Voice Assistant Project

This project implements a Python-based voice assistant that can recognize and respond to voice commands. It performs tasks such as opening applications, fetching information, telling jokes, translating text, and more.

Features

- **Speech Recognition**: Listens to voice commands using the speech_recognition library.
- **Text-to-Speech**: Responds using voice output via the pyttsx3 library.
- **Task Automation**: Opens/Closes applications, searches the web, plays YouTube videos, and switches windows.
- **Generative AI**: Uses Google Generative AI for custom query responses.
- **Wikipedia Integration**: Fetches summaries for queries from Wikipedia.
- **Jokes and Fun**: Tells jokes using the pyjokes library.
- **Multi-language Support**: Translates Hindi commands to English using the googletrans library.
- **Time Reporting**: Tells the current time.
- **Web Search**: Searches Amazon or Google based on commands.

Prerequisites

1. **Python**: Ensure Python 3.7 or later is installed.
2. **API Key**: Get an API key for Google Generative AI.
3. **Libraries**: Install the required Python libraries using:
4. pip install speechrecognition pyttsx3 wikipedia pyjokes pywhatkit pyautogui webbrowser AppOpener googletrans google-generative-ai

---

File Structure

- **jarvis.py**: Contains the primary logic for the voice assistant.
- **Dependencies**: Uses external Python libraries for specific functionalities.

How to Run

1. **Configure API Key**: Replace the placeholder API_KEY in the code with your Google Generative AI key.
2. **Run the Program**: Execute the script:

   RUN : jarvis.py or dist/jarvis.exe
3. **Interact with the Assistant**:
   - Speak commands into the microphone.
   - Example commands:
     - "Open YouTube"
     - "Tell the time"
     - "Tell me a joke"
     - "Search on Google for Python programming"

Supported Commands

Command	Action
"Open YouTube"	Opens YouTube in the default browser.
"Tell the time"	Tells the current time.
"Search on Amazon ..."	Searches the specified item on Amazon.
"Play on YouTube ..."	Plays the specified video on YouTube.
"Switch tab/window"	Switches between open windows.
"According to Wikipedia"	Provides a Wikipedia summary for the given query.
"Tell me a joke"	Tells a random joke.
Custom Queries	Uses Generative AI to provide responses for unspecified commands.

Libraries Used

1. **speech_recognition**: For speech-to-text conversion.
2. **pyttsx3**: For text-to-speech functionality.
3. **wikipedia**: For fetching summaries from Wikipedia.
4. **pyjokes**: For generating random jokes.

5. **pywhatkit**: For playing YouTube videos and performing web searches.
6. **pyautogui**: For window switching and automation.
7. **webbrowser**: For opening web pages.
8. **AppOpener**: For opening and closing desktop applications.
9. **googletrans**: For translating Hindi commands to English.
10. **google-generative-ai**: For processing natural language queries using Google's Generative AI.

Known Issues

- **Speech Recognition Errors**: May fail if the microphone input is unclear.
- **Google Generative AI Limitations**: Ensure your API key is active and valid.
- **Application Matching**: Opening/closing applications relies on accurate matching; may fail for ambiguous names.

Future Enhancements

- Add support for more languages.
- Improve error handling and debugging features.
- Expand functionality for smart home automation.
