

# GPT Models Meet Robotic Applications: Co-Speech Gesturing Chat System

Naoki Wake<sup>1</sup>, Atsushi Kanehira<sup>1</sup>, Kazuhiro Sasabuchi<sup>1</sup>, Jun Takamatsu<sup>1</sup>, and Katsushi Ikeuchi<sup>1</sup>

## Abstract

This technical paper introduces a chatting robot system that utilizes recent advancements in large-scale language models (LLMs) such as GPT-3 and ChatGPT (Fig.1). The system is integrated with a co-speech gesture generation system, which selects appropriate gestures based on the conceptual meaning of speech. Our motivation is to explore ways of utilizing the recent progress in LLMs for practical robotic applications, which benefits the development of both chatbots and LLMs. Specifically, it enables the development of highly responsive chatbot systems by leveraging LLMs and adds visual effects to the user interface of LLMs as an additional value. The source code for the system is available on GitHub for our in-house robot and GitHub for Toyota HSR.

## I. INTRODUCTION

In recent years, large-scale language models (LLMs) such as GPT-3 [1] and ChatGPT [2] have achieved remarkable success in natural language processing tasks. Meanwhile, there is a growing interest in applying LLMs to robotic applications (task planning, for example, [3]), as they have the potential to enhance human-robot interaction with its high language processing capabilities.

In this paper, we present a co-speech gesturing chat system that combines the GPT-3 model with a gesture engine developed by our team [4], [5]. The gesture engine analyzes the text of the robot's speech and selects an appropriate gesture from a library of gestures associated with the conceptual meaning of the speech, called as gesture library [6]. By integrating LLMs as the backend, we aim to provide users with a highly responsive chat system that can handle a wider range of conversational topics and contexts.

We believe that LLMs will contribute significantly to the development of practical robotic systems. Taking chatting robots as an example, developers can develop highly responsive systems with simple prompt engineering. Another interesting topic will be how adding visual effects to the user interface of LLMs, which are mostly text-based interactions, will affect the usability and conversational content. We have implemented the chat system into two robots, our in-house MSRAbot and Toyota HSR. The source code for the system is available on GitHub for our in-house robot and GitHub for Toyota HSR.

## II. PIPELINE

The overview of the pipeline is shown in Fig.2. The user sends a query to the robot system via text or microphone input. Microphone input is noise-suppressed to prevent the robot's ego noise from interfering with recognition [7], [8] and then converted to text using a third-party text-to-speech technology [9]. The robot system then generates a prompt for the GPT-3/ChatGPT model based on this input.

### A. Chatting engine

1) *GPT-3 model as backend*: As GPT-3 is not specialized for chat and is designed for text completion, the prompt needs to be carefully crafted to achieve chat-like responses. To do this, we save the conversation history between the user and the robot system and design the following prompt:

“You are an excellent chat bot. Please respond to the current message accurately, taking into account your knowledge and our previous conversations. Previous conversations: *history* Current message: *message*”

Here, *history* is assigned the conversation exchange separated by line breaks, and *message* is assigned the new input from the user. We used the Azure Open AI's davinci model as the specific model. The example of the conversation is shown in Fig.3.

2) *ChatGPT model as backend*: We used the Azure Open AI's gpt-3.5-turbo model as the specific model. Since the ChatGPT model can receive conversation history, we simply described roles in the prompts, without embedding the conversation history. The following is an example of a prompt we prepared:

“You are an excellent chat bot, named MSRAbot. You are embodied with a small robot, which makes lively gestures in response to your speech. Please keep conversations with the user by responding with short English phrases. The response can be composed of several sentences, but every sentence should be definitely short and less than 12 words. Answer in English in any situation.”

<sup>1</sup>Applied Robotics Research, Microsoft, Redmond, WA 98052, USA naoki.wake@microsoft.com

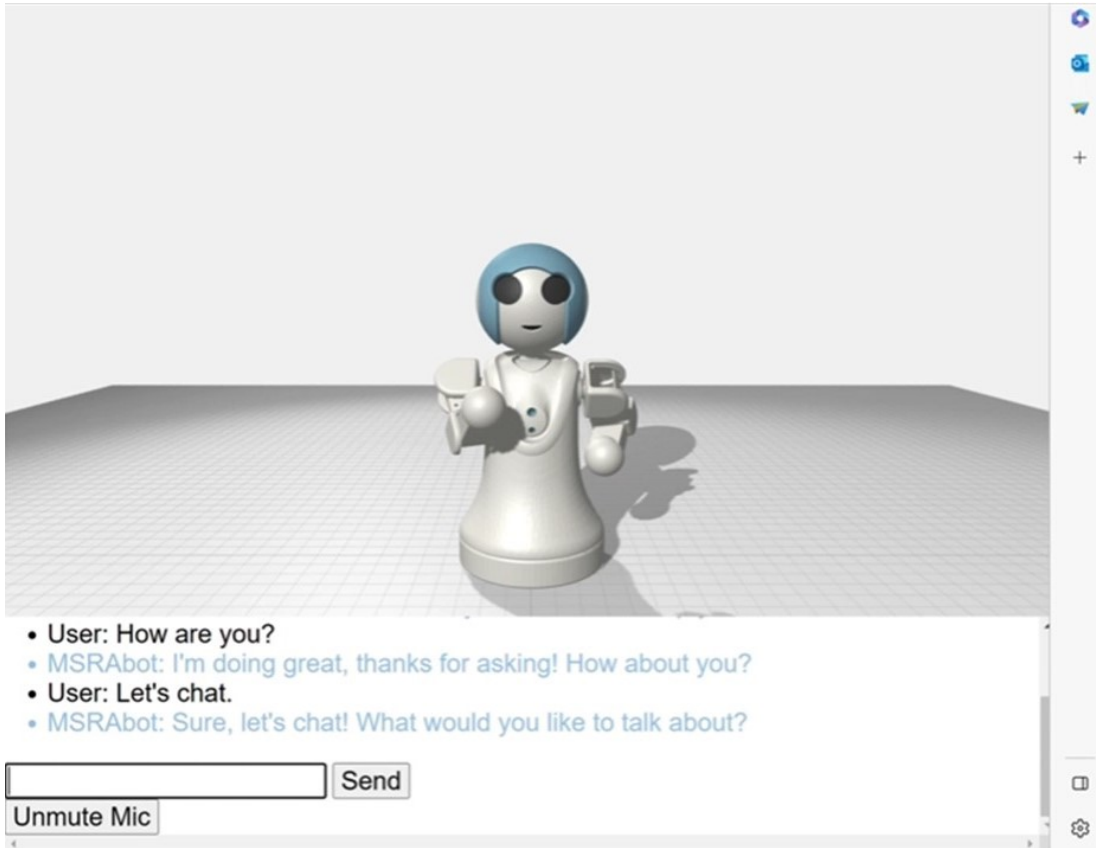


Fig. 1. Our robotic gesture engine and DIY robot, MSRABot, are integrated with a GPT-based chat system.

### B. Gesture engine

The response from GPT-3/ChatGPT is passed on to two modules: a speech generator, which converts the text into speech using a third-party text-to-speech technology [9], while the gesture engine selects a concept from the text. For concept estimation, we preliminarily analyzed a collection of everyday English conversation phrases and defined dozens of concepts that are commonly represented in conversations [6]. Based on the concept labels for the phrases, we trained the estimation model using Azure Language Understanding [9].

The gesture generator then produces gestures based on the estimated concept. When several gestures are associated with the concept, we randomly select one. The length of the gesture was modified to match the length of synthesized speech. Finally, the generated speech and co-speech gestures are presented to the user as audio-visual feedback.

### C. Open sources

Note that we store the information of gesture motions using Labanotation, a notation for describing human dance. Labanotation is an intermediate representation of human motions that contains enough information to reproduce them. Thus, by implementing a Labanotation decoder for each robot, this pipeline can be scaled to arbitrary robots [10]. Currently, we have prepared the decoder for our in-house robot, MARobot, and Toyota HSR, and we have open-sourced the code on GitHub for our in-house robot and GitHub for Toyota HSR. MSRABot was originally designed as a platform for human-robot interaction research, and we have developed and open-sourced a DIYKit. This DIYKit includes 3D models of the parts and step-by-step assembly instructions, enabling users to build the robot's hardware using commercially available items. The software needed to operate the robot is also available on the same page.

## III. DISCUSSIONS AND CONCLUSION

LLMs like GPT-3/ChatGPT have shown remarkable success in natural language processing tasks, leading to growing interest in applying them to robotic applications. However, connecting a robot with LLMs poses risks such as bias, inappropriate responses, or vulnerability to attacks. Solutions to these problems are in the process of development. To minimize those risks, it is crucial to carefully monitor and control the robot's output, utilize robust security measures, and provide proper ethical guidelines.

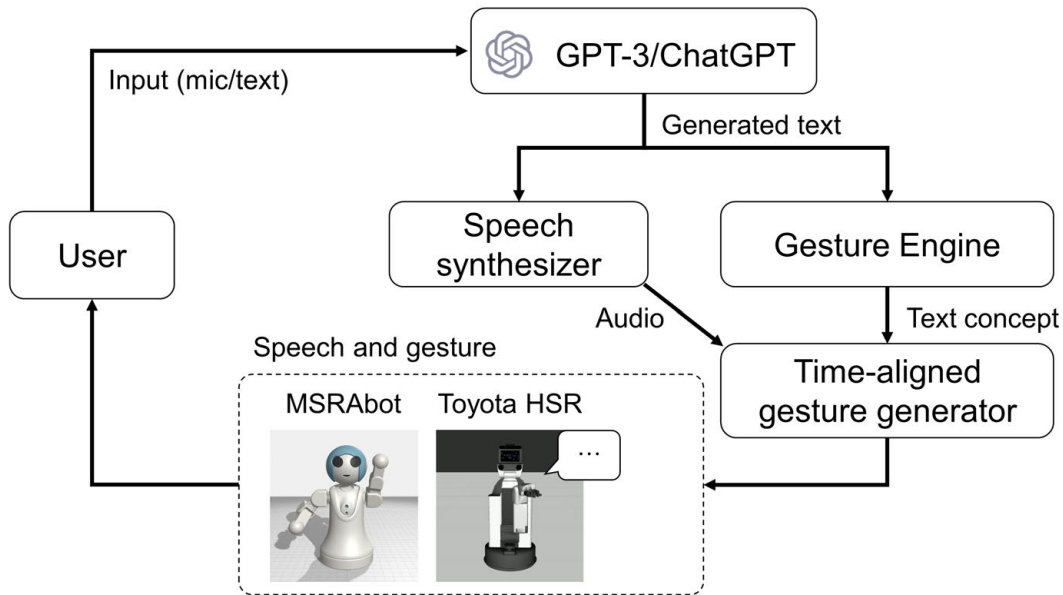


Fig. 2. The pipeline of the co-speech gesture generation system.

## Chat demo

[User]: Hi

[Robot]: there!Hi there! It's nice to meet you. How can I help you?

[User]: Nothing. Let's chat. What is your name?

[Robot]: My name is Robot. It's nice to meet you! What would you like to chat about?

[User]: Let's talk about baseball games.

[Robot]: Sure, let's talk about baseball games! What would you like to know?

[User]: Who is the popular Japanese baseball player in MLB?

[Robot]: The most popular Japanese baseball player in MLB is Shohei Ohtani, who currently plays for the Los Angeles Angels. He is a two-way player, meaning he can both pitch and hit.

[User]: What kind of awards did he receive in 2022?

[Robot]: Shohei Ohtani received the American League Comeback Player of the Year Award in 2022. He was also selected to his first All-Star Game that year.

[User]: Great. Let's change the topic. Who is the famous Japanese football player in Europe?

[Robot]: The most famous Japanese football player in Europe is Takefusa Kubo, who currently plays for Real Madrid. He is a winger and has been praised for his technical ability and speed. He previously played for FC Tokyo in Japan before moving to Europe.

[User]: Do you know other football players in FC Tokyo?

[Robot]: Yes, I do know of other football players in FC Tokyo. Some of the notable players include Shuto Abe, Keigo Higashi, and Yojiro Takahagi.

Fig. 3. Example of the conversation history between a human user and the chat system.

In conclusion, this paper introduced an LLM-empowered chatting robot system for achieving a natural and intuitive chatting experience, while also providing synchronized co-speech gestures. We believe that LLMs will facilitate the development of practical robotic applications, provided that we pay close attention to the limitations of those models.

## REFERENCES

- [1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [2] OpenAI, "Chatgpt." <https://openai.com/blog/chatgpt>.
- [3] N. Wake, A. Kanehira, K. Sasabuchi, J. Takamatsu, and K. Ikeuchi, "Chatgpt empowered long-step robot control in various environments: A case application," *arXiv preprint arXiv:2304.03893*, 2023.
- [4] N. Wake, M. Sato, K. Sasabuchi, M. Nakamura, and K. Ikeuchi, "Labeling the phrases of a conversational agent with a unique personalized vocabulary," in *2022 IEEE/SICE International Symposium on System Integration (SII)*, pp. 856–863, 2022.
- [5] H. Teshima, N. Wake, D. Thomas, Y. Nakashima, D. Baumert, H. Kawasaki, and K. Ikeuchi, "Integration of gesture generation system using gesture library with diy robot design kit," in *2022 IEEE/SICE International Symposium on System Integration (SII)*, pp. 361–366, IEEE, 2022.
- [6] K. Ikeuchi, D. Baumert, S. Kudoh, and M. Takizawa, "Design of conversational humanoid robot based on hardware independent gesture generation," *arXiv preprint arXiv:1905.08702*, 2019.

- [7] N. Wake, M. Fukumoto, H. Takahashi, and K. Ikeuchi, "Enhancing listening capability of humanoid robot by reduction of stationary ego-noise," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 14, no. 12, pp. 1815–1822, 2019.
- [8] J. Jaroslavceva, N. Wake, K. Sasabuchi, and K. Ikeuchi, "Robot ego-noise suppression with labanotation-template subtraction," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 17, no. 3, pp. 407–415, 2022.
- [9] "Microsoft azure." <https://azure.microsoft.com/>, (accessed March. 6, 2023).
- [10] K. Ikeuchi, Z. Ma, Z. Yan, S. Kudoh, and M. Nakamura, "Describing upper-body motions based on labanotation for learning-from-observation robots," *IJCV*, vol. 126, no. 12, pp. 1415–1429, 2018.