Python: Convert Speech to text and text to Speech

Speech Recognition is an important feature in several applications used such as home automation, artificial intelligence, etc. This article aims to provide an introduction on how to make use of the SpeechRecognition and pyttsx3 library of Python.

**Installation required:**

* **Python Speech Recognition module:**

pip install speechrecognition

* **PyAudio:** Use the following command for linux users
* sudo apt-get install python3-pyaudio

Windows users can install pyaudio by executing the following command in a terminal

pip install pyaudio

**Python pyttsx3 module:**

pip install pyttsx3

**Speech Input Using a Microphone and Translation of Speech to Text**

* **Allow Adjusting for Ambient Noise:** Since the surrounding noise varies, we must allow the program a second or too to adjust the energy threshold of recording so it is adjusted according to the external noise level.
* **Speech to text translation:** This is done with the help of Google Speech Recognition. This requires an active internet connection to work. However, there are certain offline Recognition systems such as PocketSphinx, but have a very rigorous installation process that requires several dependencies. Google Speech Recognition is one of the easiest to use.

**Translation of Speech to Text:**

First, we need to import the library and then initialize it using init() function. This function may take 2 arguments.

init(driverName string, debug bool)

* **drivername:** [Name of available driver] sapi5 on Windows | nsss on MacOS
* debug: to enable or disable debug output

After initialization, we will make the program speak the text using say() function.  
This method may also take 2 arguments.

say(text unicode, name string)

* **text:** Any text you wish to hear.
* **name:** To set a name for this speech. (optional)

Finally, to run the speech we use runAndWait() All the say() texts won’t be said unless the interpreter encounters runAndWait().

Below is the implementation.

# Python program to translate speech to text and text to speech

import speech\_recognition as sr

import pyttsx3

# Initialize the recognizer

r = sr.Recognizer()

# Function to convert text to speech

def SpeakText(command):

    # Initialize the engine

    engine = pyttsx3.init()

    engine.say(command)

    engine.runAndWait()

# Loop infinitely for user to speak

while(1):

    # Exception handling to handle exceptions at the runtime

    try:

# use the microphone as source for input.

        with sr.Microphone() as source2:

            # wait for a second to let the recognizer adjust the energy

threshold based on the surrounding noise level

            r.adjust\_for\_ambient\_noise(source2, duration=0.2)

            #listens for the user's input

            audio2 = r.listen(source2)

            # Using ggogle to recognize audio

            MyText = r.recognize\_google(audio2)

            MyText = MyText.lower()

            print("Did you say "+MyText)

            SpeakText(MyText)

    except sr.RequestError as e:

        print("Could not request results; {0}".format(e))

    except sr.UnknownValueError:

        print("unknown error occured")

# Speech To Text using IBM Watson Studio

**IBM Watson Studio** is an **integrated environment** designed to develop, train, manage models, and deploy AI-powered applications and is a **Software as a Service (SaaS)** solution delivered on the IBM Cloud. The IBM Cloud provides lots of **services** like Speech To Text, Text To Speech, Visual Recognition, Natural Language Classifier, Language Translator, etc.

The Speech to Text service transcribes audio to text to enable speech transcription capabilities for applications.

#### Create an instance of the service

1. Go to the [Speech to Text](https://cloud.ibm.com/catalog/services/speech-to-text) page in the IBM Cloud Catalog.
2. Sign up for a free IBM Cloud account or log in.
3. Click **Create**.

#### Copy the Credentials to Authenticate to your service instance

1. From the [IBM Cloud Resource list](https://cloud.ibm.com/resources), click on your Speech to Text service instance to go to the Speech to Text service dashboard page.
2. On the **Manage** page, click **Show Credentials** to view your credentials.
3. Copy the **API Key** and **URL** values.

**Module Needed:**

1. **Json**
2. **ibm\_watson:** This module does not comes pre-defined with Python. To install it type the below command in the terminal.

pip install ibm\_watson

Now you’re ready to use the IBM Cloud Services.

**Below code illustrates the use of IBM Watson studio’s Speech To Text Service using Python and web socket interface**

#Python Program To Use IBM Watson

# Studio's Speech To Text Below Code

# Accepts only .mp3 Format of Audio

# File

import json

from os.path import join, dirname

from ibm\_watson import SpeechToTextV1

from ibm\_watson.websocket import RecognizeCallback, AudioSource

from ibm\_cloud\_sdk\_core.authenticators import IAMAuthenticator

# Insert API Key in place of  'YOUR UNIQUE API KEY'

authenticator = IAMAuthenticator('YOUR UNIQUE API KEY')

service = SpeechToTextV1(authenticator = authenticator)

#Insert URL in place of 'API\_URL'

service.set\_service\_url('API\_URL')

# Insert local mp3 file path in place of 'LOCAL FILE PATH'

with open(join(dirname('\_\_file\_\_'), r'LOCAL FILE PATH'),

          'rb') as audio\_file:

dic = json.loads(

                json.dumps(

                    service.recognize(

                        audio=audio\_file,

                        content\_type='audio/flac',

                        model='en-US\_NarrowbandModel',

                    continuous=True).get\_result(), indent=2))

# Stores the transcribed text

str = ""

while bool(dic.get('results')):

    str = dic.get('results').pop().get('alternatives').pop().get('transcript')+str[:]

print(str)