Final Project Andrew Wirth

My final project is voice controlled chatbot, similar to siri or OK Google. The agent can be initialized by running the programming. Once the program is run the user is prompted with the agent saying “What can I help you with?”. The agent can then do a variety of tasks.

The agent has two precepts the first is the microphone, the second is the internet. With the microphone the agent is able to listen and interpret questions. It then searches the internet for answers. If an answer is available(I will go into more detail later) then the program will read the answer aloud. The program is enabled by many packages on high level it works by taking using the microphone to convert verbal language into strings it then follows its knowledge base on what to do with those strings and then takes action. During its action it will either create or use a saved response. A response is created by taking a string saving it as a mp3 and the agent then plays the mp3 .

The agent has four main features.It works as a reflexive agent going through preprogrammed interactions. It plays a game of tic-tac-toe against a user using the minimax algorithm. It also uses a machine learning algorithm to decide between whether an interaction is a question or an interaction and search it according to the actions available. Lastly, if an answer is available directly on google it will detect it and “read” it aloud.

The first feature is that it works as a reflexive agent to complete some pre programmed interactions. The first step of the program is to search for a programmed response. This is set up by two dictionaries one being the programmed user input the second is the agent output. They are related through a common key/value. The program searches the user’s phrase against the first dictionaries keys if it is found it then uses the value of the key as the key of the agent responses which returns the name of the reponse mp3 file as output. I set it up this way to make it easy to create new interactions. If you are interested in using the program to set up new interactions uncomment self.logresponses in the initialization of the AI class. After the response has been added, run the program once this will create the corresponding mp3 file for performance comment out that line in future uses.

The second feature is for the agent to play a game of tic tac toe against the user this is initialized by the command “play game”. The tic tac toe game is implemented in a separate class and inherits from a parent class that is the AI class, but modified down to just the needed methods. I set up the program this way to make it more modular. I can now create a new class with any methods and call that class to create a new AI features.

The third feature is an implementation of a Naive Bayes ML model to differentiate between the user interactions that are not in the list of programmed interactions. If the interaction is not in the list the program with use the string to classify the interaction as either a question or an intended purchase. The depending on the classification the agent will announce “I can help you with that purchase” or “I can help you with that question”. If the agent classifies the interaction as a purchase it will search the last noun of the question on target.com. If the agent classifies the interaction as a question the agent will search the question on google. The ML model is trained on over 3000 rows of phrases using a bag of words model. This is done in a separate program and then saved to a PKL file the pkl file is then loaded into the chatbot program and is used to initialize the same saved model and run it on user interactions.

The final feature of the program is simple. It is that if an answer to your question appears on google the agent will read it aloud answering your question.