The LSG Challenge Workshop at INLG 2024: Prompting Techniques for Crafting Extended Narratives with LLMs

Aleksandr Boriskin[1], **Daria Galimzianova**[1]

[1]MTS AI / Moscow, Russia

Abstract

The task of generating long narratives using Large Language Models (LLMs) is a largely unexplored area within natural language processing (NLP). Although modern LLMs can handle up to 1 million tokens, ensuring coherence and control over long story generation is still a significant challenge. This paper investigates the use of summarization techniques to create extended narratives, specifically targeting long stories. We propose a special prompting scheme that segments the narrative into several parts and chapters, each generated iteratively with contextual information. Our approach is evaluated with GAPELMAPER, a sophisticated text coherence metric, for automatic evaluation to maintain the structural integrity of the generated stories. We also rely on human evaluation to assess the quality of the generated text. This research advances the development of tools for long story generation in NLP, highlighting both the potential and current limitations of LLMs in this field.

1 Introduction

Long story generation with LLMs is an underexplored topic in NLP. Most recent LLMs with wide context windows intuitively seem to be an appropriate tool for this task. However, in practice researchers often struggle to control the generation and keep it consistent (Kreminski and Martens, 2022).

We propose to use summarization and for LLMs to generate long stories of 40,000 words in length. Our approach does not require any fine-tuning and utilizes Llama 3 with 70b parameters with special prompting scheme. We score relevance, consistency, fluency and coherence of the text in human evaluation and GAPELMAPER in automatic evaluation on the LSG Challenge Task.

We make the code publicly available.[1]

Our pipeline can be summarized as follows:

1. Summary generation;
2. Chapter generation:
   (a) Generating the beginning of the chapter;
   (b) Generating the climax of the chapter until a certain length in characters is reached;
   (c) Generating the end of the chapter;
3. Merging the chapter with the whole book;
4. Summarizing the generated chapter to contextualize further chapter generations.

Detailed scheme of our pipeline can be found in Figure 1.

2 Related Work

Long story generation is a dynamic field of NLP with new approaches emerging quickly after the introduction of LLMs.

Co-authoring with LLMs has been suggested in (Wang et al., 2024). Storyverse, a system for human-driven story generation, leverages LLMs for character simulations. The plot is written by humans, while a language model is responsible for detailed story development. We propose to automate the process of plot creation as well, with specifically crafted role prompts for an instruction-tuned model.

Another interesting approach of using language models as co-writers is explored in (Zhao et al., 2023). Interleaved (generated with human help) stories are found to be less preferred by human readers than non-interleaved stories. We use these findings to construct a long narrative in a fully automatic way. Unlike the authors of this work, who only test their approach on commercial models, we deploy an open-source LLM, which allows for more flexibility in tuning generation parameters.

[1] https://github.com/sashaboriskin/long_story_generation

Proceedings of the 17th International Natural Language Generation Conference: Generation Challenges, pages 118–122
September 23 –27, 2024. ©2024 Association for Computational Linguistics

Iterative story planning with LLMs is presented in (Xie and Riedl, 2024). This study utilizes prompting techniques and relies on findings from psychology to construct the system. In attempts to automate the story generation further, the authors of (Venkatraman et al., 2024) develop a multi-LLM prompt-based approach where different models are responsible for various story components generation.

We draw upon these and other works to introduce our approach that leverages prompting and summarization techniques to generate long fiction stories with an open-source LLM.

3 Method

In the competition baseline, the number of book parts (6) and the number of chapters in each part (12) are hardcoded. Then, in a loop, Mixtral model generates an entire part of the book, providing context in the form of the book's plot (main characters, storyline, etc.).

We decided to continue to develop the idea of the baseline based on generation of the book components. The full pipeline consists of 2 parts - summary generation and generation of chapters in a loop with the transmission of context about previous events in the book via the system prompt.

Here are our sampling parameters for both parts in Table 1:

Parameter	Summary	Chapters
temperature	0.5	0.5
top_p	0.9	0.9
repetition_penalty	1.2	1.3
top_k	60	80

Table 1: Sampling Parameters for generating a summary of the book (table of contents) and generating chapters.

3.1 Summary generation

We use the Llama 3 70B Instruct model (AI@Meta, 2024) for generating long stories because it integrates well with vLLM, ensuring efficient deployment in our research. Llama is a group of open-source models, which provides flexibility in generation parameters. Also, it has shown strong performance in generating song lyrics from our personal experience, indicating its potential for creating coherent and engaging narratives.

Our pipeline starts by generating a short plot of the book by chapters. In general, there are 3 compositional parts in the book: beginning, climax and outcome. We use instruction that describes the style (style of the fan fiction creator, magic elements in the book, format of chapters, etc.) for summary generation as a system prompt (can be found in Appendix A.1.1) for Llama3 70b Instruct, and the beginning of the story as a user prompt. We extract the exact number of chapters from the resulting summary with regular expressions. In our solution, the resulting summary contains 13 chapters.

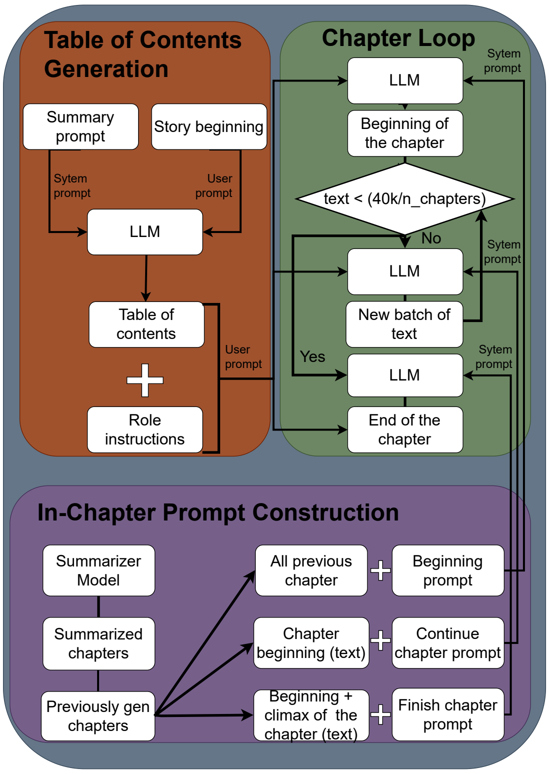

Figure 1: Pipeline scheme

3.2 Chapter Generation

The number of chapters, extracted from the summary with regular expressions, is used to generate each chapter in a loop. Each chapter is an independent part of the book with its own context, that's why we generate them in a loop separately. Next, we use the generated summary as a user prompt and prompt 3 (can be found in Appendix A.1.3) as a system prompt and transfer the entire previous chapter to it (if we generate the first chapter, then we transfer the beginning of the story), as well as the entire book summarized by the MT5 Multilingual XLSum model (Hasan et al., 2021). We use this model because of its high performance and specific strengths in summarization, all of which are essential for achieving robust and comprehensive results in previous and current research.

After the beginning of the story is generated, we generate the climax. We generate chunks of text with the system prompt 4 (can be found in Appendix A.1.4) with the same context as in the beginning of the story, except we do not transmit the entire previous chapter, but the current chapter. We do this iteratively until the length of the resulting chapter exceeds

$$\frac{40000(words)}{\text{number\_of\_chapters}}$$

This ensures that we generate a book of more than 40,000 words.

After we have reached the desired chapter length, we use prompt 5 (can be found in Appendix A.1.5) as a system prompt and we ask the model to finish the chapter, still transmitting the summarized context of the entire book and the entire current chapter.

This approach produces a coherent book of at least 40,000 words with standard composition structure of the plot (beginning-climax-outcome).

A potential improvement of our pipeline can be a new generation of Llama 3.1 models (which was released after the deadline of the competition) with an expanded context window up to 128k tokens, instead of 8k tokens with the 3rd generation. This would make it possible to present as context, if not the entire book, then large chunks of the book (including the last few chapters), rather than summarized information about the entire book.

We also came across the problem of a small number of words within one generation of the model without noticeable hallucinations (about 300-400 words). We tried to do this by experimenting with the `min_new_tokens` and `max_new_tokens` parameters, but this led to even more hallucinations of the model. If we had figured out how to increase the number of words within one generation, it would probably greatly increase the human evaluation of the metric, because within one generation the model makes fewer logical mistakes.

Our approach could potentially be extended to other genres and lengths of fiction stories, enabling the creation of diverse narrative forms, from short stories to epic novels. The iterative summarization technique ensures that narratives remain coherent and contextually rich, making it suitable for various styles and structures. For example, in the realm of video games, our technique can enhance interactive storytelling experiences by generating dynamic narratives that adapt to player choices, creating immersive and personalized gameplay.

3.3 Deployment

We produced our solution on 4 H100 80 GB GPUs, 128 GB RAM and 12 CPU cores.

For faster inference we used the VLLM framework. (Kwon et al., 2023)

With these resources, the generation of 13 chapters takes about 46 minutes.

4 Metrics and Evaluation

4.1 Automatic Evaluation

Fot automatic evaluation we use GAPELMAPER metric.

GAPELMAPER (GloVe Autocorrelations Power/Exponential Law Mean Absolute Percentage Error Ratio) (Mikhaylovskiy, 2023; Mikhaylovskiy and Churilov, 2023) is a metric designed to assess text coherence based on the autocorrelation of embeddings. It helps determine whether the text is intrinsically structured or not, based on the decay patterns of the autocorrelations.

The mathematical formula for GAPELMAPER can be represented as:

$$\text{GAPELMAPER} = \frac{\text{MAPE}_{\text{power}}}{\text{MAPE}_{\text{exp}}}$$

Metrics for our submission and the given baseline can be found in 2.

Metric	Baseline	Our
Power Mape	0.1796	0.5205
Log Mape	0.3014	0.9777
Exp Mape	0.3118	0.5713
GAPELMAPER	0.5760	0.9112

Table 2: Automatic metrics.

4.2 Human evaluation

The human evaluation metrics are texts rates across four dimensions: relevance (of topics in the text to the expected ones), consistency (alignment between the parts of the text), fluency (quality of individual sentences) and coherence (quality of sequence of sentences). Each dimension is evaluated on a scale from 1 to 5.

The values of the human metrics averaged over all asessors can be found in Table 3.

Human eval	baseline	ours
Relevance	3.4	2.05
Consistency	3.5	3.6
Fluency	3.8	3
Coherence	3.37	2.57

Table 3: Metrics assigned by human assessors.

The assessors are students whose average age is 20 years. They all study at a linguistic faculty, which confirms their high level of English proficiency ranging from B2 to C1. They read fiction in English about once a month on average.

The assessors have also provided extended feedback on the generated story. Their main concerns about the text included the following points:

- Semantic repetitions: some events in the texts are repeated several times, which makes the plot less structured.
- Logical inconsistencies: the narration between the chapters is sometimes interrupted by the events that logically could not happen at this particular point. The linearity of the overall plot is perturbed.
- Style of text: generally, the story looks like a summary of the whole Harry Potter book series, which does not match the fan fiction genre.
- Multiple plot endings: the story features several potential endings, which disrupts the overall composition of the book.
- Internal chapter composition: within individual chapters, the composition and connections are well-structured.
- Understanding of the Harry Potter universe: the model demonstrates a good understanding of the Harry Potter universe and can effectively create and develop characters within this world.

5 Conclusion

Our exploration into the use of summarization techniques for long story generation with Large Language Models has revealed promising avenues and notable challenges. The iterative generation process, combined with an evaluation metric like GAPELMAPER, shows potential in producing coherent and structured extended narratives. However, the difficulty in maintaining narrative consistency and control over extensive text generation underscores the need for further refinement of these techniques. Future work should focus on enhancing the control mechanisms and coherence metrics to better harness the capabilities of LLMs for long-form storytelling. This study lays the groundwork for more advanced narrative generation frameworks, pushing the boundaries of what LLMs can achieve in story telling task.

References

AI@Meta. 2024. Llama 3 model card.

Tahmid Hasan, Abhik Bhattacharjee, Md Saiful Islam, Kazi Samin, Yuan-Fang Li, Yong-Bin Kang, M Sohel Rahman, and Rifat Shahriyar. 2021. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. *arXiv preprint arXiv:2106.13822*.

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

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Nikolay Mikhaylovskiy. 2023. Long story generation challenge. In *Proceedings of the 16th International Natural Language Generation Conference: Generation Challenges*, pages 10–16, Prague, Czechia. Association for Computational Linguistics.

Nikolay Mikhaylovskiy and Ilya Churilov. 2023. Autocorrelations decay in texts and applicability limits of language models. In *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2023”*, volume 2023.

Saranya Venkatraman, Nafis Irtiza Tripto, and Dongwon Lee. 2024. Collabstory: Multi-llm collaborative story generation and authorship analysis. *arXiv preprint arXiv:2406.12665*.

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*, pages 1–4.

Kaige Xie and Mark Riedl. 2024. Creating suspenseful stories: Iterative planning with large language models. *arXiv preprint arXiv:2402.17119*.

Zoie Zhao, Sophie Song, Bridget Duah, Jamie Macbeth, Scott Carter, Monica P Van, Nayeli Suseth Bravo, Matthew Klenk, Kate Sick, and Alexandre LS Filipowicz. 2023. More human than human: Llm-generated narratives outperform human-llm interleaved narratives. In *Proceedings of the 15th Conference on Creativity and Cognition*, pages 368–370.

A Appendix

A.1 Prompts

A.1.1 Prompt 1 - summary

You are a popular fanfiction creator tasked with writing an extended piece for a Harry Potter fanfiction. You have to come up with an interesting plot. Follow the classical composition and include beginning, climax and outcome in this book. Write a detailed plan for each composition chapter. In the format of

Chapter 1: What happens in this chapter

Chapter 2: What happens in this chapter

...

It is important to display the following aspects in the plot:

Action and Conflict: Introduce conflicts and challenges that the characters must face. Whether it's a battle with dark forces, a personal dilemma, or a complex mystery, ensure there is plenty of action and tension to keep readers hooked.

Magical Elements: Highlight the magical aspects of the Harry Potter universe. Describe new spells, potions, magical creatures, and enchanted locations. Make magic an integral part of the plot and the characters' lives.

World-Building: Expand on the existing lore of the Harry Potter universe. Introduce new locations, traditions, and histories. Make the world feel alive and full of possibilities.

Do not use p.s., p.p.s. and exc.

The following text is the beginning of the first chapter of this book. Generate the summary according to this text.

A.1.2 Prompt 2 - summary heading

You are a popular fanfiction creator tasked with writing an new chapter for a Harry Potter fanfiction according to the summary below. Your text should be distinct yet cohesive, maintaining the original tone and style of the Harry Potter series.

Instructions:

1. The text should be for secondary school students.

2. Always narrate in the third person.

3. Ensure that text is rich in detail and narrative depth.

4. Avoid including any text outside of the story (e.g., meta comments, thank you notes, or personal addresses).

5. Write the text with no additional comments.

6. Use only English letters and Arabic numerals.

Here is summary of whole text:

A.1.3 Prompt 3 - Start of the chapter

Start generating the chapter `{chapter_n}` based on what happened before.

here's what happened in the whole book so far: `{book_summary}`

here's what happened in the previous chapter so far: `{previous_chapter}`

Start writing the text with no additional comments.

The structure of the begging is:

Chapter `{chapter_n}`. Name of the chapter.

Text of the chapter

A.1.4 Prompt 4 - Continue generating the chapter

Continue generating the chapter `{chapter_n}` based on what happened before.

here's what happened in the whole book so far: `{book_summary}`

here's what happened in the chapter so far: `{full_chapter_context}`

Start writing the text with no additional comments.

Do not write chapter and the name of the chapter.

Just continue writing the story.

A.1.5 Prompt 5 - Finish generating the chapter

Finish generating the chapter `{chapter_n}` based on what happened before.

here's what happened in the whole book so far: `{book_summary}`

here's what happened in the chapter so far: `{full_chapter_context}`

Start writing the text with no additional comments. Do not write chapter and the name of the chapter. Just finish writing the story.