

SCHOOL OF COMPUTING
COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR - 603203

December 2022

Speech Recognition (Speech To Text)

Submitted by

Sri Renga V.R.C (RA2211026010525)
AND
Shashanka Niraula (RA2211026010528)

Under the Guidance of

Dr.Saad Y. Sait

Associate Professor, Department of Computational Intelligence

In partial satisfaction of the requirements for the degree of

BACHELORS OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING
with specialization in
ARTIFICIAL INTELLIGENCE
and
MACHINE LEARNING



SRM INSTITUTE OF SCIENCE AND
TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Course Project Report titled “**Speech Recognition (Speech to Text)**” is the bonafide work done by **Sri Renga V.R.C [RA2211026010525]** and **Shashanka Niraula [RA2211026010528]** who carried out under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE

Faculty In-Charge

Dr. Saad Y. Sait

Associate Professor

Department of Computational Intelligence

SRM Institute of Science and Technology

HEAD OF THE DEPARTMENT

Dr. Annie Uthra R.

Professor and Head ,

Department of Computational Intelligence

SRM Institute of Science and Technology

Programing for Problem Solving

Subject code-21CSS101J



By:

Sri Renga V.R.C (RA2211026010525)

AND

Shashanka Niraula (RA2211026010528)

To:

Dr. Saad Y. Sait

Project Information :

Name : Speech Recognition (Speech To Text)

Problem Statement : Coding to build a program that identifies words spoken by the user and prints it in readable text .

Interpreter Used : Python

Software Used : Visual Studio Code

Table of Contents :

TITLE NUMBER	TITLE	PAGE NO.
1	What is Speech Recognition ?	3
2	Module	4
3	Code	6
4	How the Program Works	7
5	Output	9
6	Conclusion	10

What is speech recognition?

The process of understanding the words that are spoken by human beings. Remember that the speech signals are captured with the help of a microphone and then it has to be understood by the system and it is used in several application such as artificial intelligence

How does speech recognition work?

- First, speech recognition that allows the machine to catch the words, phrases and sentences we speak
- Second, natural language processing to allow the machine to understand what we speak, and
- Third, speech synthesis to allow the machine to speak.

Module :

A module is a software component or part of a program that contains one or more routines. A Python module is a file containing Python definitions and statements. A module can define functions, classes, and variables. A module can also include runnable code. Grouping related code into a module makes the code easier to understand and use. It also makes the code logically organized.

Module Used For This Project :

- Python speechrecognition
- Pyaudio
- Python pyttsx3

Installation of module:

Speechrecognition:

Pip install speechrecognition

(Speech recognition is a machine's ability to listen to spoken words and identify them. You can then use speech recognition in Python to convert the spoken words into text, make a query or give a reply. You can even program some devices to respond to these spoken words.)

Pyaudio:

Pip install pyaudio

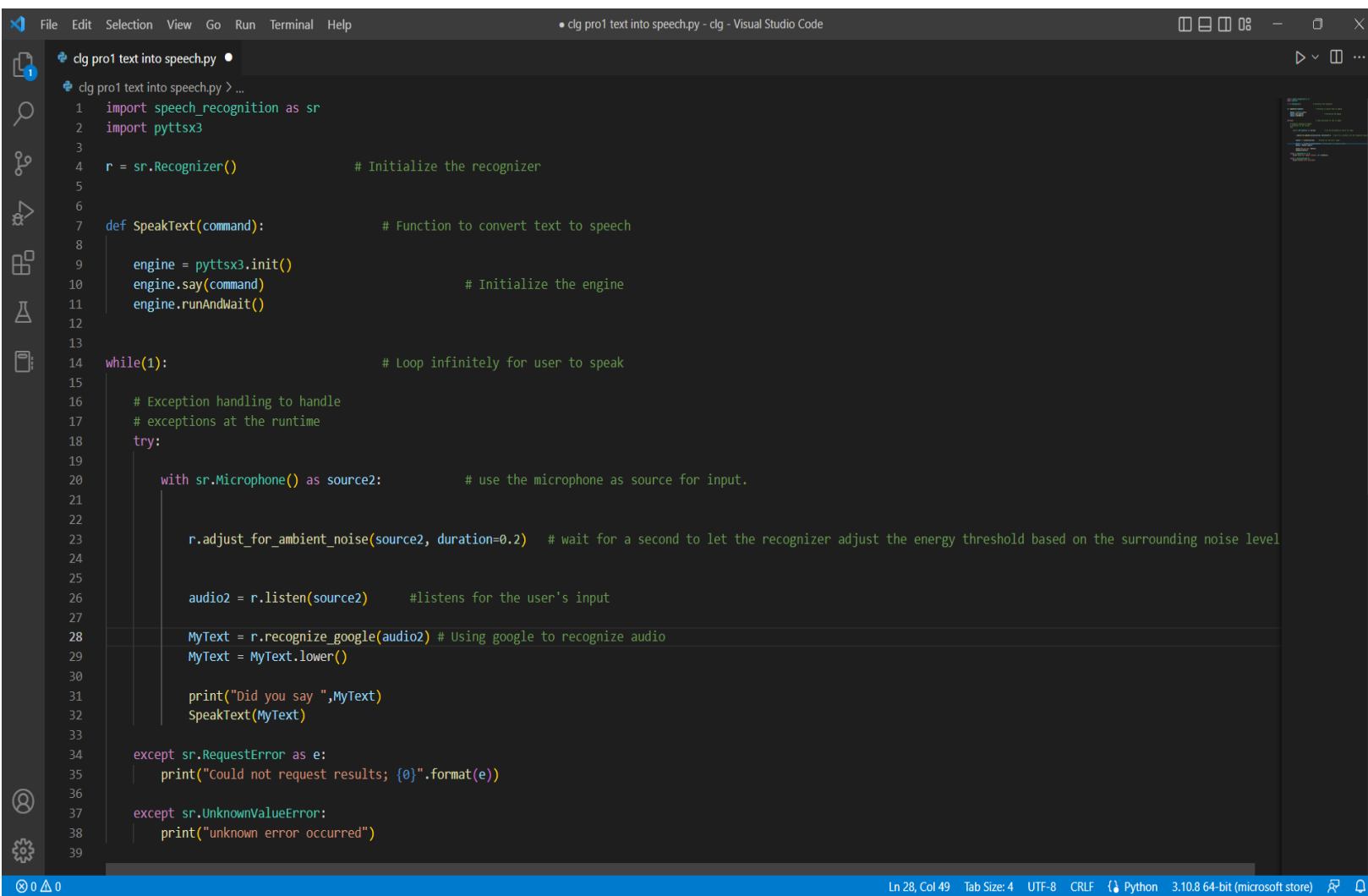
(PyAudio provides Python bindings for PortAudio v19, the cross-platform audio I/O library. With PyAudio, you can easily use Python to play and record audio on a variety of platforms, such as GNU/Linux, Microsoft Windows, and Apple macOS. PyAudio is distributed under the MIT License.)

Pyttsx3:

Pip install pyttsx3

(pyttsx3 is a text-to-speech conversion library in Python)

Code:



The image shows a Visual Studio Code editor window with a Python script titled "clg pro1 text into speech.py". The script is designed to convert text to speech using the SpeechRecognition and pyttsx3 libraries. It includes a function to speak text and a main loop that listens for user input and responds with speech. The script is written in Python 3 and uses Google's speech recognition API.

```
1 import speech_recognition as sr
2 import pyttsx3
3
4 r = sr.Recognizer()          # Initialize the recognizer
5
6
7 def SpeakText(command):      # Function to convert text to speech
8
9     engine = pyttsx3.init()
10    engine.say(command)        # Initialize the engine
11    engine.runAndWait()
12
13
14 while(1):                    # Loop infinitely for user to speak
15
16     # Exception handling to handle
17     # exceptions at the runtime
18     try:
19
20         with sr.Microphone() as source2:    # use the microphone as source for input.
21
22
23             r.adjust_for_ambient_noise(source2, duration=0.2) # wait for a second to let the recognizer adjust the energy threshold based on the surrounding noise level
24
25
26             audio2 = r.listen(source2)      #listens for the user's input
27
28             MyText = r.recognize_google(audio2) # Using google to recognize audio
29             MyText = MyText.lower()
30
31             print("Did you say ",MyText)
32             SpeakText(MyText)
33
34     except sr.RequestError as e:
35         print("Could not request results; {0}".format(e))
36
37     except sr.UnknownValueError:
38         print("unknown error occurred")
39
```

The status bar at the bottom indicates the current position is Line 28, Column 49. The editor is configured with Tab Size: 4, UTF-8 encoding, and CRLF line endings. The Python interpreter is set to 3.10.8 64-bit (microsoft store).

How does the program work ?

Part 1:

```
1  import speech_recognition as sr
2  import pyttsx3
3
4  r = sr.Recognizer()
5
6
7  def SpeakText(command):
8
9      engine = pyttsx3.init()
10     engine.say(command)
11     engine.runAndWait()
12
```

This code imports the 'speech_recognition' and 'pyttsx3' libraries and initializes the 'r' variable as a 'Recognizer' object from the 'speech_recognition' library. This 'Recognizer' object will be used to recognize speech from the user's input and convert it to text. The 'pyttsx3' library will be used to convert text to speech and speak it aloud. This code sets up the necessary components for the speech-to-text and text-to-speech program, but it does not include the full functionality of the program. a function called 'SpeakText' that takes in a string as an argument. This string is the text that will be converted to speech and spoken aloud. Inside the function, the 'pyttsx3' library is used to initialize a text-to-speech engine. The 'say' method is then used to speak the text that was passed in as an argument, and the 'runAndWait' method is used to run the engine and wait until the text has been spoken before continuing with the rest of the program. This function can be used to convert text to speech and speak it aloud, as part of a larger speech-to-text and text-to-speech program.

Part 2:

```
while(1):                                     # Loop infinitely for user input

    # Exception handling to handle
    # exceptions at the runtime
    try:

        with sr.Microphone() as source2:      # use the microphone as source for input

            r.adjust_for_ambient_noise(source2, duration=0.2) # Waits for a few seconds to
            # get an idea of ambient noise level

            audio2 = r.listen(source2)         #listens for the user's input

            MyText = r.recognize_google(audio2) # Using google to recognize speech
            MyText = MyText.lower()

            print("Did you say ",MyText)
            SpeakText(MyText)

    except sr.RequestError as e:
        print("Could not request results; {}".format(e))

    except sr.UnknownValueError:
        print("unknown error occurred")
```

This code defines an infinite loop that continuously listens for user input from the microphone, recognizes the speech using the 'speech_recognition' library, converts the speech to text, and then speaks the text aloud using the 'SpeakText' function defined earlier. The 'with' statement is used to initialize the microphone as the source of input, and the 'adjust_for_ambient_noise' method is used to adjust the energy threshold of the recognizer based on the surrounding noise level. The 'listen' method is then used to listen for user input, and the 'recognize_google' method is used to recognize the speech and convert it to text. The resulting text is converted to lowercase and printed to the screen. The 'SpeakText' function is then called and passed the text as an argument, causing it to be spoken aloud. The code includes error handling to catch any exceptions that may occur during runtime, such as a 'RequestError' if there is an issue with the request to the Google speech recognition API or an 'UnknownValueError' if the recognizer is unable to recognize the speech. In these cases, appropriate error messages are printed to the screen.

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
[Running] python -u "c:\Users\HARI\Documents\clg\clg pro1 text into speech.py"
result2:
{  'alternative': [  {'confidence': 0.92995489, 'transcript': 'hello'},
                    {'transcript': 'helo'},
                    {'transcript': 'hallo'},
                    {'transcript': 'helloo'},
                    {'transcript': 'hello I'}],
  'final': True}
Did you say  hello

[Done] exited with code=1 in 20.761 seconds
```

Difficulty and Problems while coding :

- Python being a well used and recognized programming language with various new library and modules constantly being added many old library and modules are left un-updated causing various version compatibility issues
- With more than 7000 languages around the world the program becomes limited as sometimes its unable to understand dialects of various language speakers

Conclusion:

The Program was successfully executed in Python which helped us learn how to use various features of Python and better use the knowledge we gained in theory and lab classes to create something to show and be proud of .