NLP Project Proposal

Hengxu Wu-2024011308, Kairui Li-2024011307, Mo Shen-2024011341, Yangyi Xiong-2024011319
Tsinghua University
Beijing, China
wuhx24@mails.tsinghua.edu.cn

Abstract

Our project aims to build a real-time, multimodal VTuber system powered by a Large Language Model (LLM). It is designed to interact with users through natural conversation, generating context-aware and engaging responses to deliver an immersive experience in virtual environments.

1 Introduction

Existing LLM-based vtuber system, like Open-LLM-VTuber, has made significant strides in creating interactive virtual avatars. However, these systems have also multiple drawbacks, such as lack of long-term memory and support for function calling and multi-agent collaboration. Our proposed system aims to address these limitations by integrating long-term memory capabilities, enabling function calling, and facilitating multi-agent collaboration. By enhancing these aspects, our vtuber system will provide a more immersive and dynamic user experience.

2 Implementation Plan

We have divided our project into several key components, each focusing on a specific aspect of the vtuber system: the structure graph is shown in Figure 1.

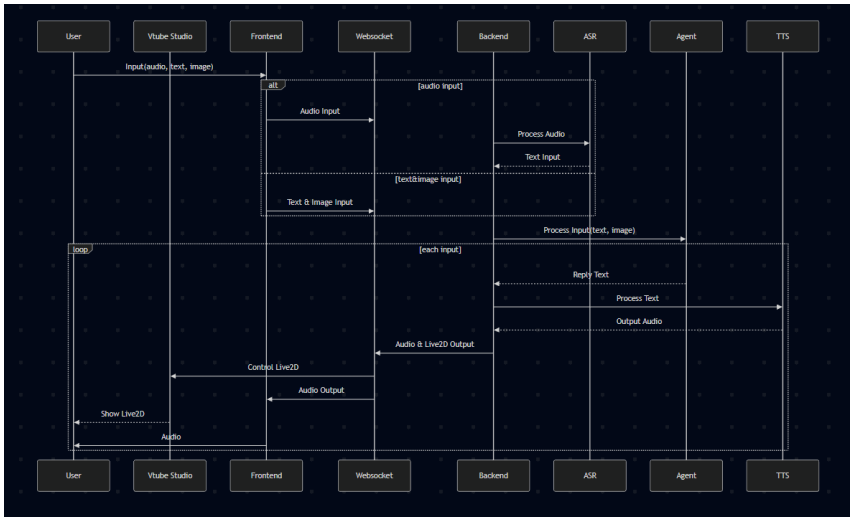

Figure 1: The structure graph of our proposed vtuber system.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Based on the framework illustrated in the diagram, user inputs, including audio, text, and visual data are initially sent to the Frontend. The audio input is first processed by the Automatic Speech Recognition (ASR) module to convert it into text. Subsequently, both the transcribed text and the visual data are processed collectively by the Backend. At the core of the system, the Agent module takes these processed inputs and generates a structured response, which includes both action control commands and reply text. This response text is then parsed by the Backend to separate the action commands from the reply content. The reply text is forwarded to the Text-to-Speech (TTS) module to synthesize the corresponding audio response. Finally, the generated audio and the action control commands are transmitted to the Frontend and VTube Studio, respectively, where they are rendered as the final output presented to the user.

3 Paper Research

Based on the structure above, we have searched for overview papers on each module to understand the current state of the art and identify suitable tools for implementation. Specifically, we focused on the areas of ASR[**?**], TTS[**?**] and long-term memory (LTM)[**?**][**?**] for the Agent module.

4 Time Schedule

We plan to complete the project within a span of 6 weeks, following the timeline below:

- **Week 1-2:** Research and select appropriate LLMs and tools for integration. Set up the development environment and initial project structure.
- **Week 3-4:** Implement the core functionalities, including long-term memory integration, function calling, and multi-agent collaboration.
- **Week 5:** Test the system extensively, identify bugs, and optimize performance. Gather feedback from initial users.
- **Week 6:** Finalize documentation, prepare a demonstration of the system, and submit the project for evaluation.

5 Current Progress

Now we have integrated the Vtube Studio with our frontend, with correct voice and action control.

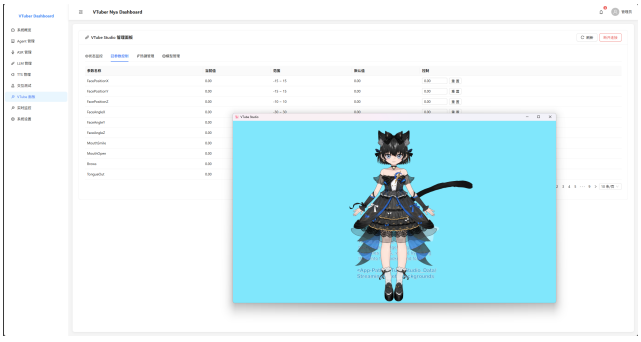

Figure 2: The frontend interface of our vtuber system along with the vtuber avatar.

Figure 3: The vtuber avatar outputs in Vtube Studio.