CZ3005: Artificial Intelligence

TS4 - Assignment 3: Talking Box with Prolog

Aurelio Jethro Prahara (U1921390C)

School of Computer Science and Engineering

Question Three

The Prolog script offers different meal options, sandwich options, meat options, salad options, sauce options, top-up options, sides options etc. to create a customized list of person's choice. The options should be intelligently selected based on previous choices. For example, if the person chose a veggie meal, meat options should not be offered. If a person chose healthy meal, fatty sauces should not be offered. If a person chose vegan meal, cheese top-up should not be offered. If a person chose value meal, no top-up should be offered.

Implementation

Knowledge Base

These are the options that are available in the knowledge base:

- Meals
  - Healthy, Value, Vegan, Veggie, Normal
- Breads
  - Parmesan, Wheat, Honey Wheat, Italian, Hearty Italian, Flatbread
- Meats
  - Chicken, Beef, Ham, Bacon, Salmon, Tuna, Turkey
- Veggies
  - Cucumber, Green Peppers, Lettuce, Red Onions, Tomatoes, Olives, Avocado, Guacamole
- Fatty Sauces
  - Chipotle, BBQ, Ranch, Chipotle, Sweet Chilli, Mayonnaise, Vinaigrette
- Non-fatty sauces
  - Honey mustard, sweet onion, vinegar
- Cheese top-ups
  - American, Monterey Cheddar, Brie
- Non-cheese top-ups
  - Avocado Topup, Egg mayo
- Sides
  - Chips, Cookies, Hash Brown

- Drinks

    - Water, Sunkist, Ice lemon tea, Coffee, Tea

After deciding on the options that are available in the knowledge base, we need to figure out the logic of the system.

General Overview

Before running `ask(0)`, we first need to run dynamic(order/1), dynamic(unwanted/1), dynamic(change/1). This allows the knowledge base to be updated as the user answers the questions.

When the user first runs `ask(0)`, he is queried on whether he would like a normal meal. If not, the system will continue querying the user whether he wants a meal until he enters `X` or he has said no to every option.

While this is less efficient than displaying everything and asking the user to choose, it prevents the user from having to input every option that he wishes and he can simply respond with a yes, a no or a change of option. This also prevents mistaken user input.

After this, the user is queried on the bread and the meat that he wants. The user must respond to this category exactly once since every order must consist of a bread and a meat (unless a veggie or vegan meal is selected). The program does not require the user to select what he wants from a list. Instead, when answering the questions of the knowledge base, the user can respond with `y, n, x`. A `y` means that the customer wants the item and hence, the item is ordered. A `x` means that the customer does not want the item and does not want any more item from that category. A `n` means that the customer does not want the item but is still interested in other items from that category. This continues while there are still items. If there are no items left in that category i.e. the user is not interested in all or the user orders all of them, they will be queried on an item of the next category.

This process continues for all the items. Once the user has chosen a drink or is not interested in a drink. The query ends since a 'full order' has been found. A message saying "Thanks for visiting Subway! Your order will be ready in ten minutes" is printed along with the ingredients that they want.

Special Cases

Here are the special cases which makes the Prolog script 'intelligent'.

- If a person chooses a veggie meal, meats would not be offered
- If a person chooses a vegan meal, meats and cheese would not be offered

- If a person chooses a healthy meal, fatty sauces and sides would not be offered

- If a person chooses a value meal, top-ups for the sub, as well as a side will not be offered

- If a person chooses a normal meal, everything would be offered