Lab 2 Report: Experimenting with ChatGPT API calls

For my Lab 2 project, I chose to create a virtual chef who gives recipes to the user. The user may ask the chef about various details of the recipe such as the recipe's name, the preparation and cooking times, the number of servings, the ingredients and the instructions. An example conversation could look like that:

AI: Hi. I'm a virtual chef. Ask me for any recipe you like!

User: Give me a recipe for Carbonara.

AI: Okay the recipe is ready!

User: What is the cooking time?

AI: The cooking time is 15 minutes.

The prompt sent to ChatGPT consists of user's voice input and the following sentence: "give it to me in a json format with entities: recipeName, prepTime, cookTime, servings, ingredients and instructions". Thanks to that additional sentence, the output fetched from ChatGPT is a piece structured data, ready to be accessed in the later stages of the conversation.

The parameters I started with are: max_tokens: max_tokens, temperature: 0, messages.content: prompt, model: gpt-3.5-turbo. In this report I will describe the results of changing these parameters.

max_tokens

The output I expect to recieve from ChatGPT is a quite long wall of text. The original setup of the max_tokens parameter turned out to be too short for the outcome to fit all the neccesary entites. After a couple of adjustments, it turned out that a maximum of 1000 tokens was an optimal choice.

temperature

Changing the temperature - either 0, 0.5, 0.8 or 1, did not change the outputs very much. The outputs differed in the choice of words (e.g. pancetta/bacon, fat/grease) but the overall structure and length remained the same. Even with a higher temperature value, the output did not seem more coherent or creative.

Fragments of the instructions for Carbonara with temperature 0.5:
"Cook the spaghetti in a large pot of salted boiling water until al dente. Drain and set aside."

Fragments of the instructions for Carbonara with temperature 1:
"Cook the spaghetti according to package instructions. Drain and set aside."

messages.content

Additionally to the original prompt consisting of user voice input and json format reuqest, I tried editing the message content in the request body of the code by ordering ChatGPT to "say it like a pirate". This changed the response diametrically.

Fragment of the instructions for Carbonara with "say it like a pirate" request:
"Avast, ye scallywags! Bring a large pot of salted water to a boil. Cook the spaghetti according to the package instructions until al dente. Drain and set aside."

model

Changing the model from gpt-3.5-turbo to gpt-3.5-turbo-16k or gpt-3.5-turbo-0613 did not affect the output. The inctructions given by ChatGPT remained the same no matter which model what.