北京邮电大学 本科毕业设计（论文）初期进度报告
Project Early-term Progress Report

学院 **School**	International School	专业 **Programme**	**Electronic Information Engineering**		
姓 **Family name**	Zhang	名 **First Name**	Ziquan		
BUPT 学号 BUPT number	2022213501	**QM 学号 QM number**	221170788	班级 **Class**	2022215117
论文题目 **Project Title**	A Study on Story Model Architectures for RPGs Using Large Language Models				

1. Introduction

Role-playing games (RPGs) represent a distinctive form of interactive narrative in which stories emerge through continuous interaction between players, characters, and the game world. Traditional RPG systems rely heavily on manually authored dialogue trees and scripted narrative branches as shown in Fig.1, which impose rigid constraints on player expression and significantly increase development cost. As a result, these systems often struggle to adapt to unexpected player input or maintain narrative freshness over long play sessions[1].

Recent advances in Large Language Models (LLMs) have introduced new possibilities for dynamic narrative generation and interactive dialogue. Owing to their ability to generate coherent and context-aware text, LLMs have been increasingly explored as virtual narrators, game masters, or non-playable characters (NPCs) in RPG environments. Prior studies demonstrate that LLMs can generate extended role-playing sessions, portray multiple characters, and autonomously drive narrative progression without human intervention[2].

Figure 1. The story dialogue screen in Ys1 and Persona5

Despite these advantages, directly integrating LLMs into RPG systems presents substantial challenges. Studies on LLM-driven NPC dialogue report recurring issues such as inconsistency in character behavior, contradictions with established game lore, and hallucinated content that breaks player immersion.[1] Moreover, the limited context window of LLMs makes it difficult to maintain long-term narrative coherence across extended interactions, a critical requirement for story-driven RPGs. The fully open-ended LLM dialogue increases player agency but also amplifies risks of incoherence and loss of structural guidance[3].

To address these limitations, recent research has emphasized the role of Prompt Engineering as a lightweight yet effective mechanism for controlling LLM behavior. Prompt engineering focuses on carefully structuring model inputs to guide generation style, persona consistency, and task alignment without modifying model parameters[4][5]. In role-playing scenarios, persona-aware prompts have been shown to help LLMs maintain character identity and narrative tone, although prompt-based control alone remains insufficient for preserving long-term contextual consistency.

Retrieval-augmented generation (RAG) addresses the memory and hallucination limitations of LLMs by incorporating retrieved external knowledge into the generation process[6]. Prior studies show that RAG reduces hallucinations and supports knowledge-intensive tasks, while in narrative systems it enables dialogue grounding in persistent world knowledge and character history to maintain long-term coherence.

Recent applied studies in game development further suggest that combining prompt engineering with RAG yields more robust NPC dialogue systems. By embedding retrieved lore and contextual information into structured prompts, LLM-based NPCs can generate responses that are both adaptive to player input and consistent with the game's narrative framework[1]. However, existing work largely focuses on specific implementations or isolated system components, leaving open questions regarding the systematic integration of prompt engineering and RAG within a unified RPG narrative architecture.

Motivated by these gaps, this project explores a system-level approach that integrates prompt engineering and retrieval-augmented generation to support controllable and coherent RPG storytelling. Rather than proposing a new language model, it focuses on architectural design choices for embedding LLMs into interactive narrative systems, with the goal of advancing long-term, player-driven storytelling.

2. Background and Related Work

2.1 LLM-based Story Generation in RPGs

Role-playing games (RPGs) rely on interactive storytelling, where narratives are shaped through player decisions and dialogue with non-playable characters (NPCs). Traditional RPG narrative systems typically employ manually authored scripts and branching dialogue trees, which provide strong narrative control but limit player agency and scale poorly as narrative complexity grows[7].

Large Language Models (LLMs) have enabled a shift toward dynamic, free-form story generation in RPGs. LLMs can respond directly to unconstrained player input, allowing narratives to evolve organically during gameplay rather than following predefined branches[7]. Empirical studies show that LLM-driven RPG systems can sustain multi-turn role-playing interactions and generate linguistically coherent dialogue, suggesting their potential as narrative engines rather than simple dialogue generators[2].

However, prior work consistently identifies limitations in directly applying LLMs to RPG storytelling. One key challenge is maintaining long-term narrative coherence. Due to finite context windows, LLMs struggle to retain persistent world state, character history, and prior events over extended interactions, which can result in contradictions or inconsistent character behavior[8]. In addition, free-form player input may prompt LLMs to generate out-of-scope or narratively inappropriate content, highlighting a tension between player freedom and narrative control[8].

To address these issues, recent research emphasizes the importance of system-level design beyond raw LLM generation. Frameworks such as PANGeA integrate LLMs with external memory and validation mechanisms to support narrative consistency in turn-based RPGs, demonstrating improved coherence compared to standalone LLM approaches[8]. Comparative studies further indicate that structured architectures—rather than unrestricted generation—are critical for balancing expressive freedom with narrative stability in LLM-based RPG systems[3].

2.2 Prompt Engineering for Controllable Narrative Generation

Prompt engineering refers to the systematic design and optimization of input prompts to guide the behavior of Large Language Models (LLMs) without modifying model parameters. Recent surveys emphasize that prompt engineering has evolved from an ad-hoc empirical practice into a

structured research area, supported by formal taxonomies, standardized terminology, and comparative evaluations across tasks[9].

Prompt engineering techniques can be grouped into several core families, including in-context learning, role and style prompting, reasoning-oriented methods, decomposition-based prompting, and self-criticism mechanisms[9]. [20]compares RAG, prompt engineering, and fine-tuning for *full-text* metaphor identification, finding that fine-tuning performs best and that CoT prompting can approach fine-tuning, while RAG and prompting results depend on model type and text length, with many errors reflecting known "grey areas" in metaphor theory rather than random mistakes. These approaches help regulate LLM outputs, making them suitable for narrative tasks that require consistency and control.

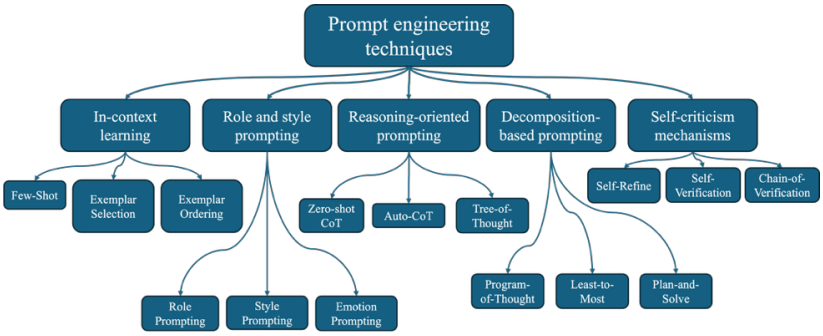

Figure 2. A taxonomy of prompt engineering techniques. Prompt engineering methods can be broadly categorized into in-context learning, role and style prompting, reasoning-oriented prompting, decomposition-based prompting, and self-criticism mechanisms, with representative sub-techniques illustrated under each category.

In the context of interactive storytelling and RPG narrative generation, role prompting and instruction-based prompting play a central role.[16] implements a ChatGPT-powered debate game and demonstrates that prompt-engineering design can reliably control response style/difficulty, debate flow, and automated scoring to improve relevance and usability in an educational setting. By explicitly assigning the model a narrative role—such as a specific character persona or narrator perspective—prompt engineering can improve character consistency and reduce off-role responses in multi-turn interactions[9].

Beyond surface-level control, advanced prompting techniques address reasoning and coherence issues in long-form generation. Reasoning-oriented methods such as Chain-of-Thought prompting encourage models to generate intermediate reasoning steps, which has been shown to improve logical consistency and reduce abrupt narrative shifts in complex generation tasks[10].

Despite these advantages, existing literature consistently highlights the limitations of prompt engineering when used in isolation. Surveys note that prompt-based control is inherently fragile, as LLM outputs remain sensitive to prompt phrasing and context length constraints[9]. A mixed-method study with AI non-expert university students shows that higher-quality prompt engineering (scored via structured prompt components) is strongly associated with higher-quality LLM outputs across two tasks, while the link between AI literacy and prompting performance is mixed and task-dependent[15].

Beyond conventional prompt engineering, recent work shows that prompt-based strategies can partially emulate retrieval-augmented generation in long-context settings by guiding LLMs to attend to relevant input segments[12]. However, such approaches remain limited by context window constraints and lack persistent memory, underscoring the need for retrieval-based augmentation in applications requiring long-term consistency and external knowledge grounding.

2.3 Retrieval-Augmented Generation (RAG)

Comprehensive surveys further show that RAG enables continuous knowledge updating without retraining the underlying model and has been widely applied in areas such as question answering, dialogue systems, and document analysis[14]. [17] shows that a naïve top-k RAG pipeline can generate game-specific, context-grounded reviews without any fine-tuning, while noting performance is constrained by retrieval quality and the model's context-window/hardware limits. As a result, RAG has become a widely adopted approach for knowledge-intensive and long-context tasks.

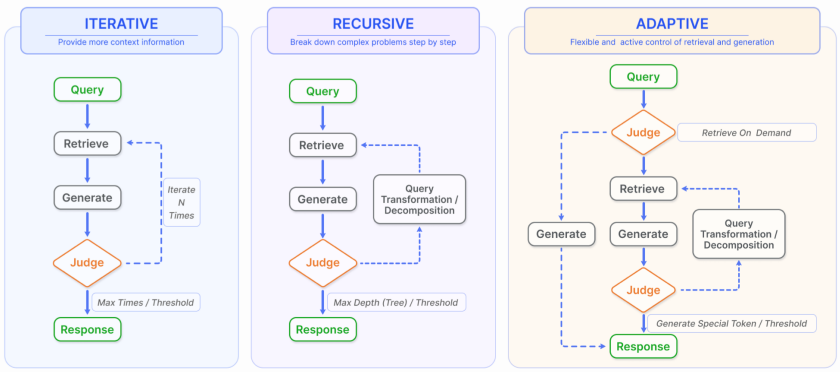

Figure 3. Illustration of three representative Retrieval-Augmented Generation (RAG) paradigms: iterative, recursive, and adaptive RAG. Iterative RAG refines generation through repeated retrieval–generation cycles, recursive RAG decomposes complex queries into sub-queries for hierarchical retrieval, and adaptive RAG dynamically controls retrieval and generation based on intermediate judgments[6].

Recent surveys further highlight a shift from static retrieval pipelines toward reasoning-aware and agentic RAG systems. Reasoning-oriented RAG frameworks embed decision-making into the retrieval process, enabling models to determine when retrieval is necessary, what information to retrieve, and how to integrate retrieved content into ongoing reasoning[11]. [13] analyze the role of retrieval in RAG and show that retrieval quality, document positioning, and noise characteristics have a substantial impact on generation effectiveness, revealing that naive retrieval strategies can even degrade model performance.

Table 1. Comparison of representative approaches for LLM-based narrative generation

	LLM-only Generation	**Prompt-based Control**	**RAG-based Systems**
Core Idea	Direct text generation using pretrained LLMs	Control LLM behavior via structured prompts	Generation with retrieved knowledge
Narrative Flexibility	High	High	Medium–High
Narrative Control	Low	Medium	Medium–High
Long-term Consistency	Low	Low–Medium	High
World / Lore Grounding	Implicit (model memory)	Implicit (prompt context)	Explicit (external knowledge source)
Scalability to Long Sessions	Limited by context window	Limited by prompt length	More scalable via retrieval
Limitations	Narrative drift, hallucination	Prompt sensitivity, lack of memory	Retrieval quality, system complexity

From the perspective of narrative generation, RAG provides an effective mechanism for maintaining long-term consistency by grounding generation in persistent external memory. [18] proposes a RAG-driven "text-to-text" game prototype that turns a single academic book into an interactive text game and shows—via a 50-question benchmark plus human/automatic

evaluation—that RAG improves factual accuracy and domain-grounded detail over a non-RAG baseline, though it still trails human gold answers.

However, existing literature also notes that RAG introduces additional system complexity and latency, and that retrieval quality critically affects generation outcomes. Poorly retrieved or redundant information can degrade coherence rather than improve it, particularly in open-ended generation tasks[11]. These findings suggest that RAG is most effective when combined with careful prompt design and retrieval control mechanisms.

In summary, RAG extends the capabilities of LLMs by providing access to external knowledge and persistent context, making it a key component for long-form and consistency-sensitive applications. Its strengths and limitations motivate hybrid approaches that integrate retrieval augmentation with prompt engineering to support controllable and coherent narrative generation in interactive storytelling systems.

3. Research Gap and Project Positioning

Existing literature demonstrates the growing potential of Large Language Models (LLMs) for interactive storytelling and role-playing game (RPG) dialogue generation[1][3]. At the same time, comparative studies highlight a fundamental design trade-off between expressive freedom and narrative control: while fully open-ended LLM-driven dialogue increases player agency, it also introduces risks of incoherence, narrative drift, and out-of-scope responses[3].

Parallel research highlights prompt engineering as a lightweight control mechanism for shaping LLM behavior. Techniques such as role prompting and structured reasoning prompts can improve character consistency, but prompt engineering alone cannot ensure persistent memory or long-term narrative coherence in extended RPG sessions[9].

Retrieval-augmented generation (RAG) grounds LLM outputs in external knowledge, reducing hallucinations and supporting long-context tasks[6]. [19] evaluates RAG configurations together with ReAct-style prompting, showing that hybrid retrieval plus structured self-evaluation prompts can substantially improve retrieval faithfulness and overall answer quality. Although widely studied in question answering and document analysis, its application to interactive narrative generation remains limited, with existing work paying less attention to narrative-specific challenges such as character continuity and player-driven story evolution.

Existing work leaves a gap between flexible but weakly controlled LLM storytelling and structured but less adaptable systems. This project bridges it with a hybrid of prompt-based control and retrieval-based memory for coherent, player-driven RPG narratives.

4. Project Progress

At the current stage, the project has completed a systematic review of relevant literature on LLM-based narrative generation, prompt engineering, and retrieval-augmented generation. The project is now transitioning from the literature review phase to the implementation phase. Specifically, the focus is on designing the system architecture and exploring concrete coding implementations, including prompt structure design, retrieval pipeline construction, and their integration within an interactive RPG narrative framework.

5. Conclusion

This report reviews LLM-based RPG storytelling, focusing on prompt engineering and RAG. While promising for player-driven narratives, key challenges remain in coherence, long-term context, and control. We therefore move toward implementing a system that integrates prompt-based control with RAG to support coherent, controllable interactive storytelling.

Reference

[1]. A. Mulyana, Y. Wibisono, and A. Anisyah, "Non-Playable Characters Based On Large Language Models For Role Playing Games (RPG)," *Brilliance: Research of Artificial Intelligence*, vol. 5, no. 2, pp. 785–792, Aug. 2025, doi: https://doi.org/10.47709/brilliance.v5i2.6779.

[2]. A. Maisto, "Collaborative Storytelling and LLM: A Linguistic Analysis of Automatically-Generated Role-Playing Game Sessions," arXiv.org, 2025. https://arxiv.org/abs/2503.20623.

[3]. E. Rimer, Rasmus Ploug, K. Skov, and M. Scirea, "Talking to NPCs: Three LLM-Driven Approaches to Dynamic RPG Dialogue," University of Southern Denmark Research Portal (University of Southern Denmark), pp. 1–2, Aug. 2025, doi: https://doi.org/10.1109/cog64752.2025.11114413.

[4]. P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal, and A. Chadha, "A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications," arXiv (Cornell University), Feb. 2024, doi: https://doi.org/10.48550/arxiv.2402.07927.

[5]. B. Han, T. Susnjak, and A. Mathrani, "Automating Systematic Literature Reviews with Retrieval-Augmented Generation: A Comprehensive Overview," Applied Sciences, vol. 14, no. 19, pp. 9103–9103, Oct. 2024, doi: https://doi.org/10.3390/app14199103.

[6]. Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," arXiv.org, Dec. 18, 2023. https://arxiv.org/abs/2312.10997#:~:text=Retrieval%2DAugmented%20Generation%20(RAG) .

[7]. A. van der Torre, "Large language models and narrative storytelling in video games," *Preprint*, Apr. 2025, doi: 10.13140/RG.2.2.17147.02086.

[8]. S. Buongiorno, L. J. Klinkert, T. Chawla, Z. Zhuang, and C. Clark, "PANGeA: Procedural Artificial Narrative using Generative AI for Turn-Based Video Games," arXiv.org, Apr. 30, 2024. https://arxiv.org/abs/2404.19721.

[9]. S. Schulhoff *et al.*, "The Prompt Report: A Systematic Survey of Prompting Techniques," *arXiv.org*, Jun. 16, 2024. https://arxiv.org/abs/2406.06608.

[10]. B. Chen, Z. Zhang, N. Langrené, and S. Zhu, "Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review," *arXiv.org*, Oct. 27, 2023. https://arxiv.org/abs/2310.14735.

[11]. Liang, G. Su, H. Lin, Y. Wu, R. Zhao, and Z. Li, "Reasoning RAG via System 1 or System 2: A Survey on Reasoning Agentic Retrieval-Augmented Generation for Industry Challenges," *arXiv preprint* arXiv:2506.10408, Jun. 2025.

[12]. J. Park, K. Atarashi, K. Takeuchi, and H. Kashima, "Emulating Retrieval Augmented Generation via Prompt Engineering for Enhanced Long Context Comprehension in LLMs," *arXiv.org*, 2025. https://arxiv.org/abs/2502.12462.

[13]. F. Cuconasu et al., "The Power of Noise: Redefining Retrieval for RAG Sys-tems," 2024, doi: https://doi.org/10.1145/3626772.3657834.

[14]. W. Fan et al., "A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models," arXiv (Cornell University), vol. 24, pp. 6491–6501, Aug. 2024, doi: https://doi.org/10.1145/3637528.3671470.

[15]. N. Knoth, A. Tolzin, A. Janson, and Jan Marco Leimeister, "AI Literacy and its Implications for Prompt Engineering Strategies," *Computers and education. Artificial intelligence*, vol. 6, pp. 100225–100225, Apr. 2024, doi: https://doi.org/10.1016/j.caeai.2024.100225.

[16]. E. Lee, N. Gogo, Gil Hwan An, S. Lee, and K. Lim, "ChatGPT-Based Debate Game Application Utilizing Prompt Engineering," Aug. 2023, doi: https://doi.org/10.1145/3599957.3606244.

[17]. P. Chauhan, Rahul Kumar Sahani, S. Datta, A. Qadir, M. Raj, and Mohd Mohsin Ali, "Evaluating Top-k RAG-based approach for Game Review Generation," Feb. 2024, doi: https://doi.org/10.1109/ic2pct60090.2024.10486273.

[18]. M. Hoffmann, J. Fillies, S. Peikert, and A. Paschke, "From Text to Text Game: A Novel RAG Approach to Gamifying Anthropological Literature and Build Thick Games," pp. 246–256, Jan. 2025, doi: https://doi.org/10.5220/0013215400003932.

[19]. I. Papadimitriou, I. Gialampoukidis, S. Vrochidis, Ioannis, and Kompatsiaris, "RAG Playground: A Framework for Systematic Evaluation of Retrieval Strategies and Prompt Engineering in RAG Systems," arXiv.org, 2024. https://arxiv.org/abs/2412.12322.

[20]. M. Fuoli, W. Huang, J. Littlemore, S. Turner, and E. Wilding, "Metaphor identification using large language models: A comparison of RAG, prompt engineering, and fine-tuning," arXiv.org, 2025. https://arxiv.org/abs/2509.24866 (accessed Jan. 12, 2026).

是否符合进度？On schedule as per GANTT chart?

YES

下一步 Next steps:

In the next stage, the project will focus on finalizing the system architecture for the adaptive RPG storytelling prototype. Based on the completed literature review, key design decisions will be made regarding the integration of prompt engineering and retrieval-

augmented generation, including the definition of core modules such as narrative state management, retrieval components, and response generation logic. This stage aims to translate theoretical insights from prior work into a coherent and implementable system design.

Following the architectural design, the project will proceed with the implementation of the narrative generation pipeline. This includes constructing a basic retrieval mechanism for accessing contextual information and world state, as well as integrating it with large language model–based response generation. In parallel, a simple text-based interaction interface will be developed to enable player input and system output, allowing early testing of end-to-end functionality.

Finally, preliminary testing and iterative refinement will be conducted to evaluate basic narrative coherence and system stability. Initial experiments will focus on validating prompt structures, retrieval behavior, and their interaction during multi-turn dialogue. Feedback from these tests will be used to adjust prompt templates and system logic, laying the foundation for more comprehensive evaluation and user testing in later stages.