *Entropy*, 22(1):126.

Laura Gibbs et al. 2002. *Aesop's fables*. OUP Oxford.

Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Andrew K Lampinen, Ishita Dasgupta, Stephanie CY Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L McClelland, Jane X Wang, and Felix Hill. 2022. Can language models learn from explanations in context? *arXiv preprint arXiv:2204.02329*.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022a. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022b. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Albert Webson and Ellie Pavlick. 2021. Do prompt-based models really understand the meaning of their prompts? *arXiv preprint arXiv:2109.01247*.

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*, 35:24824–24837.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. 2023. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155*.