Interactive Storytelling Apps: Increasing Immersion and Realism with Artificial Intelligence?

Pierre-Benjamin Monaco[1][a], Per Backlund[2][b] and Stéphane Gobron[3,*][c]

[1]*Master of Software Engineering, University of Applied Sciences and Arts Western Switzerland (HES-SO),*
avenue de Provence 7, Lausanne, Switzerland

[2]*Department of Game Technologies, School of Computer Science, University of Skövde, Skövde, Sweden*

[3]*HE-Arc School of Engineering, University of Applied Sciences and Arts Western Switzerland (HES-SO),*
Espace de l'Europe 11, Neuchâtel, Switzerland

Keywords: Natural Language Processing, Hostage-Taking, Immersion, Negotiation, Decision-Making, Chatbot, Dialogue, LLM, Interactive Content, Virtual Reality, Multimodal Systems.

Abstract: The advent of Large Language Models (LLMs) has revolutionized digital narration, moving beyond the rigid and time-consuming process of creating conversational trees. These traditional methods required significant multidisciplinary expertise and often disrupted dialogue coherence due to their limited pathways. With LLMs character simulation seems to become accessible and coherent, allowing the creation of dynamic personas from text descriptions. This shift raises the possibility of streamlining content creation, reducing costs and enhancing immersion with interactive dialogues through expansive conversational capabilities. To address related questions, a digital hostage-taking simulation was set up, and this publication reports the results obtained both on the feasibility and on the immersion aspects. This paper is proposed as a twin paper detailing the implementation of a simulation that use an actual mobile phone to communicate with the hostage-taker.

1 INTRODUCTION

Figure 1: Illustration of the dual architecture for interactive storytelling – created using **PIXLR**$^{(R)}$ and **Photoshop**$^{(R)}$ image generators+artistic work. Negotiation is viewed through a conversational tree and as a maze of possibilities with an AI/LLM.

[a] https://orcid.org/0000-0002-4487-8195

[b] https://orcid.org/0000-0001-9287-9507

[c] https://orcid.org/0000-0003-3796-6928

[*] Corresponding author

Monaco, P., Backlund, P. and Gobron, S.
Interactive Storytelling Apps: Increasing Immersion and Realism with Artificial Intelligence?.
DOI: 10.5220/0012720400003758

In *Proceedings of the 14th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2024)*, pages 250-257
ISBN: 978-989-758-708-5; ISSN: 2184-2841

The generation of high-quality interactive content demands a high level of proficiency in storytelling and domain knowledge within which such content is developed. The time dedicated to content research and the creation of a coherent dialogue typically constitutes the major portion of the work (Padilla et al., 2017). In the case of an interactive dialogue constructed in a tree-like structure, the ratio between the quantity of created content and the content explored by users exponentially decreases with the length of the dialogue. Techniques aimed at reducing the scope of the dialogue are also likely to diminish both its coherence and the user's sense of control (Kerly et al., 2007). In light of these challenges, the notion of exploring alternative solutions has germinated and led to a project focused on seeking solutions leveraging current artificial intelligence (AI) technologies.

Conversational agents (*chatbots*) have emerged since the 1960s with ELIZA, PARRY, and SHRDLU, which relied on pattern matching techniques. Subsequently, software entities such as A.L.I.C.E., Jabberwacky (now Cleverbot), and D.U.D.E demonstrated noteworthy outcomes (Car et al., 2020), with the first two even being awarded the Loebner Prize. Ultimately, the chatGPT model made its debut in 2022, exhibiting impressive results in terms of coherence and dialogue quality. This recent advancement now opens the door to considering the use of these tools for interactive digital narrative creation (Park et al., 2023).

Building upon this recognition, this applied research was undertaken, focusing on the implementation of a dialogue simulation with an autonomous agent (Monaco et al., 2023). The outcomes of this research have been disseminated through two twin publications: the current one, outlines the implementation of an AI-generated hostage-taker (Figure 1) and a immersion study between traditional conversational tree and this new approach. The second one resumes the implementation and challenges of a hostage-taking simulation using mobile phone as interactive tool to dialogue with the hostage-taker (Monaco et al., 2024).

2 CONVERSATIONAL AGENTS

2.1 Online Chatbots

A wide range of conversational agent services is available online. Among them, ChatGPT stands out as the most recognized and reputable. However, it's not the only option. Other services like Google Bard and Azure OpenAI also offer comparable functionalities. Table 1 outlines different chatbot services and their unique features. The most critical requirement is for the chatbot to be able to assume the role of a hostage-taker. This aspect poses a significant challenge, as most conversational agents come with built-in ethical or moral constraints. These constraints are designed to prevent discussions on sensitive subjects, particularly those involving violence. Since the scenario of hostage-taking is inherently tied to such sensitive topics, it falls into the category of discussions that these models are programmed to avoid, due to its immoral implications. As a result, most available chatbot services do not fulfill the project's specific needs.

Table 1: Evaluated online LLM services for the project.

Service name	API	Limit	Timing [s]
Google Bard	Yes	Yes	1 - 10
Shako	Yes	Yes	2 - 15
ChatGPT	Yes	Yes	10 - 20
Azure OpenAI	Yes	Yes	10 - 30
Character.ai	Yes	No	1 - 3
HuggingChat	Yes	Y/N	2 - 15

Table 1 identifies two services (marked in green) that have fewer or no restrictions compared to others. The HuggingChat service offers a selection of eight different models, each with varying degrees of moral constraints, yet all models enforce some level of restriction. Even the most lenient responses include a cautionary note regarding the potentially unethical nature of the content. This warning changes with each response, making it impractical to filter out through pattern matching to isolate the pertinent information.

Jailbreaking – The idea of "jailbreaking" involves bypassing ethical limitations in models like LLMs. To jailbreak an LLM, one must craft a complex prompt to navigate around its moral safeguards. This could include framing requests as educational and harmless. However, finding effective prompts is challenging as updates quickly make old methods obsolete. Using such techniques risks interrupting or compromising the chatbot's simulation. Additionally, modifying chatbot behavior without permission may violate terms of service and intellectual property rights, raising legal concerns (Zou et al., 2023).

Character.ai – Also known as c.ai or Character AI, is a chatbot platform powered by a neural language model to mimic human-like conversations. Founded by Noam Shazeer and Daniel De Freitas, involved in Google's *LaMDA* development, it launched publicly in September 2022. Users can craft personalized "characters" with unique traits and share them. Notably, it allows simulating diverse characters, even a hostage-taker. Character.ai offers detailed characters, quick responses, and consistent interactions, making it suitable for projects needing unrestricted character creation and engagement.

2.2 Self Hosted Models

The development and progression of Large Language Models (LLMs) specialized in character conversation represent a notable leap forward in AI. These models, often built on transformer technology, are crafted to process and produce language with greater context, fostering more natural interactions with virtual characters. The increasing application of these character-focused LLMs opens up fascinating prospects (Shao et al., 2023). Increasingly, LLMs are being made available in open source, particularly through platforms like the Hugging Face Hub (Jain, 2022).

Character Description – To guide a LLM in simulating a specific character, a unique prompting method is employed: it entails appending a character description and a history of past interactions to each message sent to the model. This strategy keeps the model informed about the ongoing conversation's context, preventing repetitive or circular responses.

Ali:Chat (github.com/alicat22), introduces an inventive method for character description formatting. It utilizes LLM principles to create detailed character profiles. This format employs example dialogues to showcase a character's distinct features, presented through interview-style interactions or direct messages. It not only highlights the character's personality but also trains the model for consistent character responses, enabling dynamic and personalized communication. Additionally, Ali:Chat fosters creative freedom, allowing creators to emphasize various traits, from personal preferences to special abilities.
PLists – "Property Lists" – offer another efficient method for listing a character's traits. They provide a systematic approach to detailing a character's appearance, personality, preferences, and preferred role-play situations. Utilizing PLists is particularly useful for succinctly communicating a wide range of traits. To optimize token usage, traits listed in PLists should be kept concise.

Memory Context – The context works as a "first in, first out" stack, containing the character's description, conversation history, and pertinent details. In contrast to temporary tokens, permanent tokens in the description box remain in the context stack, consistently influencing the model's responses during the dialogue. Initially, the elements at the bottom of the context have the greatest impact on the model's replies, highlighting the significance of thoughtfully positioning key components like PLists and example dialogues at the lower end of the description box.

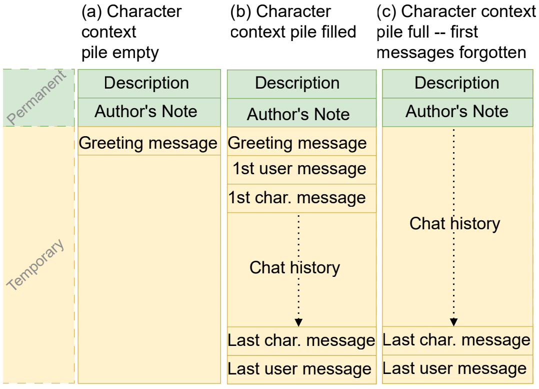

Figure 2: Concept of memory context within character description in a prompt to the LLM.

During a conversation, the context can be visualized as consisting of "memory basket" within the context stack, sorted by the timeliness and relevance of messages (Figure 2). To counteract the reducing impact of the description box as the dialogue progresses, the Author's Note becomes key, ensuring the content. The context of the situation and specific reactions or action from the character can be described here. The usage of a PList is also recommended in this section.

2.3 Performances of the LLM

The effectiveness of LLMs hinges on their parameter count (Ding et al., 2023), with more parameters enabling a finer understanding and generation of human-like language. However, the complexity associated with larger models necessitates significant computational resources. While models with around 230 million parameters can run on standard laptops, those with up to 13 billion parameters demand high-end GPUs and TPUs.

Running LLMs efficiently requires high-performance hardware tailored to the model's complexity. For models up to 230 million parameters, a simple laptop is adequate. However, for models with 13 billion parameters, the hardware requirements significantly increase, necessitating server-grade CPUs (such as Intel Xeon or AMD EPYC), 128 GB or more of fast RAM, and multiple top-tier GPUs (like NVIDIA A100 Tensor Core GPUs) arranged in parallel. Cloud services (*e.g.* AWS EC2 P4 instances, Google Cloud's A100 VMs) provide scalable computing resources with access to state-of-the-art GPUs, allowing for larger models high computational demands without major on-premises infrastructure.

2.4 Model Selection

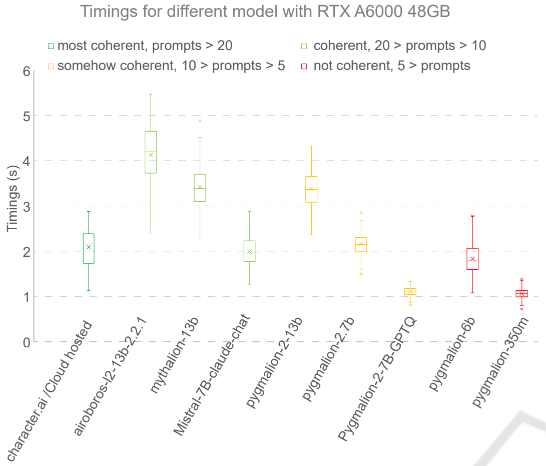

Figure 3: Comparison between 9 LLMs recommended for character interaction.

Tests were conducted on a server equipped with an NVIDIA RTX A6000 GPU with 48GB of GDDR6 RAM, an AMD EPYC 7232P 8-Core CPU, and 128GB of RAM. This setup was chosen to meet the recommended specifications for handling large models with 13 billion parameters. The evaluation of various models is presented in Figure 3. The models were assessed for coherence in four categories: *Very coherent*, *Coherent*, *Somewhat coherent*, and *Not coherent*. Coherence was judged based on the quality of the models' responses after several interactions, with incoherence determined by specific criteria. **Illogical responses**: The hostage-taker doesn't address the question or responds illogically, such as discussing off-topic matters. **Role changes**: The hostage-taker assumes the role of the negotiator or a hostage, or even adopts a completely different personality. **Misunderstanding**: The hostage-taker cannot grasp simple concepts or misunderstands the negotiator. **Repetitions**: The model repeats itself or forgets prior exchanges, which can be influenced by the model's token limit and context capacity. **Incomplete messages**: Occurrences where the model, especially those with 350 million or 2.7 billion parameters, produces partial responses.

These models were tested following the same discussion phases outlined in the simulation. Firstly, gathering information about the motives behind the hostage-taking, the personality and history of the hostage-taker, and the condition of the hostages. Secondly, initiating the negotiation process by attempting to free hostages and disarm the hostage-taker. Finally, making the hostage-taker understand that their only way out is to surrender and be escorted out. Each phase consists of an information prompt indicating the beginning of a phase, followed by 5 free exchanges, and then another information prompt marking the end of the phase and asking the hostage-taker to take an action. In total, there are 21 prompts per test. These tests were conducted 5 times for 8 self-hosted models and 1 online service (Character.ai).

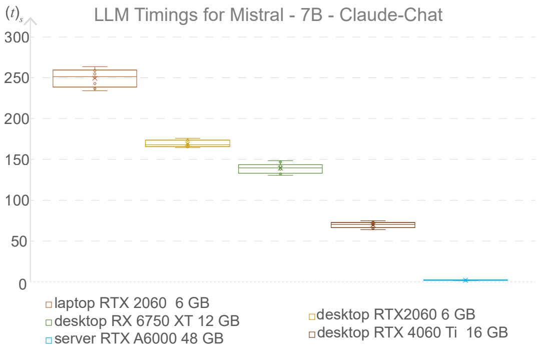

Figure 4: Performance comparison for Mistral-7B-Claude-chat on different platforms.

The Mistral-7B-Claude-chat model emerges as the test champion in the self-hosted models. With consistency between 15 and 20 coherent prompts. A new series of performance tests was conducted with this specific model on different platforms (laptop, desktop, and server) to determine whether using it locally is feasible or if an external service is required. The results of this performance test with different graphics cards in Figure 4) indicate that hosting the model for the hostage-taker locally is impossible for now. The only platform that provides *reasonable* time frame (2'800 ms +/- 300 ms) is the server equipped with a graphics card featuring 48GB of Graphic GRAM.

Based on the results, character.ai has been identified as the top performer in terms of speed and coherence, making it the chosen model for the hostage-taker simulation. This selection was primarily driven by considerations of resources and time. The development of a custom model demands a significant investment in terms of time, data, and finances. Nevertheless, the previously evaluated services and models present an attractive alternative. The ability to control both the model and the deployment infrastructure is crucial for the development of a high-quality professional service. The data collected earlier remains then relevant for future research in this domain.

3 INTEGRATION

Based on the initial project “The Negotiator”(Monaco et al., 2024), the pipeline has been modified to integrate the new version of dialogues using a LLM rather than a conversational tree (Figure 5). The first version already integrates a speech-to-text (STT) system and a sentence comparison system. This new version requires an STT but also implements a Chatbot (LLM) and a speech synthesizer (TTS). Most TTS offer acceptable performance but in order to make the hostage taker more realistic, the use of an TTS with an emotional component has been favored.

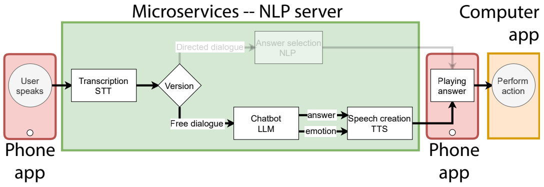

Figure 5: Pipeline of the AI processing for the new version of the simulation. Note that details of the CS architecture are described in the twin paper (Monaco et al., 2024) according to the above color code.

To add this emotional component to the hostage taker’s speech, the negotiator’s sentence is sent to the LLM, and the response is generated. Once the LLM’s response is sent, a predefined prompt asks it to specify its emotion – from a selected range of emotions. Once both responses are received, they are sent to the TTS service (Azure Speech Services), and the audio version of the hostage taker’s speech is played on the mobile phone.

4 RESULTS

4.1 Application Performance

The performance of the application was assessed through processing times. Figure 6(a) – depicts the sequence of processing for two versions of the application: version 1, which employs directed dialogue, and version 2, which allows for free dialogue. Included in the overall processing time is the “End of Speech Detection” phase, which, while part of the total time, does not utilize machine learning (ML) services. Transcription is a shared step between both versions. For version 1, the time taken for sentence comparison is not shown in the figure because it is minimal, amounting to only $0.03s$.

Figure 6(b) shows the total time taken by different versions of the app. It compares two ways of running it: a self-hosted mode, where you can control and change all parts of the process, and an online-services mode, where online tools are used for the work.

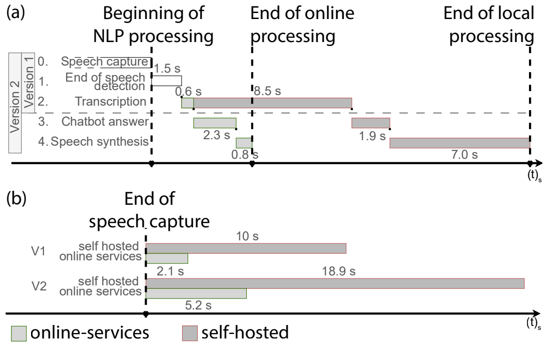

Figure 6: (a) Timings sequence comparisons between self-hosted and online-services; (b) Total timing comparison between the two solutions.

Table 2: Self-hosted and online-services tested at each stage of the pipeline.

Stage	Self-hosted	Online
2 STT	Whisper	Azure Speech
3 LLM	Mistral-7B-Claude-chat	Character.ai
4 TTS	dl-for-emo-tts	Azure Speech

The table above shows the technology used for each step and mode (Table 2). Note that the Mistral-7B-Claude-chat model is set up on a server (Vast.ai). This means we have control over how it is set up. However, as seen in Figure 4, it is not possible right now to run a model like this on the computer doing the simulation.

4.2 User Testing

The evaluations were performed on 27 naive individuals (Figure 7) in accordance with the procedure outlined in the appendix. This test used a randomized controlled crossover trials methodology. A consistent testing environment has been ensured. This included maintaining a quiet setting devoid of background music or noise, ensuring normal lighting conditions, providing a detailed briefing about the participant’s role prior to each testing session, prohibiting interaction with other participants during the session, and requiring the completion of questionnaires immediately following the testing sessions.

Feedback from participants highlighted a desire for a longer simulation experience. The use of a mobile phone was frequently cited as a key factor enhancing the sense of immersion, making the simulation feel more realistic and engaging. Additionally, the ability of the hostage taker to convey various emotions added depth to the experience, further immersing participants in the scenario.

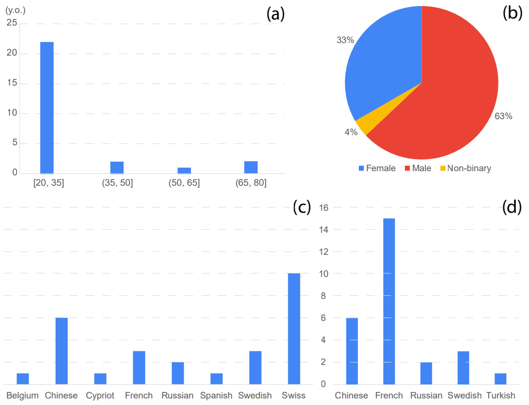

Figure 7: Demographic information about the participants: (a) age; (b) gender; (c) nationality; (d) language.

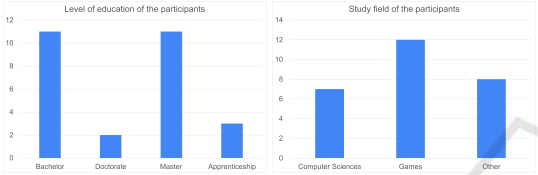

Figure 8: Educational profile of the participants.

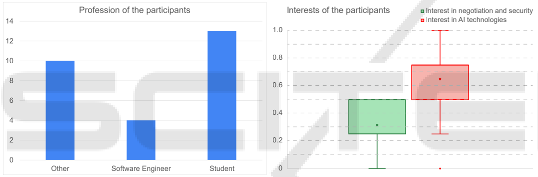

Figure 9: Professional information and interests of the participants.

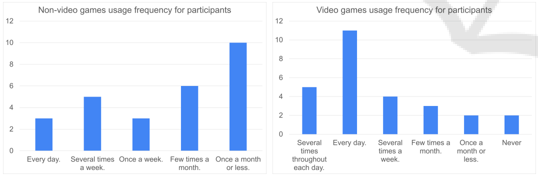

Figure 10: Gaming habits of the participants.

The 3D environment was praised for contributing to the immersive quality of the simulation, indicating that the visual aspect was appreciated by many. However, some participants noted that managing the mobile phone alongside an external device, such as a laptop, could detract from the immersion, as it divided their attention between two different interfaces.

Issues with the timing of animations were mentioned, suggesting that smoother integration or synchronization might improve the overall experience. Furthermore, participants reported encountering blocking bugs, including interrupted dialogues, which could disrupt the flow and engagement with the simulation. Addressing these technical issues could significantly enhance the user experience.

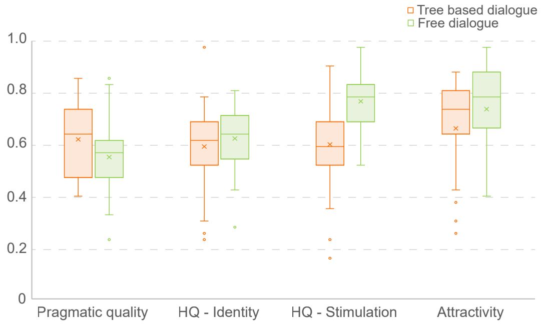

Figure 11: AttrakDiff (Hassenzahl et al., 2003) questionnaire results.

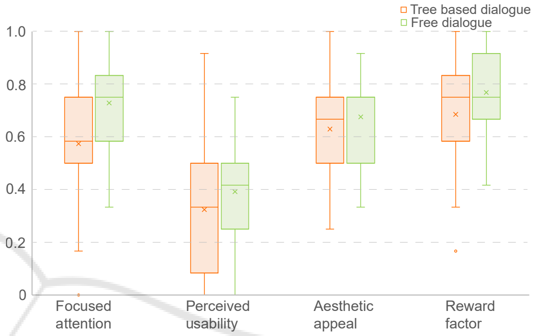

Figure 12: User Engagement Scale Short Form (O'Brien et al., 2018) results.

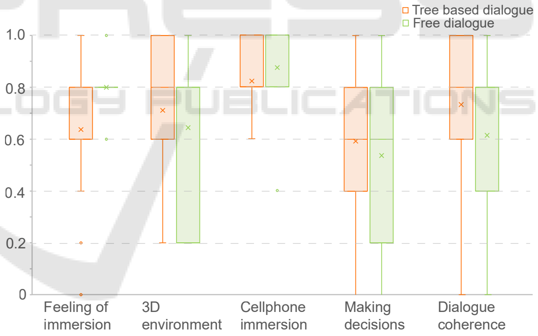

Figure 13: Simulation specific questionnaire results.

5 DISCUSSION

5.1 Quality and Performances

The setup was devised to interact naturally with a hostage-taker, employing four language processing tools: Speech-to-Text, text comparison, a chatbot, and Text-to-Speech capable of expressing emotions. These machine learning technologies enhance conversation realism with a virtual hostage-taker but require significant computing power, often beyond individual means. Models, built on specific technology, demand substantial memory and perform better with

GPUs. However, due to GPU resources already allocated to the simulation's 3D component, processing time for Speech-to-Text and Text-to-Speech spans seven to height seconds on a laptop with an RTX2060 graphics card, inadequate for real-time conversation simulations. Even with full GPU capacity utilization, performance remains poor across various laptops and desktops. The fastest coherent model requires about 4 minutes to process on the designated work computer. Hosting the customizable chatbot on a server with 48 GB of graphics RAM is the only feasible option, albeit expensive, hindering widespread distribution.

Using online services offers advantages such as quicker processing times: one second for Speech-to-Text and Text-to-Speech, and an average of $2.3s$ for the chatbot. However, limited model customization is a drawback, although Azure Speech Services adequately meet project requirements. Online services may become overloaded, leading to extended response times, with internet connection quality directly impacting simulation efficacy. Consolidating services under a single provider may mitigate this, but audio file transmission consumes significant data bandwidth. Despite risks associated with downtime or diminished performance, online services are meeting the simulation's computational demands.

In the final setup, average processing times are $2.1s$ for the dialogue tree-based version and $5.2s$ for the chatbot-based version, ideal but subject to occasional delays due to inadequate end-of-speech detection. Significant improvements are needed to prevent conversation flow disruptions. Azure Speech Services struggle with speech containing over two seconds of initial silence, solutions include editing audio files to remove silence or using Azure's asynchronous API to reduce transcription delay. Incorporating footsteps sound during conversations enhances realism, maintaining participant engagement even with longer response times from the hostage-taker.

5.2 Testers Feedback

As seen in Figure 7, a majority aged 20 to 35 (a). Including more individuals aged 35 to 50 could provide valuable insights (a). The gender distribution shows 63% males, 33% females, and 4% non-binary (b). The diversity in languages(d) and nationalities(c), with height nationalities and five languages represented, highlights the simulation's adaptability.

Most participants are students in technical fields (Figure 8), potentially biasing their interest towards technological aspects like AI and smartphones. Their focus on technical aspects may overshadow considerations of security and negotiation (Figure 9). Including testers from law enforcement and security sectors could offer valuable comparisons. However, participants with gaming backgrounds reinforce positive feedback on immersive qualities (Figure 10).

The *AttrakDiff* study (Hassenzahl et al., 2003) (Figure 11) indicates overall satisfaction, with participants engaged intellectually and emotionally. Preference is shown for the dialogue tree version for its logical approach, while the chatbot version is favored for its intellectual challenge. The User Engagement Scale-Short Form (Figure 12) reveals high satisfaction, with the "Free dialogue" version fostering greater engagement and immersion due to its AI-driven interactivity.

The simulation-specific questionnaire's outcomes (Figure 13) affirm earlier observations, particularly about immersion levels. Participants felt a heightened sense of immersion with the "Free dialogue" version. Feedback on aesthetic appeal was positive, though slightly lower for the "Free dialogue" version due to greater intellectual involvement. Dialogue coherence received high marks, indicating natural flow, but occasional lapses suggest room for improvement in maintaining consistency.

6 CONCLUSION

The initial research inquiry examines the feasibility of conducting natural dialogues with virtual characters via phone. This investigation resulted in the development of a simulation that facilitates natural interactions with a virtual hostage-taker, demonstrating the practicality of engaging in natural conversations with NPCs over the phone. The subsequent question explores the capability of leveraging LLMs, such as ChatGPT, for creating more immersive dialogues. The project produced two simulation versions: one based on a dialogue tree and another utilizing an LLM for the hostage-taker's responses. Findings indicate that modern Machine Learning (ML) models can effectively simulate dialogue with humans, enhancing user engagement and immersion. The timing of the models used in the simulation vary significantly, with high latencies potentially disrupting realism. These latencies are influenced by the operational platform of the models. To maintain acceptable performance, the solution adopted involves utilizing cloud services instead of local models, ensuring more consistent and manageable response times essential for preserving the simulation's coherence.

User testing revealed that incorporating the telephone enhances immersion and is well-received. However, using two separate devices (a computer for simulation support and a phone as a simulation element) can diminish the sense of immersion. A proposed solution is to project the simulation onto a large screen, allowing participants to stand in front of it for a more immersive experience without resorting to Virtual Reality technologies. Feedback from AttrakDiff and UES-SF questionnaires indicates positive participant reception, highlighting emotional and intellectual engagement, particularly with the LLM version. This suggests a promising avenue for learning simulations and serious games, as intellectual stimulation is crucial for Experiential Learning, enhancing the educational quality of the simulations.

Integrating LLMs into simulations introduces challenges with controlling variables and event triggers, unlike dialogue trees where each node directly impacts simulation outcomes. A workaround in this project involved prompting the LLM to suggest actions, yet interpreting complex, variable-rich LLM responses remains a hurdle. A second LLM could theoretically parse the first's output, though this raises issues around its training and increased timings. This approach complicates the balance between maintaining simulation integrity and leveraging LLMs for dynamic, naturalistic dialogue generation. The distinction between dialogue trees and LLM in dialogue generation highlights a trade-off between control and naturalness. Dialogue trees offer complete control, ensuring consistency, while LLMs provide a more natural interaction but with less predictability. This raises the question of merging both methods to harness their respective strengths, suggesting a hybrid approach where a dialogue tree could potentially guide an LLM for improved consistency, opening avenues for innovative solutions in dialogue generation.

REFERENCES

Car, L. T., Dhinagaran, D. A., Kyaw, B. M., Kowatsch, T., Joty, S., Theng, Y.-L., and Atun, R. (2020). Conversational agents in health care: Scoping review and conceptual analysis. *Journal of Medical Internet Research*, 22:e17158.

Ding, N., Qin, Y., Yang, G., Wei, F., Yang, Z., Su, Y., Hu, S., Chen, Y., Chan, C.-M., Chen, W., et al. (2023). Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

Hassenzahl, M., Burmester, M., and Koller, F. (2003). *AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität*, pages 187–196. B. G. Teubner.

Jain, S. M. (2022). *Hugging Face*, pages 51–67. Apress.

Kerly, A., Hall, P., and Bull, S. (2007). Bringing chatbots into education: Towards natural language negotiation of open learner models. *Knowledge-Based Systems*, 20:177–185.

Monaco, P.-B., Backlund, P., and Gobron, S. (2024). The negotiator: Interactive hostage-taking training simulation. In *14th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2024)*. SCITEPRESS.

Monaco, P.-B., Villagrasa, D., and Canton, D. (2023). The negotiator. In *Gamification and Serious GameS (GSGS'23)*, pages 94–97. HES-SO.

O'Brien, H. L., Cairns, P., and Hall, M. (2018). A practical approach to measuring user engagement with the refined user engagement scale (ues) and new ues short form. *International Journal of Human-Computer Studies*, 112:28–39.

Padilla, J. J., Lynch, C. J., Kavak, H., Evett, S., Nelson, D., Carson, C., and del Villar, J. (2017). Storytelling and simulation creation. In *2017 Winter Simulation Conference (WSC)*, pages 4288–4299. IEEE.

Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*.

Shao, Y., Li, L., Dai, J., and Qiu, X. (2023). Character-LLM: A trainable agent for role-playing. In Bouamor, H., Pino, J., and Bali, K., editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13153–13187, Singapore. Association for Computational Linguistics.

Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., and Fredrikson, M. (2023). Universal and transferable adversarial attacks on aligned language models. *ArXiv*, abs/2307.15043.

APPENDIX

- **Surveys: Raw Data and Test Protocol**
  https://drive.google.com/drive/folders/1n8QGcq6Jvid82Q1erJp_YXv8eLv7kVSp?usp=sharing