

Name: 許安

Class: 百川16

Student ID: 112950004

Assignment 2: Conversation AI

For this project, I created an interactive 3D avatar in Unity that responds to voice input using OpenAI's GPT-based Realtime API. The avatar is capable of interpreting user speech, expressing emotions through animations, and interacting with scene objects in real time.

What I Did:

1. Integrated GPT via Realtime WebSocket API

- Used OpenAI's Realtime API to stream speech recognition and generate live avatar responses.

2. Emotion Classification

- Sent GPT's responses through an emotion classification prompt using the Chat Completions API.
- Mapped the detected emotion (e.g., happy, sad, angry) to animation triggers on the avatar. A total of 7 possible emotion classes were introduced, including happy, sad, angry, excited, thankful, congratulating, or neutral. All avatar animations were downloaded from Mixamo.

3. Environment & Object Interaction

- Enabled the avatar to react to subjects in speech, such as:
 - "snow" → activates snowfall (spawn particle system)
 - "box" → shows or moves a 3D box, and change the color of the box
- Used Unity's SetActive and transform.position to control scene objects based on input.

4. Confetti & Feedback Effects

- When congratulatory language is detected, a confetti effect (particle system) is triggered for visual celebration.

5. Natural Dialogue Personality

- Gave the avatar a helpful, friendly personality through GPT system prompts, making it behave like a bubbly, curious, and helpful character. This personality design helps make interactions feel more alive and playful.

6. Customized avatar

- Changed the avatar model used to customized avatar. Created an avatar in the Meta Horizon App, then published the Apps to enable linking.

How I Did It:

- Realtime Speech Input:
Used AudioRecorder and RealtimeAPIWrapper.cs to capture and send microphone input directly to GPT.
- Response Processing Pipeline:
Responses from GPT were stored in a queue, then:
 - Displayed in the console
 - Emotionally classified
 - Used to trigger Unity animations and world interactions
- Emotion Mapping:
Used an if-else chain to map classified moods to animation triggers (animator.SetTrigger("Happy"), etc.).
- Keyword Detection:
Parsed GPT replies for specific keywords like "box" or "snow" to trigger corresponding scene actions.
- Scene Setup:
Pre-placed objects (e.g., boxObject) were toggled or moved dynamically using simple Unity API calls.

Example Demo Lines:

Here are example lines I can say to the avatar, and what happens:

- "What's your favorite thing about today?" → Avatar replies happily (happy animation)
- "Guess what? We're launching a rocket tomorrow!" → Avatar gets excited (excited animation)
- "It's freezing today!" → Snowfall effect activates
- "Can you show me the box?" → Box appears in the scene
- "Can you move the box over there?" → Box moves position
- "Can you change the color of the box?" → Box changes color to red

Video report link: <https://youtu.be/8ab1Gbbmeo4>

Project cloud link:

<https://drive.google.com/file/d/15FoeDzZx8x25C-fzFil9TTNrdUjpe-91/view?usp=sharing>