# Research on An Independent Game Intelligent NPC Adaptation Solution Based on Lightweight LLMs—Taking "Ghost Valley: Eight Wastelands" as An Example

Ziquan Wang

{outlook_B2D6DB9E9CCA7FDE@outlook.com}

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China

**Abstract**: Independent game development has long been constrained by the rigidity of NPC interaction models, while haphazardly introducing heavy, large language models (LLMs) faces the practical barrier of excessively high computational costs. To resolve this engineering dilemma, this paper constructs and implements a lightweight LLM-based intelligent NPC adaptation framework. Using the cultivation-themed game "Ghost Valley: Eight Directions" as a testbed, the study integrates the DeepSeek-r1-14B model at the core layer and locally deploys it with INT4 quantization technology, fully establishing the decoupled three-tier chain of "model-API-engine." Test data demonstrates that this solution runs smoothly on ordinary consumer-grade PCs. It not only maintains the logical coherence of NPC dialogue and character consistency but also reduces memory usage and inference latency to levels acceptable for independent teams. Starting from engineering practice, this research provides a cost-effective Artificial Intelligence(AI) implementation reference for development teams with limited computational resources.

**Keywords:** Intelligent NPC adaptation framework; Lightweight large language model; Independent game development; INT4 quantization; Computational cost optimization

## 1 Introduction

In recent years, the explosive growth of large language models (LLMs) has indeed made NPCs in games increasingly lifelike[1, 2, 3]. However, for indie game teams with tight budgets, this remains an elusive technological dividend. Traditional static script trees are time-consuming to write and easily lead to player fatigue; conversely, directly interfacing with GPT-3.5 or LLaMA 2-70B would incur exorbitant API costs and stringent hardware requirements, instantly draining small teams' development funds.

This issue has recently drawn academic attention[4, 5, 6, 7]. For instance, Li Dianfeng noted that generative AI NPCs hold great promise for enriching video game narratives but face significant deployment barriers due to computational costs[8]. At the practical application

ICIAAI 2026, May 22-24, Singapore, Singapore

DOI 10.4108/eai.22-5-2026.2365241

level, while Wang et al. demonstrated the feasibility of AI-driven dynamic campaigns in *Tripo Clash*, their solution heavily relies on expensive cloud-based LLM APIs, making it difficult to scale to small/medium-sized development teams[9]. Additionally, studies like Kawai's on constrained game development strategies and Sakura et al.'s generative AI-powered NPC dialogue frameworks exist[10,11]. Yet, closer examination reveals these studies either remain at the proof-of-concept stage or still lack full independence from cloud computing, failing to deliver a dedicated engineering solution for lightweight local deployment in indie games. Thus, this paper adopts a clear approach: maximizing interactive experience under limited computational resources. Using *Ghost Dynasty: Eight Profound Professions* as the test subject, this paper employs open-source lightweight LLMs combined with model quantization techniques to develop a highly deployable NPC system, focusing on its practical usability in constrained computing environments.

## 2 System architecture design

How to clean up the system and make it easy to reuse in the later stage is the first step in engineering implementation. This article adopts the most classic three-layer decoupling architecture of model layer, API adaptation layer, and engine layer for this purpose.

The bottom model layer directly abandons the complex cloud-based multi-model scheduling and single point locks DeepSeeker r1-14B (INT4 quantized version). This layer is pulled up locally through the Olama environment and independently handles all text generation tasks with minimal resource overhead.

The API adaptation layer in the middle utilizes the FastAPI framework to build RESTful services. It acts as a data relay station, specifically responsible for cleaning and routing the fragmented data (such as concurrent requests and context parameters) passed over from the game engine. After the model declares it, it is then serialized into a standardized format and sent back to the game.

The top engine layer is mainly designed with a dedicated communication plugin for Unity (2022 and above versions). The benefits of doing so are immediate. Developers don't have to touch the game's underlying legacy code at all. They can directly mount preset components onto game objects and inject scene information and trigger interactions in the front-end UI.

## 3 Model selection and quantitative performance

Choosing a model is actually a process of walking a tightrope between "logical reasoning ability" and "video memory consumption". After comprehensive consideration, DeepSeeker r1-14B has the best cost-effectiveness performance. Considering that RTX 3060 or 4060-level graphics cards are commonly installed in the chassis of independent developers and gamers, this paper performed INT4 quantization compression on the model.

In order to see how useful quantification is, this article specifically selected 100 typical dialogues from "Ghost Valley Eight Wilderness" and conducted a stress test. The specific changes in core indicators are shown in Table 1. Table 1 intuitively reflects the huge engineering benefits brought by quantization operations: previously, FP16 required 22GB of video memory

for precision, but now it has been forcibly reduced to 8GB (a decrease of 64%); Previously, a single inference would take 1.5 seconds, but now the result can be obtained in 0.8 seconds (with a speed increase of 47%). Although the cost is a 4% decrease in dialogue accuracy and a 3% decrease in relevance, this small precision loss is almost imperceptible in actual gaming scenarios, and it is completely logical to exchange it for the absolute smoothness of the system.

**Table 1.** Performance Comparison of DeepSeeker r1-14B Model Before and After Quantization

| Performance indicators | Original model (FP16) | Quantitative model (INT4) | Change amplitude |
|---|---|---|---|
| Memory usage (GB) | 22 | 8 | Reduced by 64% |
| Generation speed (s/piece) | 1.5 | 0.8 | Speed up by 47% |
| Dialogue accuracy (%) | 94% | 90% | Loss of 4% |
| Dialogue relevance (%) | 96% | 93% | Loss of 3% |

## 4 Scene adaptation and interface implementation

Architecture alone is not enough; it must be tested in a real environment. This article selects the three most interactive typical scenes in "Ghost Valley Eight Wilderness" for adaptation: trading in markets, experience in Yongning Prefecture, and alchemy.

Taking "marketplace trading" as an example, the underlying communication adopts a stateless design to support a dynamic perception of the scene. When the front-end initiates a POST request, the system will package game_stcene (current location), role_info (NPC's basic character and mounted tasks), and user_query (pure text input by the player) for transmission. Assuming the player clicks on the phrase 'How much does the Foundation Building Pill cost? How can I obtain it?', the model, after understanding the prompt context, sends back a real-time message similar to 'The Foundation Building Pill is priced at 500 spirit stones and can be purchased or obtained through alchemy in the marketplace'. This dynamic response not only firmly adheres to the boundaries of the character design of the "neighborhood shopkeeper" but also accurately issues task guidance clues, and the entire interactive experience far exceeds the traditional predetermined branch of writing.

## 5 Evaluation indicators and experimental expectations

The evaluation of the system was divided into two parts: objective performance and subjective experience. The hardware benchmark environment was uniformly set as an Intel i9-14900HX processor with 16GB of memory and an RTX 4060 graphics card.

In terms of system carrying capacity, the key is to monitor whether the API interface will crash under 100 consecutive high-frequency request bombardment, and it is required that the average delay of the network and inference must be firmly stuck within 200ms to prevent front-end screen lag.

In the subjective interaction testing section, a 5-point Likert scale was introduced, and 30 test players were organized to conduct A/B blind tests with the original script NPC. The scoring dimensions were refined to include the degree of mismatch between the character design and worldview, task clue extraction efficiency, and contextual coherence. Based on current convergence data, this solution has an overall interaction fit that is over 40% higher than traditional scripts. This also indirectly confirms one point: as long as the large model is properly tuned, even if it is a lightweight version with precision cut, the narrative immersion it brings to players still has strong technical penetration.

## 6 Conclusion

Ultimately, the research motivation of this article is to unravel the deadlock of independent gaming teams that lack computing power but want to become highly intelligent NPCs. By introducing the DeepSeeker r1-14B model into the development process and forcefully implementing INT4 quantization, this paper has confirmed that this technological path is completely feasible on a regular PC with only 8GB of VRAM. The previously unattainable large language model has successfully lowered the video memory threshold by 64%, with a single sentence response time of less than one second, and the cost-effectiveness of engineering implementation is directly maximized.

More importantly, the three-layer decoupling architecture of "Model API Engine" people have built encapsulates all the complex AI logic at the bottom like a black box. This means that even independent game planners or programs that do not understand the underlying algorithms can quickly pull up an intelligent NPC system like building blocks by installing a Unity plugin, greatly reducing the cost of trial and error.

In the actual testing of specific business scenarios in "Ghost Valley Eight Wilderness", the lightweight LLM, which is strongly constrained by specific Prompt, demonstrated extremely high literacy. It not only brings to life various shopkeepers and casual cultivators in the world of cultivation, but also executes the core game mechanism of sending missions and passing clues in a stylish and immersive way, far surpassing traditional script dialogue trees.

Of course, the current middleware solutions are not yet perfect. The next research plan is very clear: firstly, people will introduce quantitative indicators of natural language processing, such as BLEU, to explore how rich the diversity of NPC dialogue texts is; Secondly, it is necessary to establish a mathematical model to accurately calculate the success rate of AI-guided players in completing tasks. Once these foundations are firmly established, people will also throw this framework into the complex narrative network of "Sect Game" with multiple interwoven threads, striving to accumulate a truly useful and robust AI infrastructure for the independent game open source community.

## References

[1] Z. Wu, Z. Chen, D. Zhu, C. Mousas, and D. Kao, "A Systematic Review of Generative AI on Game Character Creation: Applications, Challenges, and Future Trends," IEEE Transactions on Games, 2025, doi: 10.1109/TG.2025.3564869.

[2] A. Valdivia and P. Burelli, "Evaluating Quality of Gaming Narratives Co-Created with AI," in Proc. IEEE Conf. Games (CoG), Lisbon, Portugal, 2025, pp. 1-4, doi: 10.1109/CoG64752.2025.11114354.
[3] M. Sas, "Unleashing Generative Non-Player Characters in Video Games: An AI Act Perspective," in Proc. IEEE Gaming, Entertainment, and Media Conf. (GEM), Turin, Italy, 2024, pp. 1-4, doi: 10.1109/GEM61861.2024.10585442.
[4] Y. Kannaka, H. Iima, and M. Yoshisada, "Generation of Game Levels Using A GPT," in Proc. SICE Festival with Annual Conf. (SICE FES), Chiang Mai, Thailand, 2025, pp. 1034-1039, doi: 10.23919/SICEFES67750.2025.11236691.
[5] B. Patel, P. Shinde, D. H. Joshi, D. S. Rathod, P. R. Vichare, and S. P. Yadav, "Generative AI for Dungeons and Dragons," in Proc. 6th Int. Conf. Data Intell. and Cognitive Informatics (ICDICI), Tirunelveli, India, 2025, pp. 1871-1876, doi: 10.1109/ICDICI66477.2025.11135422.
[6] A. C. H. Chen, "Training Strategy Based on Game Theory for Generative AI," in Proc. Int. Conf. Smart Syst. for applications in Electrical Sciences (ICSSES), Tumakuru, India, 2024, pp. 1-6, doi: 10.1109/ICSSES62373.2024.10561423.
[7] B. Marr, "Video Game Design and Testing: The Generative AI Approach," in Generative AI in Practice: 100+ Amazing Ways Generative Artificial Intelligence is Changing Business and Society, Wiley, 2024, pp. 155-164.
[8] D. Li, "Digital Labyrinth and Waxy Wings: Prospects and Challenges of Generative Intelligent NPCs in Video Games," Journal of Beijing Film Academy, no. 12, pp. 39-48, 2024.
[9] Z. Wang et al., "AI-Driven Dynamic Video Game Battles: Case Study of the Design of Tripo Clash," in Proc. 28th Int. Conf. Comput. Supported Cooperative Work in Design (CSCWD), Compiegne, France, 2025, pp. 360-366, doi: 10.1109/CSCWD64889.2025.11033545.
[10] Y. Kawai, "Using Generative AI for Game Development Subject to Technical Constraints," in Proc. Int. Conf. Cyberworlds (CW), Kofu, Japan, 2024, pp. 376-377, doi: 10.1109/CW64301.2024.00081.
[11] M. Sakura, K. Tasaka, M. Numao, and S. Nishide, "NPC Dialogue Generation in a Custom-Built Game Using Generative AI," in Proc. IEEE 14th Global Conf. Consumer Electron. (GCCE), Osaka, Japan, 2025, pp. 416-420, doi: 10.1109/GCCE65946.2025.11275322.