Dialogue-system-2-2024 Lab2

For this lab, I have made two dm files for the two parts:

* dm.ts
* dm2.ts

For part 2 exploration, I chose the option 2: rule-based dialogue and implemented it in ”dm2.ts”.

I tried to implement a ”Burger restaurant ordering” scenario with a preset ”menu”.

The dialogue system first prompts to ask ”What can I help you?”, and then listens to the user. The user could make an order or request to know the menu (if neither, prompt again). The system will reply accordingly and then jump to either ”Prompt” again or ”Confirm” the order.

I designed two actors, one for fetching **an NLU response of the order (including an intent and entities)** in the JSON format from llama, another one for fetching the **attitude (either positive or negative)** of the reply for confirmation in the string format.

The dialogue machine now has the basic flow of making an order and then confirm an order, but I will need more time to figure out how to store the ”amount” of each food or drink item and deal with the increments and decrements in the follow-up orders.

Also, I’ve noticed that for an NLU task like simple ordering or attitude confirming, LLM like llama might not be the best tool because there is an extra process from ”chat text” to ”useful data”.

Azure NLU seems to be more direct and convenient to use to include multiple intents and entities.