

An end-to-end Guide on Converting Text to Speech and Speech to Text

[ADVANCED](#) [AUDIO PROCESSING](#) [NLP](#) [PYTHON](#) [TEXT](#)

This article was published as a part of the [Data Science Blogathon](#).

Hey Folks!

In this article, we are going to discuss Speech Recognition and its application of it by implementing a Speech to Text and Text to Speech Model with Python. Speech Recognition is also known as Speech Text conversion or simply Voice Recognition. This is the technique of making computers understand human language. Have you ever wondered how amazon's Alexa apple's Siri and google's voice assistant talk to us and understand our language, this is done by Speech Recognition?

Table of Content

1. Basic Idea behind Speech Recognition
2. Implementing Speech2Text Model
3. Implementing the text2speech Model
4. Language Translation

INTRODUCTION

Speech Recognition is a very important task in NLP. Speech Recognition is the only medium to make computers understand our spoken speech. As we know computers can easily understand a written text by converting text into features (numerical features) by implementing various feature extraction techniques.

Here the idea is to convert spoken speech into text and then feed it to computers.

There are numerous applications of Speech Recognition some major applications are:

- It is very useful for making projects for physically disabled people.
- Designing a talking Bot
- Language Translator using Speech
- Offensive speech detection
- Smart Gadgets working on voice commands
- Military Equipment

Speech to Text Conversion

Nowadays interaction with computers and smart devices is tending towards the voice. Devices working on Voice Commands are quick effective and have to be smarter. Since machines can understand the text by

applying some feature extraction techniques our goal is to convert any speech into a text.

Business Problem

We want to convert speech into text

Solution

there are various technologies available to perform speech to text but [PyAudio](#) provides a very easy and efficient implementation.

Implementation Using Python

installing libraries

```
!pip install SpeechRecognition !pip install PyAudio

# if pip install PyAudio throws error try: !conda install pyaudio
```

PyAudio is used to record and play an audio file with Python. it enables the microphone with python

SpeechRecognition takes an AudioData instance and converts it into text. this works online using the Google Speech Recognition API.

```
import speech_recognition as sr
r=sr.Recognizer()
with sr.Microphone() as source:
    print("Please say something")
    audio = r.listen(source)
    print("Time over, thanks")
    try:
        print("You said: "+r.recognize_google(audio,language = 'en-US'))
    except:
        pass
```

Output

```
Please say something Time over, thanks you said: This is Speech Recognition done by NLP
```

- **sr.Recognizer()** is a recognizer instance

recognizer_instance.recognize_google(audio_data, language = "en-US")

- We can switch the language we are speaking by changing parameters. the default language is set to **'en-US'**
- If you want to recognize HINDI we need to change the language parameter only **recognize_google(audio, language = 'hi-IN')**

Text to Speech Recognition

TTS(Text to Speech) interface that allows the computer to read a text like a human. this is also called *read-aloud* technology.

In the real world, we can see numerous applications of the TTS system. this is widely used to make smart devices that can interact with humans.

There are some major applications of the TTS system:

1. Devices for blind people who can't see but can listen. A device that can read text using OCR (Optical Character Recognition) and using text to speech it can read aloud.

2. Smart Devices and Voice Assistants

3. Text to Speech comes very useful for physically disabled people, ie it can be used in mobile phones, computers to guide blind people.

Problem

We want to create a system that can read a given text in a human's voice.

Solution

There could be multiple ways to perform Text2Speech but the easiest and most efficient way is to use Google's API using the [gTTS](#) library

Implementation using Python

- Installing **gTTS** library

```
!pip install gTTS
```

- After installing **gTTS** let's load and work with it

```
from gtts import gTTS input_text = "I like NLP and now this is machine voice" convert = gTTS(text=input_text, lang='en', slow=False)
```

- Saving the converted audio into an mp3 file

```
convert.save('audio.mp3')
```

If you play audio.mp3 you would listen to *"I like NLP and now this is machine voice"* in a human's voice.

there are some parameters used to change the voice and control voice speed using parameters. For more information refer to this [link](#).

Language Translation

We have discussed Speech to Text and Text to Speech now we will talk about language translation using python

Using these 3 technologies we can create our own Language Translator that takes Speech and convert it into the desired language's Speech

As we all know Language translation is widely used nowadays. language translation can take language in the form of speech, text as well as pictures.

Google's Language Translator system is most widely used and it supports almost every major language.

Google's Language Translator is supported by Attention layers that make it very robust compared to other translator models.

Problem

Create a Model that can translate a given text into the desired language

Solution

The most effective and easiest way to implement language translation for your project is to use the library **goslate** that works using Google's Translator API in the backend

goslate provides us python API to google translation service by querying google translation website.

Implementing Language Translator using Python

- Installing and importing **goslate**

```
!pip install goslate import goslate
```

- Creating a translator function

```
text = "Bonjour le monde" gs = goslate.Goslate() translatedText = gs.translate(text, 'en')  
print(translatedText)
```

Output

```
Hello World
```

- **goslate.Goslate()** is a translator's instance
- we can switch language by language parameters

goslate can also be used to detect language. **Goslate.detect('text')** returns the language of the text.

```
gs.detect('hallo welt')
```

we can also query concurrent text by passing an array of text into **.translate()** method.

For more detailed documentation on **goslate** refer to this [link](#).

Use Cases

- You can create a device that can read the text and read aloud using low-end computer devices like the raspberry pi. this can be really useful for blind people who can't read or have low vision.
- Using these libraries you can create a Translator device using a low-end computer like raspberry pi that can take speech and translate it back into a speech. This can be done using text2speech, language translation, and speech2text. We can also implement OCR for character recognition for language translation(image to text). Such devices are easy to create and it's great for the portfolio showcase.

Industry Applications of NLP

I believe that you are comfortable with the [basics of natural language processing](#), you have already implemented some basic NLP tasks, and you are ready to solve some real-world business problems using NLP

In the Next Article, we will Implement Industry Applications of NLP ie.

- Consumer complaint classification
- Data stitching using record linkage
- Text summarization for subject notes
- Document clustering
- Search engine and learning to rank

These Tasks contain some series of concepts of NLP that will be leveraged while building these applications. So Stay Tuned for My next article that going to be an end-to-end guide on industry applications of NLP

EndNote

In this article, we have discussed speech2text using (**pyaudio**, speech recognition) and implemented on python. then we covered text2speech using the library **gTTS** that simply queries to google's text2speech API in the backend. then we covered Language Translation using the library **goslate** that is again supported by Google's Translator API in the backend.

Read more articles on converting [text to speech](#) topics.

If you have any suggestions or questions for me feel free to hit me on my [Linkedin](#).

The media shown in this article is not owned by Analytics Vidhya and are used at the Author's discretion.

Article Url - <https://www.analyticsvidhya.com/blog/2022/01/an-end-to-end-guide-on-converting-text-to-speech-and-speech-to-text/>



[Abhishek Jaiswal](#)