Chatbot Implementation Report

Muhammad Ahsan

19/2/2023

1 Introduction

The chatbot code presented in this submission is a basic implementation of a chatbot that uses dictionaries in Python to provide predefined answers to commonly asked questions. Additionally, it is integrated with the OpenAI API to answer more complex questions.

2 OpenAI API Configuration

Before running the chatbot, the user must enter their OpenAI API key. The API key is used to authenticate the client and access the OpenAI API services. The API key is stored as an environment variable to maintain security.

3 Chatbot Implementation

The chatbot responses are defined as a dictionary with the questions as keys and their respective answers as values. If the user enters a question that is in the dictionary, the chatbot will provide the pre-defined answer. If the question is not present in the dictionary, the chatbot will use the OpenAI API to generate an appropriate response.

The function `ask_openai(question)` uses the OpenAI API to generate a response. The response is based on the provided question and is generated using a pre-trained language model provided by OpenAI. The language model is selected during configuration and can be changed to a different model if needed.

4 Limitations

The chatbot implementation presented in this submission is a basic implementation and has some limitations. The chatbot is limited to answering questions that are present in the dictionary. If the user asks a question that is not present in the dictionary and cannot be answered by the OpenAI API, the chatbot will provide a generic response, ”I'm sorry, I don't understand. Can you please rephrase your question?”.

5 Conclusion

This chatbot implementation provides a basic framework for integrating the OpenAI API with a chatbot. The implementation can be further improved by adding more questions and answers to the dictionary and by implementing more advanced algorithms to handle a wider range of user input.