MIS4311
Machine Learning Applications

Spring 2025

Lecture #9

Speech Recognition

Speech processing system has mainly **three tasks**:

- **speech recognition** that allows the machine to catch the words, phrases and sentences we speak
- **Natural language processing** to allow the machine to understand what we speak
- **Speech synthesis** to allow the machine to speak.

We will focus on **speech recognition** that is the process of **understanding the words** that are spoken by human beings.

Speech Recognition or Automatic Speech Recognition (ASR) is the center of attention **for AI projects like robotics.**

Visualizing Audio Signals with Python

Two steps:

Recording

When you have to read the audio signal **from a file**, then record it using a microphone, at first. (You can **download the sample.wav file** from bulut.marmara.edu.tr)

Sampling

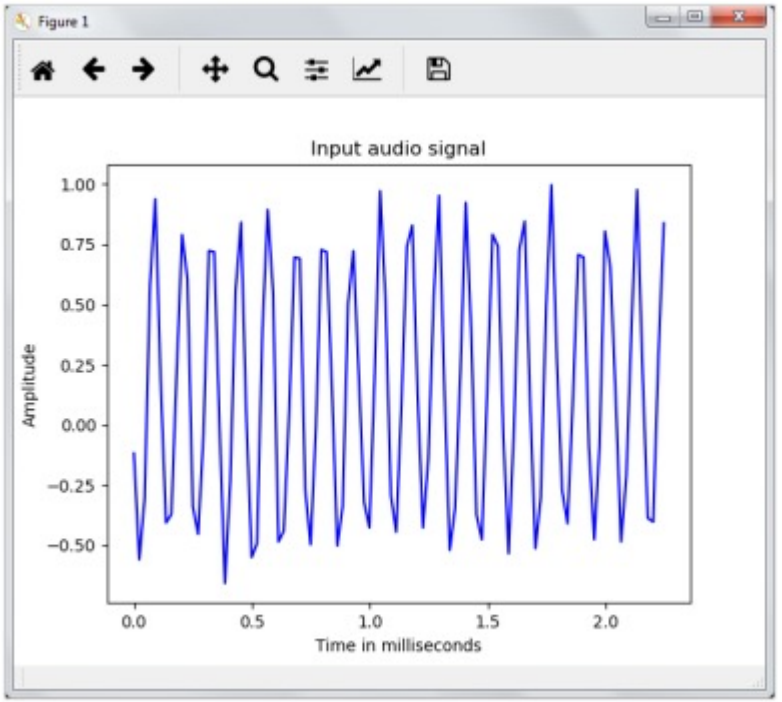
When **recording with microphone**, the signals are **stored in a digitized form.** But to work upon it, the machine needs them in the **discrete numeric form**. Hence, we should perform sampling at a certain frequency and **convert the signal into the discrete numerical form.** Choosing the **high frequency for sampling** implies that when humans listen to the signal, they **feel it as a continuous audio signal**.

Visualizing Audio Signals with Python

We show a stepwise approach to analyze an audio signal, using Python, which is stored in a file.

Display the parameters like **sampling frequency** of the audio signal, **data type** of signal and its **duration.**

Visualize the audio signal.



Generating Monotone Audio Signal with Python

In the second application we generate the audio signal with some **predefined parameters** and **save** the audio signal in an **output file**.

We define these parameters as:
duration = 4 # in seconds
frequency_sampling = 44100 #in hertz
frequency_tone = 784

Recognition of Spoken Words with Python

In the third application we recognize the speech via michrophone.
Speech recognition means that when humans are speaking, a machine understands it.
Here we are using Google Speech API in Python to make it happen.

We need to install the following packages:

conda install -c conda-forge **SpeechRecognition** #for sound recognition
conda install -c conda-forge **gTTS** #to use Google Translate API
conda install -c conda-forge **Playsound** #to play sound
conda install -c conda-forge **PyAudio** #for sound input and output

Next Week

- **Natural Language Processing (NLP)**

Thank you for your participation ☺