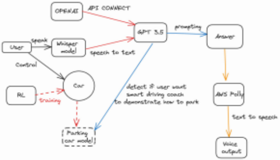
Smart Driving Coach based on GPT-3.5

Student: Po-Hung, Sung　　　　Advisor: Pf. Wei-Liang, Lin

Overview

Traditional driving training methods have many limitations, such as* limited training time, inconsistent teaching quality, and the nervousness of learners during practical operations. *To address these issues, we have developed an intelligent driving coach system using the existing language model GPT-3.5.* By incorporating Unity and reinforcement learning, *the intelligent driving coach can learn how to perform tasks such as parallel parking and demonstrate them in real-time.

Research Method



- Prompting: Use prompting to restrict the dialogue content of the smart driving coach. If the user asks questions unrelated to the lesson, they will be reminded to focus on the class. Prompting is also used to create a learning plan and set goals.
- Voice Interaction Interface: **Whisper converts the learner's verbal commands or questions into text with high accuracy**, even in noisy environments, which is crucial for driving instruction. This text is then passed to the GPT-3.5 model, which interprets the learner's input and generates appropriate text responses. **These responses are converted into natural and clear speech using AWS Polly**, providing feedback to the learner in a voice close to human speech.
- Implementing Reinforcement Learning with ML-Agents in Unity: We use **ML-Agents in Unity to implement reinforcement learning**, enhancing the intelligent driving coach's learning and demonstration capabilities. We **apply the PPO (Proximal Policy Optimization) algorithm in reinforcement learning** to ensure training stability and efficiency. By simulating driving scenarios in 20 virtual environments within Unity, the intelligent driving coach can continuously practice and improve its parking skills in a safe virtual world.
- PPO (Proximal Policy Optimization): PPO is a reinforcement learning algorithm used to train agents. **The PPO algorithm limits the extent of each policy update to ensure that the policy changes are not too drastic, thereby maintaining stability in the learning process.** This helps the agent to more effectively explore and exploit different actions during training, improving its performance across various tasks.

[1] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov, "Proximal Policy Optimization Algorithms" ,Aug 2017

Result

The voice interaction interface

Parking scenarios