Dynamic Gimmick Learning for Navigation Agents in Social VR Through User-Agent Dialogue

Atsuya Matsumoto*
Cluster Metaverse Lab
The University of Tokyo

Hikari Yanagawa†
Cluster Metaverse Lab
University of Tsukuba

Yuichi Hiroi‡
Cluster Metaverse Lab

Yuji Hatada§
The University of Tokyo

Takuji Narumi¶
The University of Tokyo

Takefumi Hiraki‖
University of Tsukuba
Cluster Metaverse Lab

Figure 1: (a) Navigation Pixie [3] learns the principles of gimmick operation through user dialogue, acquiring knowledge that touching the white ball activates the door mechanism. (b) When encountering a structurally similar gimmick at a different location, the agent autonomously applies its acquired knowledge to solve the gimmick. (c) Upon encountering an analogous gimmick featuring a black ball, the agent adapts its learned knowledge to operate the gimmick based on the user's hint.

Abstract

User-generated interactive mechanisms ("gimmicks") in metaverse platforms offer unique experiences, but their complexity hinders player exploration. Existing navigation agents, reliant on static metadata, cannot comprehend these dynamic gimmicks. We propose the "Conversational Knowledge Acquisition Loop," a method for agents to dynamically learn gimmicks through player conversation. A preliminary study showed that our agent successfully acquired knowledge via conversation and applied it to solve analogous gimmicks.

Index Terms: Autonomous Agent, Virtual Reality, Embodied Conversational Agent, and Social VR.

1 Introduction

On metaverse platforms such as VRChat[1], Resonite[2], and Cluster[3], players can freely create virtual spaces ("worlds") and interactive mechanisms ("gimmicks") as user-generated content (UGC), leading to the development of numerous unique worlds [2]. Players can visit these worlds and interact with various gimmicks, which are triggered to induce changes to the world or player, enabling immersive experiences. However, while this high level of freedom offers new experiences, it also presents a challenge for first-time visitors, who may struggle to understand how to enjoy the world and may abandon their exploration.

To solve this problem, Yanagawa et al. proposed Navigation Pixie, a navigation agent that integrates Large Language Models (LLMs) with structured spatial information called "metadata" [3]. The system uses navigation point information retrieved via platform APIs as metadata to provide adaptive spatial guidance in response to player requests. However, due to the limitations of metadata, the agent struggles to understand interactive gimmicks. The current navigation method retrieves predefined metadata (e.g., names, coordinates, and detailed descriptions). In contrast, the metadata for interactive gimmicks consists of variable names and values, which are not designed to convey linguistic information to the LLM. For instance, if a gimmick is intended to "open a door," it can retrieve the variable name (e.g., `OpenDoor`) and its value (e.g., 0 or 1), but it cannot ascertain from the metadata whether interacting with the gimmick actually opens the intended door or if there are accompanying auditory or visual changes.

To address this limitation, this research proposes a method called Conversational Knowledge Acquisition Loop (CKAL), which elicits and utilizes knowledge obtained from conversation with the player. The agent acquires accurate knowledge of gimmicks by engaging in conversation to confirm the effects after a gimmick is executed. To validate the effectiveness of the proposed method, we conducted a preliminary investigation. The results showed that the proposed method can acquire correct knowledge of the gimmicks through conversation and apply this knowledge to efficiently solve analogous ones. This suggests the potential to realize more dynamic guidance that aligns with player perception.

*e-mail: atsuya.maya@cyber.t.u-tokyo.ac.jp
†e-mail: h.yanagawa@cluster.mu
‡e-mail: y.hiroi@cluster.mu
§e-mail: hatada@cyber.t.u-tokyo.ac.jp
¶e-mail: narumi@cyber.t.u-tokyo.ac.jp
‖e-mail: hiraki@slis.tsukuba.ac.jp

[1] https://hello.vrchat.com/
[2] https://resonite.com/
[3] https://cluster.mu/

2 System Architecture

Conversational Knowledge Acquisition Loop Our navigation agent is based on Navigation Pixie [3], enhanced with CKAL. Following the architectural pattern of the Hierarchical Language Agent (HLA) [1] by Liu et al., CKAL comprises three hierarchical components: (1) The **Planning Component** decomposes player requests into sequences of sub-goals. (2) The **Sub-goal Component** generates concrete action plans to achieve each sub-goal provided by the Planning Component. (3) The **Tool Component** receives tool execution requests, populates them with specific parameter values, and invokes the corresponding APIs on the metaverse platform.

Knowledge Acquisition from Dialogue CKAL acquires information through three dialogue scenarios: (1) In **failure recovery**, when the agent encounters unresolvable errors, the system executes a replanning process that incorporates user guidance. (2) In **goal clarification**, the LLM confirms the user's intentions regarding instructions or advice to specify task objectives. (3) **Action result confirmation** occurs when gimmicks undergo state changes in metadata. The system acquires information about changes that are not directly observable through user dialogue and constructs comprehensive gimmick knowledge.

These dialogue logs are stored as structured data, incorporating timestamps and spatial coordinates. During subsequent action planning phases, the LLM retrieves relevant memories from this accumulated data and uses the acquired knowledge to reconstruct action strategies.

3 Experiment

3.1 Research Questions and Experimental Design

We conducted a preliminary experiment to validate two research questions:

RQ1: Does gimmick knowledge acquisition through user dialogue improve task execution efficiency compared to using only metadata?

RQ2: Can the knowledge acquired from dialogues be applied to solving and guiding similar novel gimmicks?

We constructed an experimental environment containing gimmicks A and B requiring two-stage reasoning (black ball → white ball sequence opens door) and gimmick C with a different sequence (white ball → black ball sequence opens door).

3.2 Experimental Conditions

Four experimental conditions were established: (i) *dialogue with knowledge*, (ii) *instruction-only with knowledge*, (iii) *dialogue without knowledge*, (iv) *instruction-only without knowledge*. *Dialogue* conditions involved collaborative task execution through experimenter interaction, while *instruction-only* conditions provided initial experimenter instructions only. Also, *knowledge* conditions used information acquired from dialogue and metadata for the subsequent resolution of gimmicks.

3.3 Procedure

The experimental procedure proceeded as follows. First, the agent was placed at the start location with initialized memory and instructed to cross the door at location A. Movement invariably failed due to the closed door state. Each condition attempted gimmick A resolution through dialogue or trial-and-error approaches.

Subsequently, the location B movement was instructed regardless of the outcome of the gimmick A. Dialogue conditions received the advice "try the same method as A." Finally, identical procedures were executed for gimmick C, with dialogue conditions receiving "try the reverse sequence from A and B" guidance. Due to the preliminary nature, each condition was subjected to a single trial.

Table 1: Quantitative Performance Comparison by Task

Task	**(i)**	**(ii)**	**(iii)**	**(iv)**
A	✓ (12 / 2)	✓ (19 / 3)	✓ (10 / 2)	× (8 / 4)
B	✓ (6 / 1)	✓ (8 / 1)	× (6 / 2)	× (14 / 4)
C	✓ (12 / 2)	✓ (14 / 3)	× (6 / 2)	× (7 / 4)
Total	✓ **(30 / 5)**	✓ **(41 / 7)**	× **(22 / 6)**	× **(29 / 12)**

Format: Status (# of Actions / # of Planning).
✓ = Success, × = Failure.

4 Results

4.1 Task Performance Metrics

Table 1 shows the result. The number of total actions represent the sum of tool executions per task, while the number of total plans indicate planning iteration counts. From the result, the knowledge retention conditions (i and ii) achieved the completion of the gimmick C compared to the no-knowledge conditions (iii and iv). Additionally, dialogue condition (i) achieved task success with fewer actions and plans than instruction-only condition (ii).

4.2 Knowledge Acquisition Analysis

From this experiment, we can obtain the following preliminary answers to the RQs:

4.2.1 RQ1: Dialogue-Based Learning Effectiveness

Dialogue with knowledge condition (i) confirmed accurate acquisition of gimmick knowledge through user dialogue. Following experimenter feedback stating "the door opened," the system accurately generated causal relationship knowledge: "door opening through black ball operation followed by white ball operation." Conversely, instruction-only condition (ii) exhibited knowledge generation lacking confidence in action results, stating "related door opening status unclear" due to insufficient feedback.

4.2.2 RQ2: Knowledge Transfer Capability

Dialogue with knowledge condition (i) demonstrated the application of acquired knowledge to similar gimmicks. During gimmick C resolution, responding to experimenter advice "try reverse sequence from A and B," the system confirmed intention through dialogue: "white ball then black ball sequence, correct?" and successfully resolved gimmick C using accurate operation sequence. In contrast, instruction-only with knowledge condition (ii) lacked dialogue-based trajectory correction, repeatedly executing black-to-white operations until white-to-black patterns emerged coincidentally, resulting in fortuitous gimmick C resolution.

5 Conclusion

In this research, we propose a method for an agent to dynamically acquire knowledge through player conversation and apply it to its guidance. The results of our ablation study confirmed that the knowledge enables the agent to efficiently solve similar gimmicks and accurately explain their functions. Future work will focus on improving the robustness of the agent in UGC worlds with diverse gimmicks.

Acknowledgments

This work was partially supported by JST Moonshot Research & Development Program Grant Number JPMJMS2013.

References

[1] J. L. et al. Llm-powered hierarchical language agent for real-time human-ai coordination, 2024. 2

[2] G. D. Ritterbusch and M. R. Teichmann. Defining the metaverse: A systematic literature review. *IEEE Access*, 11:12368–12377, 2023. doi: 10.1109/ACCESS.2023.3241809 1

[3] H. Yanagawa, Y. Hiroi, S. Tokida, Y. Hatada, and T. Hiraki. Navigation pixie: Implementation and empirical study toward on-demand navigation agents in commercial metaverse. In *2025 IEEE International Symposium on Mixed and Augmetend Reality (ISMAR)*, to appear, 2025. 1, 2