**Bài 16: Nhận diện giọng nói - phần 2 - Lập trình AI bằng Python**

**4. Tạo tín hiệu âm thanh đơn điệu :**

Hai bước mà bạn đã thấy rất quan trọng để tìm hiểu về các tín hiệu. Bây giờ, bước này sẽ hữu ích nếu bạn muốn tạo tín hiệu âm thanh với một số thông số được xác định trước. Lưu ý rằng bước này sẽ lưu tín hiệu âm thanh trong tệp đầu ra.

Trong ví dụ sau, ta sẽ tạo một tín hiệu đơn điệu, sử dụng Python, sẽ được lưu trữ trong một tệp

**import** numpy as np

**import** matplotlib.pyplot as plt

from scipy.io.wavfile **import** write

Cung cấp tệp nơi tệp đầu ra sẽ được lưu

output\_file = 'audio\_signal\_generated.wav'

 chỉ định các tham số bạn chọn như sau :

duration = 4 # in seconds

frequency\_sampling = 44100 # in Hz

frequency\_tone = 784

min\_val = -4 \* np.pi

max\_val = 4 \* np.pi

Trong bước này, chúng ta có thể tạo ra tín hiệu âm thanh

t = np.linspace(min\_val, max\_val, duration \* frequency\_sampling)

audio\_signal = np.sin(2 \* np.pi \* tone\_freq \* t)

Lưu tệp âm thanh trong tệp đầu ra :

**write**(output\_file, frequency\_sampling, signal\_scaled)

Trích xuất 100 giá trị đầu tiên cho biểu đồ

audio\_signal = audio\_signal[:100]

time\_axis = 1000 \* np.arange(0, len(signal), 1) / float(sampling\_freq)

Hình dung tín hiệu âm thanh được tạo ra như sau:

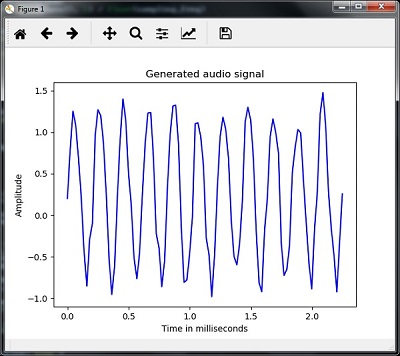
**plt**.plot(time\_axis, signal, color='blue')

**plt**.xlabel('Time in milliseconds')

**plt**.ylabel('Amplitude')

**plt**.title('Generated audio signal')

**plt**.show()



**5. Tính năng trích xuất từ giọng nói :**

Đây là bước quan trọng nhất trong việc xây dựng bộ nhận dạng giọng nói vì sau khi chuyển đổi tín hiệu giọng nói sang miền tần số, chúng ta phải chuyển nó thành dạng đặc trưng có thể sử dụng được. Chúng tôi có thể sử dụng các kỹ thuật trích xuất tính năng khác nhau như MFCC, PLP, PLP-RASTA, v.v. cho mục đích này.

Trong ví dụ sau, mình sẽ trích xuất các tính năng từ tín hiệu, từng bước, sử dụng Python, bằng cách sử dụng kỹ thuật MFCC.

**import** numpy as np

**import** matplotlib.pyplot as plt

from scipy.io **import** wavfile

from python\_speech\_features **import** mfcc, logfbank

Bây giờ, hãy đọc tệp âm thanh được lưu trữ. Nó sẽ trả về hai giá trị - tần số lấy mẫu và tín hiệu âm thanh. Cung cấp đường dẫn của tệp âm thanh nơi nó được lưu trữ.

frequency\_sampling, audio\_signal = wavfile.read("/Users/admin/audio\_file.wav")

Lưu ý rằng ở đây mình đang lấy 15000 mẫu đầu tiên để phân tích.

audio\_signal = audio\_signal[:15000]

Sử dụng các kỹ thuật của MFCC và thực hiện lệnh sau để trích xuất các tính năng của MFCC

features\_mfcc = mfcc(audio\_signal, frequency\_sampling)

In các tham số MFCC như sau :

print('\nMFCC:\nNumber of windows =', features\_mfcc.shape[0])

print('Length of each feature =', features\_mfcc.shape[1])

Vẽ features MFCC như sau :

**features\_mfcc** = features\_mfcc.T

plt.matshow(features\_mfcc)

plt.title('MFCC')

Ta sẽ làm việc với filter bank features , đầu tiên cần phải extract filtter bank features :

filterbank\_features = logfbank(audio\_signal, frequency\_sampling)

In các đối số filter bank

print('\nFilter bank:\nNumber of windows =', filterbank\_features.shape[0])

print('Length of each feature =', filterbank\_features.shape[1])

Vẽ và visualize filter bank features :

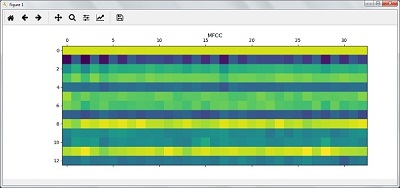
**filterbank\_features** = filterbank\_features.T

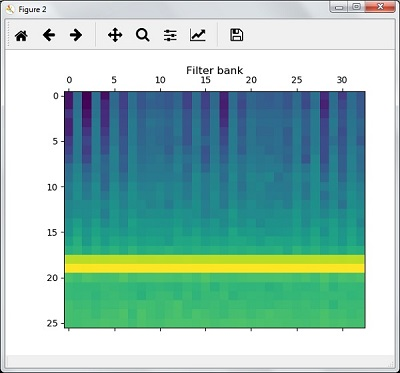
plt.matshow(filterbank\_features)

plt.title('Filter bank')

plt.show()

Theo kết quả của các bước trên, bạn có thể quan sát các kết quả đầu ra sau: Hình 1 cho MFCC và Hình 2 cho Filter Bank :





**6. Nhận biết các từ đã nói :**

Nhận dạng giọng nói có nghĩa là khi con người đang nói, một cỗ máy sẽ hiểu nó. Ở đây, mình đang sử dụng Google Speech API bằng Python để biến điều đó thành hiện thực

* **Pyaudio :** pip íntall Pyaudio
* **SpeechRecognition :**pip install SpeechRecognition.
* **Google-Speech-API :** pip install google-api-python-client

Quan sát ví dụ sau để hiểu về nhận dạng lời nói :

**import** speech\_recognition as sr

Tạo object như sau :

recording = sr.Recognizer()

Bây giờ, mô-đun Micrô () sẽ lấy giọng nói làm đầu vào :

**with** sr.Microphone() **as** **source**: recording.adjust\_for\_ambient\_noise(**source**)

print("Please Say something:")

audio = recording.listen(**source**)

Bây giờ Google API sẽ nhận dạng giọng nói và đưa ra đầu ra.

**try**:

print("You said: \n" + recording.recognize\_google(audio))

**except** Exception **as** e:

print(e)

Output như sau ;

Please Say Something:

You said:

[**BÀI TIẾP THEO: HEURISTIC SEARCH >>**](https://vncoder.vn/bai-hoc/heuristic-search-413)