

Evidencia NLP


Jose Edmundo Romo Castillo - A01197772

Descripción

La actividad presente se centra en utilizar un reconocimiento automático de voz pre entrenados que son esenciales para convertir archivos de audio en un texto que luego puede ser transcrito, esto permitiendo que los algoritmos de procesamiento del lenguaje natural revisen los datos que estos presentan. Este proyecto también cuenta con el uso de una API de la plataforma OpenAI, posteriormente se conectó a través de streamlit para la creación de una WebApp que pudiera mostrar no solo el audio que se utilizará como datos, sino también que mostraría el Transcript completo y un Summary de lo que se trato el audio.

WebApp

Con el apoyo de una API de OpenAI para utilizar GPT-3.5 Turbo como el algoritmo de inteligencia artificial.



The screenshot displays the 'Whisper Summarizer' web application. At the top, the title 'Whisper Summarizer' is shown in a large, bold, white font. Below it, the subtitle 'Análisis de Audio para Transcripción y Resumen Escrito' is displayed in a smaller white font. A white audio player bar is positioned below the subtitle, showing a progress bar at 0:00 / 8:25 and a play button. Underneath the audio player, the section 'Transcription:' is highlighted in a bold white font. The main content area features a transcription of an audio recording in a white font, discussing fitness and training schedules. The background of the entire interface is a solid dark blue.

Whisper Summarizer

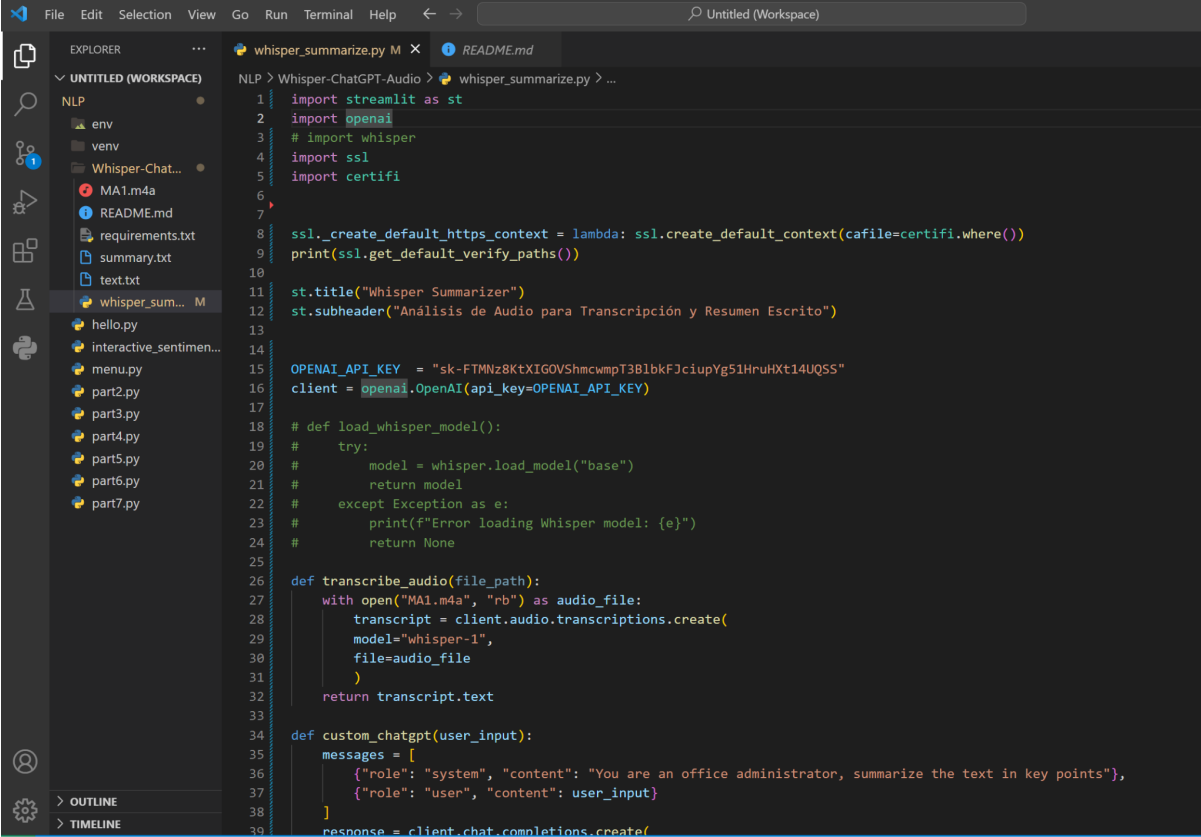
Análisis de Audio para Transcripción y Resumen Escrito

0:00 / 8:25

Transcription:

Especialmente, and I would even argue in the last four weeks, you really can't build fitness for the Ironman distance, but you can ruin your fitness for the Ironman distance, especially within the last two to three weeks, by, because, so it takes, it takes your body to adapt to a full load, a full Ironman load, it takes four to six weeks for your body to fully adapt to it, for you to go through the full depression cycle, all the way through the compensation cycle, and come back to the place where you're ready to race again. So if you take, if you take your body, and you're in that super compensation phase, and you start training in that, which is gonna be in these last two weeks, which you're in right now, and you add load to it, and drop down again, you're gonna lose everything that you gained from those big efforts that you did in the four to six weeks out from your race, okay? So you have to give your body time to rest, you need to let your body fully recover from those efforts, and even get stronger. So you can keep the sword sharp by doing short, quick efforts, you can get out the door to work out some of your, whatever your issues are, if they're mental, I need to go out and do something, fine, zone one, zone two, easy, 30, 45 minutes to at least open the body up, and remind yourself what you're gonna be doing, those things are fine. No weight training in the last two to four weeks, right now, you know, that does not belong in a program in the last two weeks for sure, I'd recommend usually even in the last four weeks, unless you've been super consistent with it, all the way through your, all the way through your training. Like not even squat, or push up, oops, yeah, stop, today, it's not worth it. And how about the volume, I mean, I look at my schedule, it's like 50% of my peak, you think it should be? That's appropriate, right? Yeah. Yeah. So you wanna be going from your peak volume, you should be cutting back to like 75%, four weeks out, two thirds, three weeks out, and then two weeks out, you should be almost half, half of your peak volume. Okay. Yeah. And then in the last week, it really depends, everybody's different, everybody tapers differently. And so you're probably not gonna get

Código



The image shows a Visual Studio Code editor window with a dark theme. The Explorer sidebar on the left displays a file tree for a project named 'NLP'. The file tree includes a folder 'UNTITLED (WORKSPACE)' containing subfolders 'env' and 'venv', and a file 'Whisper-Chat...'. Below this, a file named 'MA1.m4a' is listed with a red error icon. Other files include 'README.md', 'requirements.txt', 'summary.txt', 'text.txt', and a folder 'whisper_sum...' containing files 'hello.py', 'interactive_sentimen...', 'menu.py', 'part2.py', 'part3.py', 'part4.py', 'part5.py', 'part6.py', and 'part7.py'. The main editor area shows the 'whisper_summarize.py' file. The code is a Python script that imports 'streamlit as st', 'openai', 'ssl', and 'certifi'. It defines a lambda function for creating an SSL context, sets the title and subheader for a Streamlit app, and defines an OpenAI client. It includes a function to load the Whisper model and a function to transcribe audio from a file. The script also defines a custom chat function that uses the OpenAI API to summarize text.

```
1 import streamlit as st
2 import openai
3 # import whisper
4 import ssl
5 import certifi
6
7
8 ssl._create_default_https_context = lambda: ssl.create_default_context(cafile=certifi.where())
9 print(ssl.get_default_verify_paths())
10
11 st.title("Whisper Summarizer")
12 st.subheader("Análisis de Audio para Transcripción y Resumen Escrito")
13
14
15 OPENAI_API_KEY = "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"
16 client = openai.OpenAI(api_key=OPENAI_API_KEY)
17
18 # def load_whisper_model():
19 #     try:
20 #         model = whisper.load_model("base")
21 #         return model
22 #     except Exception as e:
23 #         print(f"Error loading Whisper model: {e}")
24 #         return None
25
26 def transcribe_audio(file_path):
27     with open("MA1.m4a", "rb") as audio_file:
28         transcript = client.audio.transcriptions.create(
29             model="whisper-1",
30             file=audio_file
31         )
32     return transcript.text
33
34 def custom_chatgpt(user_input):
35     messages = [
36         {"role": "system", "content": "You are an office administrator, summarize the text in key points"},
37         {"role": "user", "content": user_input}
38     ]
39     response = client.chat.completions.create(
```