Integrating Cognitive Agents into Visual Novels

Sandra Sačarić, Markus Schatten
Artificial Intelligence Laboratory
University of Zagreb
Faculty of Organization and Informatics
Pavlinska 2, 42000 Varaždin, Croatia
ssacaric20@student.foi.hr,markus.schatten@foi.unizg.hr

Abstract. *In this paper we introduce the integration of cognitive agents in visual novels to enhance player interaction and immersion as part of an experimental student project during a Game Development Platforms course. Traditional visual novels rely on predetermined dialogue options, limiting the player's engagement and creating a static narrative experience, while our approach uses cognitive agents for real-time, dynamic conversations. The agent, trained on an extensive question-answer database, improves the realism of in-game dialogues. Despite challenges in linking the cognitive agent with the visual novel, our work marks a significant step towards more interactive and lifelike gaming experiences.*

Keywords. cognitive agents, natural language processing, AI integration, visual novel, game development, interactive storytelling, Monogatari engine, JavaScript, Python, Chatterbot

1 Introduction

Cognitive agents (Markus Schatten, Đurić, Peharda, and Tomičić, 2022), as sophisticated software entities capable of simulating human-like cognitive functions, have become a pivotal area of research and development in artificial intelligence (AI). Cognitive agents, as opposed to chatbots, have the ability to understand, learn, and interact within complex environments using various cognitive interfaces possibly including visual as well as audio (especially human language) interfaces based on various text to speech (TTS) and speech to text (STT) technologies. Additionally, they usually have an animated avatar making their presence even more realistic. Such characteristics make them ideal for enhancing user experiences across various domains, including interactive digital entertainment. This paper explores the innovative integration of cognitive agents into visual novels, aiming to transform traditional narrative experiences into dynamic, immersive interactions.

The core concept of cognitive agents lies in their capacity for natural language understanding, decision-making, and adaptive learning. Unlike static, hard-coded systems, cognitive agents can process and respond to user inputs in real-time, offering personalized and engaging interactions. These agents operate on datasets, making use of machine learning algorithms to continuously improve their performance and adapt to new scenarios. This approach not only broadens the scope of player interaction but also sets a standard for the application of cognitive agents in narrative-driven game environments.

Visual novels (Cavallaro, 2009) are interactive, narrative-driven video games that combine written text, static or animated graphics, and sound to tell a story, often with branching plotlines and multiple endings based on player choices. They are a popular genre in Japanese gaming culture, known for their rich storytelling and immersive experiences, spanning various genres such as romance, mystery, and fantasy.

The integration of cognitive agents into visual novels presents unique challenges, particularly in terms of ensuring the agents are embedded properly within the visual novel. However, the potential benefits, including more lifelike character interactions and adaptive storytelling, make this a promising area for future exploration. Through this work, we aim to demonstrate how cognitive agents can elevate the interactivity and immersion of digital narrative experiences, making way for more user-centered game design.

Herein we will showcase an experimental implementation of a visual novel game that includes cognitive agents as actors in-game. The integrated cognitive agents are based on the Beautiful ARtificial Intelligence Cognitive Agent (B.A.R.I.C.A.) architecture that we have developed in previous work (Markus Schatten, Đurić, and Peharda, 2021; Markus Schatten, Đurić, Peharda, and Tomičić, 2022; Markus Schatten, Protrka, et al., 2021; Markus Schatten, Marinela Schatten, and Inkret-Martinčević, 2023; Šokec, 2019). For the implementation we have used the Monogatari visual novel engine[1]. The experimental project has been conducted during a Game Development Platforms undergraduate course at the Faculty of Organization and Informatics University of Zagreb in which a team of 12 students

[1] See https://monogatari.io/

have been tasked to develop a visual novel by integrating cognitive agents technology under the guidance of the course lecturer.

The rest of this paper is organized as follows: firstly in section 2 we provide an overview of related work. Then, in section 3 the methodology of implementation is shown. In section 4 we showcase our results and in section 5 we draw our conclusions and provide guidelines for future research.

2 Related Work

While there has been numerous applications of chatbots in various gaming contexts (Khan and Sabahat, 2024) including but not limited to chatbots in online communities (Seering et al., 2020), virtual reality games Ren et al., 2020 as well as serius games (Van Rosmalen et al., 2012) there is, up to our best knowledge, no direct precedent for the specific application of cognitive agents in visual novels. Nevertheless, there are some related works that demonstrate the potential of cognitive systems within various forms of entertainment media. Two notable examples are the multimodal chatbot (Bhushan et al., 2020). and the cognitive modeling approach (Funge, 1999).

Bhushan et al. (2020) present No Body lives here (ODO), a sophisticated multimodal chatbot for interactive theater performances. This chatbot integrates vision and natural language processing capabilities, performing tasks like face detection and crowd movement tracking. While it operates within a live theatrical context, its relevance to our research lies in demonstrating how cognitive agents can manage complex, real-time interactions within a narrative framework. However, ODO's focus on multimodal inputs and its movable LED embodiment significantly differs from our work, which emphasizes integrating cognitive agents for natural dialogue and storytelling in a digital visual novel setting.

John Funge's book, "AI for Games and Animation," discusses cognitive modeling for creating autonomous characters in animated and real-time game contexts. Funge's focus on advanced character animation and automated cinematography underscores the role of cognitive modeling in enhancing digital character realism and autonomy. While relevant to our project, his approach centers on animation and gameplay, differing from our work, which uses cognitive agents to generate dynamic dialogue and respond to player inputs in visual novels.

Both works highlight the versatile application of cognitive systems in entertainment media but do not address integrating cognitive agents in visual novels.

Additionally, in our previous work we have used our B.A.R.I.C.A. cognitive agent architecture (overview shown on Fig.1) for the implementation of various application including for university student support (Markus Schatten, Đurić, and Peharda, 2021; Šokec,

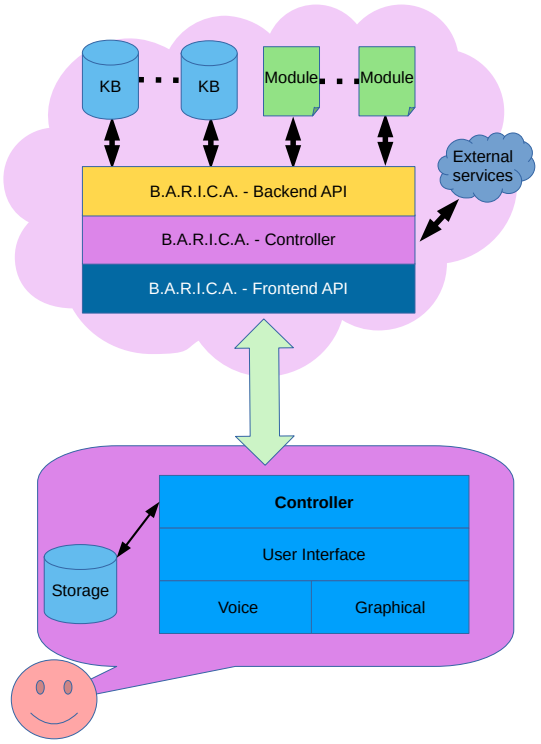

Figure 1: The B.A.R.I.C.A. cognitive agent architecture overview (Markus Schatten, Đurić, and Peharda, 2021)

2019), smart mobility (Markus Schatten, Đurić, Peharda, and Tomičić, 2022), telemedicine (Markus Schatten, Protrka, et al., 2021) as well as primary school education (Markus Schatten, Marinela Schatten, and Inkret-Martinčević, 2023). Some of the implementations of these applications are shown on Fig.2

Herein we aim to extend the possible application of cognitive agents to visual novels to transform static dialogue systems in visual novels into dynamic, responsive interactions, introducing unprecedented realism and interactivity in the genre, setting a new standard in interactive storytelling.

3 Methodology

At the beginning of a 15 week Game Development Platforms course a team of 12 students had been tasked to develop a visual novel game using the Monogatari engine, incorporating cognitive agents based on the B.A.R.I.C.A. cognitive agents architecture. The project involved creating and preparing a visual novel game where interaction is achieved through speech or menu selections. The game was to be hosted on a server running Debian or Ubuntu OS, and designed in an adventure style. The team needed to create a narrative with alternative endings, design characters (cognitive agents), and implement machine learning mod-

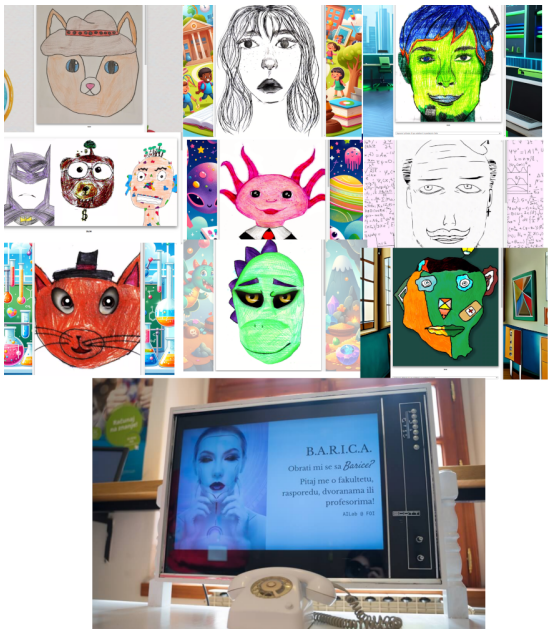

Figure 2: Some incarnations of the B.A.R.I.C.A. cognitive agents (Markus Schatten, Marinela Schatten, and Inkret-Martinčević, 2023)

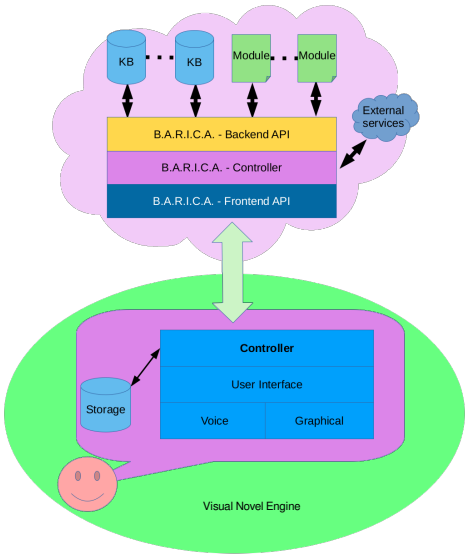

Figure 3: Extended B.A.R.I.C.A. architecture

els using the Python Chatterbot[2] module. Character animations should have been created using tools like Crazy Talk[3] or other speech animation applications.

The team's responsibilities included creating a project plan and Game Desing Document (GDD), designing the game's graphical user interface (GUI), developing character animations representing cognitive agents, and ensuring the use of free/open-source assets compliant with licensing. Additionally, the team had to implement, test the game, and develop an update system to pull and install the latest code from the project repository. Comprehensive project documentation detailing all components and usage instructions should have been prepared, with all code and documentation published in the project repository.

Additionally, the team had to elect leaders which were tasked to organize the team into departments, oversee the work on the project as well as to report to "upper management" (the lecturer). In the following we detail their development process of creating and embedding cognitive agents within the visual novel, going over details regarding how the models had been trained, the server communication setup, and integration of the bot with speech recognition capabilities. The newly developed system roughly followed the extended architecture shown on Fig.3.

3.1 Game Desing and Storytelling

The storytelling department was instrumental in designing the overall narrative and characters for the visual novel "Betrayed Trust." They crafted an immersive fantasy world where the player, as a guild member, must navigate a complex web of trust and betrayal. The narrative focuses on the player's interactions with their two companions, Bizarr, a mysterious assassin, and Rioth, a fearless warrior. The story has alternative endings, hinging on the player's choices and interactions with these characters. The team emphasized creating a dynamic storyline where the player's decisions influence the narrative's progression, allowing for a personalized experience.

In designing the characters, the team utilized AI-generated visuals to create distinct and memorable protagonists. Bizarr and Rioth (shown on Fig.4), the two main characters, were designed to reflect their complex personalities and roles within the story. Despite initial technological limitations, which prevented more detailed hand-drawn designs, the team managed to create compelling character animations.

3.2 Bot Training and Development

Initially, the team utilized the ChatterBot library for training individual cognitive agents. Each agent was tailored with a unique personality by crafting a specific database of questions and answers. This approach ensured that each agent responded distinctly, aligning with its predefined character traits and narrative role within the visual novel.

The dialogue script was finalized by the storytelling department, creating a structured database of potential questions and corresponding responses for each character. This database included not just direct responses but also context-specific variations to reflect the character's personality and role in the narrative.

For each agent, there was a Chatterbot instance con-

[2]See https://pypi.org/project/ChatterBot/
[3]See https://crazytalk.reallusion.com/animator.html

Figure 4: Character design generated by AI

figured with custom logic adapters, one tasked with identifying and returning responses based on the user's input, which is determined based on which response matches the input best. The other logic adapter was introduced to handle inputs that didn't confidently match any response. This prompts the user to clarify their input and ensures that the agent's responses are relevant and appropriate regarding the context. This will later be brought up in a bit more detail.

3.3 Integration

In order to integrate the cognitive agents into the Monogatari visual novel engine a special Monogatary component was developed that allowed for the inclusion of the client cognitive agent code as characters in the visual novel.

To enable real-time interaction between the cognitive agents and the backend server, the team implemented a WebSocket communication protocol. A dedicated Python class on the Flask[4] based backend side managed WebSocket communication, enabling continuous messaging between the agents and the server. This setup included configuring the ChatterBot instance with custom logic adapters and a SQLite[5] database to store conversation history and manage response variability.

The Python class included methods to initialize a read-only bot instance and handle incoming and outgoing messages. The key features included continuous listening, which kept the bot in a permanent state of actively listening to the surrounding sounds, processing and responding to messages in real-time, as well as message buffering, with the buffer ensuring that responses remained unique by checking for duplicates and maintaining a history of recent interactions.

A Python script was developed to manage the integration of the cognitive agents with web-based functionalities. This script utilized Flask for serving web content and WebSocket capabilities to handle bidirectional communication. The script offered a flexible architecture, supporting both training mode for bot refinement and real-time interaction mode for seamless gameplay experiences.

The script operated in two modes. First, the training mode, meant for training the chatbot and using the provided data. Then the real-time interaction mode, which would initialize the server and Flask application to handle live user interactions.

Additionally, the WebSocket server ran on a separate thread, to ensure that web content was effectively managed via Flask on the main thread. This allowed for efficient multitasking and ensured a responsive user interface.

Integrating speech recognition was a necessary step, in which lied the key to the interactivity of our cognitive agents. Th team implemented this using the JavaScript speech recognition API[6], allowing the system to process user speech and generate appropriate responses.

A JavaScript object was initialized to capture and transcribe spoken inputs. This transcript was then sent to the WebSocket server for processing by the cognitive agents. The script also included handlers for various events as was previously mentioned, one which was triggered when speech was successfully recognized, prompting the bot to respond. The other being used for error handling; managed scenarios where speech recognition failed or returned low confidence, ensuring robust user interaction.

To reach our intended goal of the bot being visually and audibly supported by voice lines and character animations, a dedicated JavaScript function controlled audio and video playback based on user-triggered questions during conversations. This function synchronized audio segments with corresponding video segments, optimizing the timing and duration of multimedia elements based on contextual dialogue within the visual novel. This was achieved by using a single audio and video file containing all recorded answers. Each answer had a manually marked timestamp, allowing a playback of specific segments according to the user's input.

4 Results

By the end of the project, the team had successfully developed dynamic and contextually aware cognitive agents that significantly enhanced the interaction experience within the visual novel.[7] These agents were

[4]See https://flask.palletsprojects.com/en/3.0.x/

[5]See https://www.sqlite.org/index.html

[6]See https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API/Using_the_Web_Speech_API

[7]See the final GitHub repository here https://github.com/AILab-FOI/PRRI-CognitiveAgents2024 as well as the corre-

meticulously trained using the ChatterBot library, each equipped with a unique personality and a structured database of potential questions and answers, which ensured distinct and authentic responses. The agents' interactions were made more immersive through the integration of supporting audio and video materials, including voice lines and character animations, adding depth to the user experience. Fig.5 shows the opening screen of the visual novel game[8].

Figure 5: Opening screen of the visual novel with a cognitive agent shown

The cognitive agents were embedded within the visual novel through a WebSocket communication system that enabled real-time interaction between the agents and the game environment. This system, managed by a dedicated Python class, enabled continuous messaging and real-time responses, maintaining a fluid conversational flow. Additionally, the agents were enhanced with speech recognition capabilities, allowing them to respond to user inputs in real-time. This feature allowed recognized speech transcripts to be sent to the server, enabling the agents to process and respond dynamically.

Everything was further supported by an audio and video control system, which synchronized multimedia elements with the agents' dialogue based on user-triggered questions. This ensured an engaging user experience, blending visual and auditory components effectively to create a more believable and interactive storytelling environment.

5 Conclusion & Future Research

The integration of cognitive agents into visual novels, as demonstrated by the "Betrayed Trust" project, showcases a significant advancement in interactive digital storytelling. By employing cognitive agents capable of real-time dialogue and adaptive responses, the project has elevated the narrative experience beyond traditional static interactions. The use of the B.A.R.I.C.A. architecture and the Monogatari engine provided a robust framework for embedding these intelligent agents, resulting in a more immersive and engaging user experience. The successful implementation of speech recognition and real-time multimedia synchronization further enhanced the interactivity and realism of the game, marking a noteworthy achievement in student-led game development projects.

Despite the project's success, several challenges were encountered, particularly in integrating cognitive agents with the visual novel framework and ensuring smooth real-time interactions. These difficulties highlighted the need for more sophisticated tools and techniques for seamlessly blending AI-driven dialogue systems with traditional game engines. Future research should focus on refining these integration processes, exploring more advanced machine learning models for dialogue generation, and enhancing the scalability of cognitive agents in more complex game environments. Additionally, user feedback on the interactive experience might provide valuable insights for improving the design and functionality of cognitive agents in future projects.

Future research could also investigate the application of cognitive agents in other genres of interactive fiction and gaming. Expanding beyond visual novels to include role-playing games (RPGs), simulations, and other narrative-driven experiences can further demonstrate the versatility and potential of cognitive agents in enhancing user engagement. Moreover, integrating more advanced AI technologies, such as deep learning and natural language understanding, can significantly improve the responsiveness and realism of these agents. By continuing to innovate and explore new applications, the potential for cognitive agents to transform digital storytelling and interactive media remains vast and promising.

Acknowledgements

This project was carried out under the scope of our faculty, specifically during the Game Development Platforms course, where we explored practical implementations of game development platforms. This was followed through by our mentor, prof. dr. sc. Markus Schatten.

We would like to extend our deepest gratitude to the entire team for their invaluable contributions. Project leaders Jakov Kadić and Sandra Sačarić coordinated tasks and motivated the team, handling documentation, backend support, and visual and content refinement.

Jana Jambrešić handled the UI and made prototypes for the interface, ensuring usability and responsiveness. Luka Pošta created backgrounds, voice lines for the agents, and edited the videos, and also single-handedly made the game's trailer.

Dorijan Kos initially trained the bot and later developed character animations and video materials, while Dora Garafolić designed characters, contributed to the story structure, and assisted with backend integration.

sponding itch.io site of the project here https://ailab-foi.itch.io/prri-cognitiveagents2024

[8]The Fantasy Ren'py GUI template asset pack by Sklolaztika has been used for the user interface https://skolaztika.itch.io/fantasy-renpy-gui-template

From the writing team, Sonja Kolarić shaped the narrative's vision and wrote the story, while Ivan Simić crafted dialogues and implemented narrative elements in JavaScript, ensuring all project requirements were met.

The backend and AI team, comprising Karlo Rosenthal, Frano Šimić, Karmelo Mrvica, and Mateo Zović, collaboratively trained the chatbot and developed server-side solutions, successfully integrating the bot's dynamic HTML display.

Each member's dedication and effort were pivotal in bringing this project to fruition, and their contributions are deeply appreciated.

ChatGPT has been used to to refine the phrasing of parts of this paper.

References

Bhushan, R., Kulkarni, K., Pandey, V. K., Rawls, C., Mechtley, B., Jayasuriya, S., & Ziegler, C. (2020). Odo: Design of multimodal chatbot for an experiential media system. *Multimodal Technol. Interact.*, 68.

Cavallaro, D. (2009). *Anime and the visual novel: Narrative structure, design and play at the crossroads of animation and computer games*. McFarland.

Funge, J. D. (1999). *Ai for games and animation: A cognitive modeling approach*. A K Peters/CRC Press.

Khan, N., & Sabahat, N. (2024). The role of ai chatbots in revolutionizing gaming experiences-a survey. *VFAST Transactions on Software Engineering*, *12*(1), 93–104.

Ren, X., et al. (2020). Artificial intelligence and depression: How ai powered chatbots in virtual reality games may reduce anxiety and depression levels. *Journal of Artificial Intelligence Practice*, *3*(1), 48–58.

Schatten, M. [Markus], Đurić, B. O., & Peharda, T. (2021). A cognitive agent for university student support. *2021 IEEE Technology & Engineering Management Conference-Europe (TEMSCON-EUR)*, 1–6.

Schatten, M. [Markus], Đurić, B. O., Peharda, T., & Tomičić, I. (2022). A cognitive agent's infrastructure for smart mobility. *Transportation Research Procedia*, *64*, 199–204.

Schatten, M. [Markus], Protrka, R., et al. (2021). Conceptual architecture of a cognitive agent for telemedicine based on gamification. *Central European Conference on Information and Intelligent Systems*, 3–10.

Schatten, M. [Markus], Schatten, M. [Marinela], & Inkret-Martinčević, L. (2023). Development of gamified cognitive agents for primary schools-a case study. *Central European Conference on Information and Intelligent Systems*, 3–8.

Seering, J., Luria, M., Ye, C., Kaufman, G., & Hammer, J. (2020). It takes a village: Integrating an adaptive chatbot into an online gaming community. *Proceedings of the 2020 chi conference on human factors in computing systems*, 1–13.

Šokec, T. (2019). Modeliranje kontekstualno svjesnog agenta za razgovor na hrvatskom jeziku uz pomoć konačnog automata i strojnog učenja [Mentor: Markus Schatten].

Van Rosmalen, P., Eikelboom, J., Bloemers, E., Van Winzum, K., & Spronck, P. (2012). Towards a game-chatbot: Extending the interaction in serious games. *The 6th European Conference on Games Based Learning*.