

Tecnológico de Costa Rica.

Escuela de Ingeniería en Computación.

IC: 7602-Redes - 2 Semestre 2022.

2018086509 - Jocxan Sandi Batista.

DOCUMENTACIÓN - TAREA CORTA I - Autrum

Índice

- A. Funcionamiento de grabar audio y hacer las gráficas a la vez.
- B. Funcionamiento de analizar un audio existente.
- C. Funcionamiento de usar los archivos Autrum creados.
- D. Manual de usuario
- E. Pruebas Unitarias
- F. Módulos
- G. Referencias

A) Funcionamiento de grabar audio y hacer las gráficas a la vez.

El programa utiliza multi procesos ya que el modulo "**matplotlib**" no permite el uso de hilos.

Los procesos se inician en el método **grabar_graficar**. Además que se declaran cuatro colas para el manejo de banderas y de datos:

- q1: Guarda los datos del dominio del tiempo.
- q2: Sirve como bandera para terminar los procesos.
- q3: Sirve como bandera entre el proceso 1 y 2 es el que lleva el tiempo y permite el envío de datos.
- q4: Sirve como bandera para saber si se Graba o se Pausa.

Proceso 1

Se da con el método **Grabar2**. En este método inicia la grabación del audio por medio del micrófono, este proceso depende del proceso 2 que cada 3 segundo envía la señal a la cola para que se envíen los frames a través de la cola. Por último el método guarda todos los frames como un archivo de audio en un archivo llamado "Recorded.wav" y en un archivo ".txt" llamado **dominio_frecuencia.txt**.

Proceso 2

Se da con el método **Timer**. En este proceso se lleva la cuenta del tiempo a esperar para realizar cada gráfica. Envía la señal para que el proceso 1 envíe la información al proceso 3 cada **n** segundos ($n=3$).

Proceso 3

Se da con el método **grafica_tiempo_real**. En este método se realizan las gráficas del sonido en el dominio del tiempo y la Representación en Frecuencia.

Para esta parte se utilizaron dos fragmentos de código de diferentes fuentes

- La primera está basada en un código de Joser Zapata de: La parte que se utiliza aquí es para realizar el análisis de Fourier al sonido enviado desde el proceso 1.
- La segunda parte está basada en un código de Geeksforgeeks y modificada por Joxan Sandí: La parte que se utiliza aquí es para realizar la gráfica del dominio en el tiempo y la frecuencia de forma cíclica, ya que el proceso envía los datos cada 3 segundos, entonces se actualiza cada 3 segundos.

B) Funcionamiento de analizar un audio existente.

Se inicia en el método **audio_bash**. Método que recibe el nombre de un archivo ".wav" y realiza las gráficas en el dominio del tiempo y la frecuencia. También utiliza la base del código de Joser Zapata.

1. Solicita el nombre del archivo al usuario con el método **get_nombre_audio**.
2. Obtiene los datos del archivo con el módulo **scipy**.
3. Hace las gráficas con el método **graficas_parte_2**.
4. Escribe los archivos donde se guardan los datos del dominio del tiempo y la frecuencia.
5. Comprime los archivos en un archivo Autrum.

C) Funcionamiento de usar los archivos Autrum creados.

1. Solicita el nombre del archivo al usuario y los descomprime.
2. Lee los datos de los archivos del dominio del tiempo y la frecuencia.
3. Hace las gráficas con el método **graficas_parte_2**.

D) Manual de usuario.

Llamada: **py tarea1.py**

El programa funciona por la línea de comandos por lo que al inicio muestra las opciones que tiene el programa:

- 1. Para grabar inserte 1
- 2. Para ver el archivo Autrum inserte 2
- 3. Para usar un archivo de audio existente inserte 3
- 4. Para descomprimir un Autrum 4
- 5. Para salir inserte 5
- Inserte la opción que desee:

En este punto el usuario debe seleccionar lo que desee hacer con el programa.

1. Para grabar inserte 1

2. Para ver el archivo Autrum inserte 2

A continuación el sistema hace otra solicitud, en este caso el nombre del archivo atm de la siguiente forma:

- **Escriba el nombre del archivo atm para abrir, sin la extensión:**
- Al escribir el nombre y presionar Enter, el programa abre una ventana donde se muestran las gráficas.
- Al cerrar la ventana se termina el programa y vuelve al menú principal.

3. Para usar un archivo de audio existente inserte 3

A continuación el sistema hace otra solicitud, en este caso el nombre del archivo de audio de la siguiente forma:

- **Escriba el nombre del audio sin la extension:**
- Al escribir el nombre y presionar Enter, el programa abre una ventana donde se muestran las gráficas.
- **Escriba el nombre del archivo atm a guardar, sin la extension:** Esto es para guardar los datos en un archivo Autrum.
- Al cerrar la ventana se termina el programa y vuelve al menu principal.

4. Para descomprimir un Autrum 4

A continuación el sistema hace otra solicitud, en este caso el nombre del archivo ".atm" de la siguiente forma:

- **Escriba el nombre del archivo atm para abrir, sin la extension:**
- Vuelve al menu principal.

5. Para salir inserte 5

- Finaliza el programa

E) Pruebas Unitarias

Tabla de decisiones.

Condiciones	R1	R2	R3	R4	R5	R6
Inserte la opción que desee	1	2	3	4	5	Cualquier otra cosa
Acciones						
Grabar y gráficas	x					
Ver un archivo Autrum		x				
Usar archivo wav existente			x			
Descomprimir un Autrum				x		
Salir					x	
Volver a imprimir el menu						x

F) Módulos

Necesita los siguientes módulos:

- pyaudio 0.2.12
- wave 0.0.2

- scipy 1.9.1
- matplotlib 3.5.3
- numpy 1.22.3

G) Referencias

Obtener datos de un audio: <http://blog.espol.edu.ec/telg1001/audio-en-formato-wav/>

Representación en Frecuencia: https://joserzapata.github.io/courses/mineria-audio/representacion_audio/

Gráficas normal: https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.html

Gráficas en loop: <https://www.geeksforgeeks.org/how-to-update-a-plot-on-same-figure-during-the-loop/>

Comprimir Archivos: <https://code.tutsplus.com/es/tutorials/compressing-and-extracting-files-in-python--cms-26816>

Grabar y guardar: <https://www.thepythoncode.com/article/play-and-record-audio-sound-in-python>