

# **Artificial Intelligence**

## **LAB PROJECT SUBMISSION**

**Submitted to:** Ms Navpreet

**Submitted by:**

Anupriya Lathey 102103373

Nitleen Kaur 102103377

Hargun Singh Walia 102103386

Radhika Jasra 102103815

---

### **Problem statement:**

To understand speech commands given by user, record them and convert them to text.

### **Description of problem:**

The AI app uses speech recognition technology to convert spoken words into text format. The built-in microphone that can pick up the user's voice and send the audio data to the speech recognition algorithm. The code uses various techniques, such as signal processing, machine learning, and pattern recognition to convert the audio into text.

Once the text is extracted, the app identifies the website mentioned in the user's speech. The app then uses browser automation technology to open the website mentioned by the user automatically. The automation technology uses computer vision techniques to locate and interact with the browser interface, allowing the app to type the website's URL and open it.

It user-friendly and easy to use. The user simply needs to speak the website's name, and the code does the rest. This can be used to improve productivity and save time by quickly

opening websites without the need to manually type in the website's URL. Additionally, this project can be used by individuals with disabilities that make typing difficult or impossible, making it an inclusive solution for website browsing.

## Utility:

Speech recognition AI is the process of converting spoken language into text. The technology uses machine learning and neural networks to process audio data and convert it into words that can be used in businesses. Speech recognition AI can be used for various purposes, including dictation and transcription.

The technology is also used in voice assistants like Siri and Alexa. Speech communication is using speech recognition and speech synthesis to communicate with a computer. Speech recognition can allow users to dictate text into a program, saving time compared to typing it out. Speech synthesis is used for chatbots and voice assistants like Siri and Alexa. AI and machine learning are used in advanced speech recognition software, which processes speech through grammar, structure, and syntax. Imprecise and misleading translations.

Speech recognition software can occasionally misinterpret what someone is saying. Computers have difficulty understanding the contextual relation of words and sentences, leading them to misinterpret what a speaker means.

## Explanation:

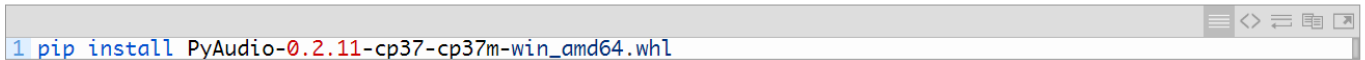
For the installation you can just use pip, also you can download the source distribution from [PyPI](#), and extract the archive.

A screenshot of a terminal window with a light gray background. The command '1 pip install SpeechRecognition' is entered in the terminal's input line. The terminal has standard window controls (minimize, maximize, close) in the top right corner.

### **PyAudio (for microphone users):**

**PyAudio** is required if and only if you want to use microphone input (Microphone).

PyAudio version 0.2.11+ is required, as earlier versions have known memory management bugs when recording from microphones in certain situations. now if you are using Python 3.6 you can install pyaudio using *pip install pyaudio*, but if you are using Python 3.7 or 3.8 you need to download the .whl file from website, [PyAudio Whl Download](#). For example, in Python 3.7 you can use *PyAudio-0.2.11-cp37-cp37m-win\_amd64.whl* and use the command as go to the download directory.

A screenshot of a terminal window with a light gray background. The command `1 pip install PyAudio-0.2.11-cp37-cp37m-win_amd64.whl` is entered in the terminal. The terminal has standard window controls (minimize, maximize, close) in the top right corner.

In this example we are going to convert our audio to text , we want to say something using Microphone, and after that it will be automatically converted to text and saved in our working directory.

### **Brief of the libraries used for this purpose:**

- **SpeechRecognition** incorporates computer science and linguistics to identify spoken words and converts them into text. It allows computers to understand human language.
- **PyAutoGUI** lets Python scripts control the mouse and keyboard to automate interactions with other applications.
- **PyAudio** provides Python bindings for PortAudio v19, the cross-platform audio I/O library. With PyAudio, you can easily use Python to play and record audio on a variety of platforms, such as GNU/Linux, Microsoft Windows, and Apple macOS.
- **pyttsx3** is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

The code primarily uses two modules - **speech\_recognition** and **pyautogui** for speech recognition and browser automation respectively. The `speech_recognition` module uses a combination of algorithms and data structures to recognize the user's voice input, such as signal processing, machine learning, and pattern recognition techniques. On the other hand, the `pyautogui` module is mainly used for automating the keyboard and mouse input to control the browser. This module uses various data structures and algorithms, such as computer vision techniques to locate and interact with specific elements on the screen.

```
1  import speech_recognition as sr
2  import pyautogui
3
4  def main():
5
6      r = sr.Recognizer()
7
8      with sr.Microphone() as source:
9          r.adjust_for_ambient_noise(source)
10
11         print("Please say something")
12
13         audio = r.listen(source)
14
15         print("Recognizing Now .... ")
16
17
18         # recognize speech using google
19
20         try:
21             website = r.recognize_google(audio)
22             print("You have said: " + website)
23             print("Audio Recorded Successfully \n ")
24
25             # open the website in the default browser
26             pyautogui.press("win")
27             pyautogui.write("chrome")
28             pyautogui.press("enter")
29             pyautogui.write("https://www." + website + ".com")
30             pyautogui.press("enter")
31
32         except Exception as e:
33             print("Error : " + str(e))
34
35         # write audio
36         with open("recorded.wav", "wb") as f:
37             f.write(audio.get_wav_data())
38
39 if __name__ == "__main__":
40     main()
```

In here we have created the object of our Recognizer and also we are using Microphone as source.

```
1 r = sr.Recognizer()
```

also we need to add this line of code, it is used for removing noises if we have in the sound.

```
1 r.adjust_for_ambient_noise(source)
```

And in here we are recognizing the speech using Google Speech.

```
1 print("You have said \n" + r.recognize_google(audio))
```

If you need to record your audio than you can use this code.

```
1 with open("recorded.wav", "wb") as f:  
2     f.write(audio.get_wav_data())
```

Run the code say something in the Microphone and this is the result.

```
"C:\Program Files\Python37\python.exe"  
Please say something  
Recognizing Now ....  
You have said  
hello friend how are you  
Audio Recorded Successfully  
  
Process finished with exit code 0
```

Opening Website Using Speech Recognition:

```
#This can be done by using following commands-  
  
import pyautogui
```

```
# open the website in the default browser
pyautogui.press("win")
pyautogui.write("chrome")
pyautogui.press("enter")
pyautogui.write("https://www." + website)
pyautogui.press("enter")
```

## Convert Recorded Audio To Text:

Till now we have learned that how we can convert audio using microphone in python, now sometimes we need to convert a recorded audio to text. This can be done with following command.

## Datasets used by SpeechRecognition:

Google Speech Recognition uses a variety of datasets to train its machine learning models. Some of the major datasets used by Google for speech recognition include:

- Google's internal speech corpus: This dataset includes a large collection of anonymized voice search queries and commands from Google Assistant users, which is used to improve the accuracy and performance of Google's speech recognition system.
- Common Voice: This is an open-source dataset developed by Mozilla that contains over 9,000 hours of speech data contributed by volunteers from around the world. Google uses this dataset to train its models for recognizing a wide variety of accents and languages.
- Switchboard Corpus: This is a dataset of telephone conversations collected by the Linguistic Data Consortium at the University of Pennsylvania. It is widely used for training speech recognition models, including by Google.

- VoxCeleb: This is a dataset of over 1 million short audio clips of celebrities speaking, which is used to train models for speaker recognition and verification.
- LibriSpeech: This is a dataset of audiobooks that contains over 1,000 hours of speech data in English, which is used to train models for speech recognition and natural language processing.

Overall, Google uses a combination of proprietary and publicly available datasets to train its speech recognition models, with a focus on improving accuracy and performance for a wide range of languages, accents, and use cases.

## Requirements:

To use all the functionality of SpeechRecognition library, you should have:

- Python 3.8+ (required)
- PyAudio 0.2.11+ (required only if you need to use microphone input, Microphone)
- PocketSphinx (required only if you need to use the Sphinx recognizer, `recognizer_instance.recognize_sphinx`)
- Google API Client Library for Python (required only if you need to use the Google Cloud Speech API, `recognizer_instance.recognize_google_cloud`)
- FLAC encoder (required only if the system is not x86-based Windows/Linux/OS X)
- Vosk (required only if you need to use Vosk API speech recognition `recognizer_instance.recognize_vosk`)
- Whisper (required only if you need to use Whisper `recognizer_instance.recognize_whisper`)
- openai (required only if you need to use Whisper API speech recognition `recognizer_instance.recognize_whisper_api`)

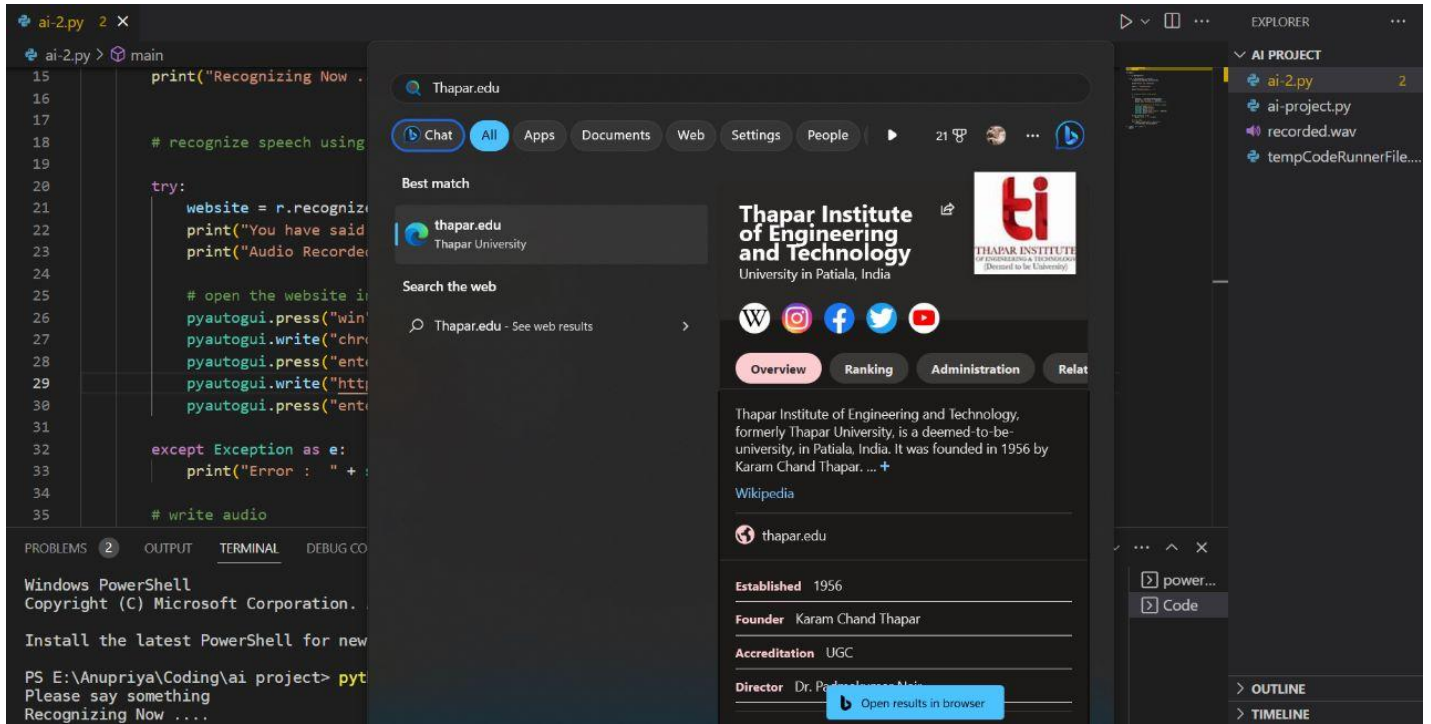
## WORKING EXAMPLES:

```
ai-2.py > main
15     print("Recognizing Now .... ")
16
17
18     # recognize speech using google
19
20     try:
21         website = r.recognize_google(audio)
22         print("You have said: " + website)
23         print("Audio Recorded Successfully \n ")
24
25         # open the website in the default browser
26         pyautogui.press("win")
27         pyautogui.write("chrome")
28         pyautogui.press("enter")
29         pyautogui.write("https://www." + website )
30         pyautogui.press("enter")
31
32     except Exception as e:
33         print("Error : " + str(e))
34
```

PROBLEMS 2 OUTPUT TERMINAL DEBUG CONSOLE

```
PS E:\Anupriya\Coding\ai project> python -u "e:\Anupriya\Coding\ai project\ai-2.py"
Please say something
Recognizing Now ....
You have said: geeks.org
Audio Recorded Successfully
Please say something
Recognizing Now ....
You have said: thapar.edu
```





## Link to the Project:

<https://drive.google.com/drive/folders/148eda0C2EKMFxAPxUr-VnPIEQHhYAUwq?usp=sharing>