**Proyecto Promp**

**de control por voz basada en Inteligencia Artificial**

**Escuela Colombiana De Ingeniería Electrónica**

**Carlos Andrés Buitrago**

**1.Objetivo Objetivo**: El propósito fundamental de este proyecto es concebir, diseñar y materializar una aplicación de control por voz basada en Inteligencia Artificial, con funcionalidades análogas a las ofrecidas por asistentes virtuales líderes como Alexa. El enfoque integral abarcará todas las fases del desarrollo, desde la creación del dataset hasta la implementación en un Sistema en Chip (SOC) específicamente la ESP32.

**2.Objetivos Específicos** :

* **Fase 1: Creación del Dataset**

En esta etapa inicial, se recopiló y preparó un dataset exhaustivo que abarca diversas entradas de voz y comandos de usuarios. La calidad y la diversidad del dataset son esenciales para garantizar la robustez y la capacidad de adaptación del modelo de inteligencia artificial.

* **Fase 2: Generación del Modelo de Entrenamiento**

Se procedió a la implementación del modelo de inteligencia artificial, utilizando técnicas avanzadas de aprendizaje profundo y procesamiento de lenguaje natural. Durante este proceso, se prestaron especial atención a la eficiencia y a la optimización del modelo para adaptarse a las restricciones de recursos de la ESP32.

* **Fase 3: Implementación en la ESP32 (SOC)**

La implementación en el Sistema en Chip (SOC) de la ESP32 marca la fase actual del proyecto. Aquí, se están llevando a cabo las siguientes actividades:

• Preparación del Entorno de Desarrollo: Configuración adecuada del entorno de desarrollo de Arduino con soporte para la ESP32 y la instalación de herramientas necesarias.

Procedimiento: Procedimiento Mejorado para la Primera Fase: Creación del Dataset con Modelos de Hugging Face y ChatGPT

**3.Procedimiento:**

* **Fase 1**

**Generación de Audio con Modelos de Hugging Face**:

Utilizar modelos de texto a voz (TTS) de Hugging Face para generar archivos de audio a partir del texto ingresado.

Almacenar los archivos de audio en un formato estándar como WAV o MP3 para facilitar la manipulación y reducir problemas de compresión posterior.

**Generación de Audios para el Dataset:**

Combinar los archivos de audio generados por Hugging Face con las respuestas de ChatGPT, asegurando la homogeneidad de la palabra clave.

Incorporar un módulo de limpieza para eliminar posibles ruidos no deseados o inconsistencias en los archivos de audio.

**Organización y Etiquetado del Dataset:**

Estructurar el dataset de manera ordenada, utilizando una nomenclatura clara para los archivos y directorios.

Incluir un archivo de metadatos que detalle las palabras clave, las variaciones generadas y cualquier información relevante.

**Compresión del Dataset**:Utilizar un formato de compresión estándar como ZIP o TAR para empaquetar el dataset completo.

Verificar que la compresión no cause pérdida de calidad en los archivos de audio.

**Descompresión y Verificación**:Implementar un proceso de descompresión en Colab que garantice la integridad de los archivos y la estructura del dataset.

Realizar una verificación posterior a la descompresión para asegurar que todos los archivos estén presentes y en buen estado.

**Enlace con Google Drive**:Utilizar la API de Google Drive en Colab para facilitar la transferencia de archivos entre Colab y Google Drive.

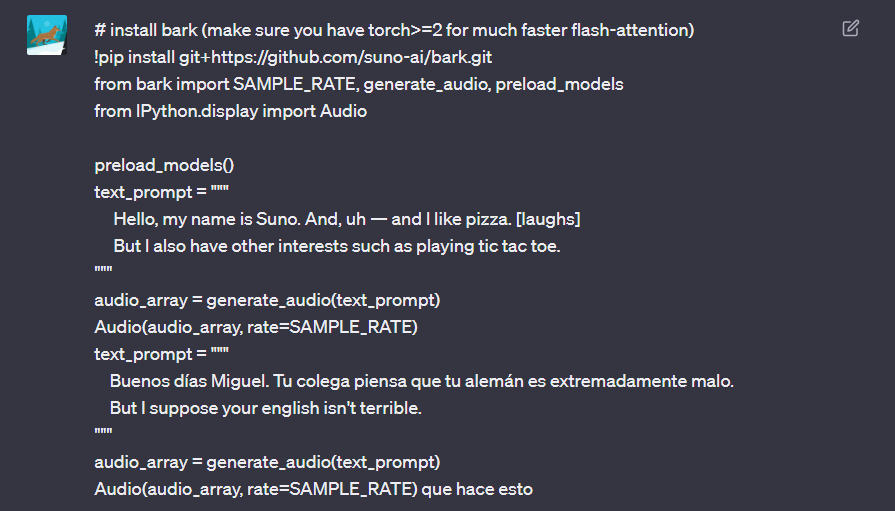
Establecer un flujo de trabajo claro para cargar los archivos comprimidos a Google Drive y descargarlos posteriormente cuando sea necesario.

**Manejo de Problemas de Compresión**:

Identificar y abordar problemas específicos de compresión, como pérdida de calidad, archivos dañados o tamaño excesivo del archivo comprimido.

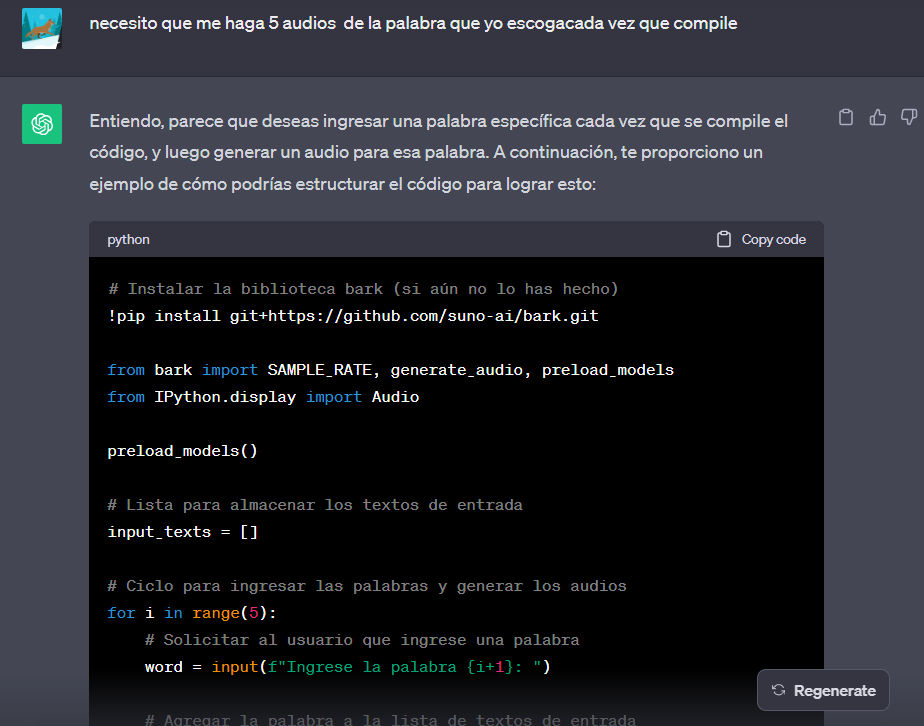
Experimentar con diferentes configuraciones de compresión para optimizar el equilibrio entre tamaño del archivo y calidad de los datos.

**Promps usados para llegar al resultado**:



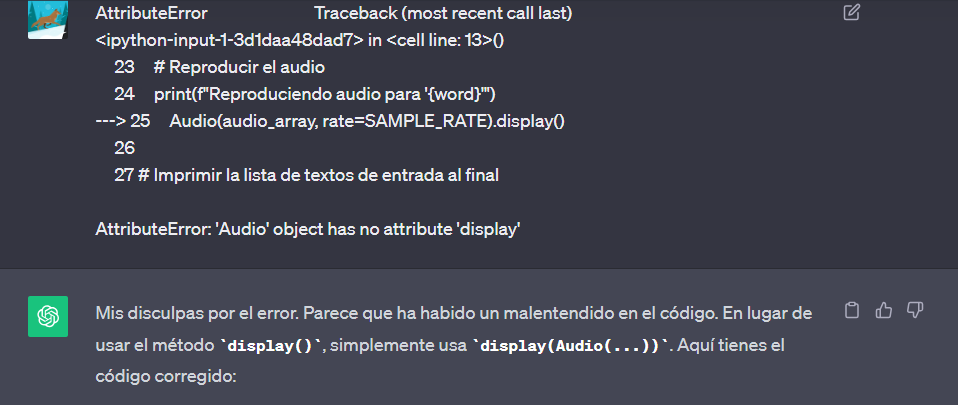
Una captura de pantalla de una red social

Descripción generada automáticamente



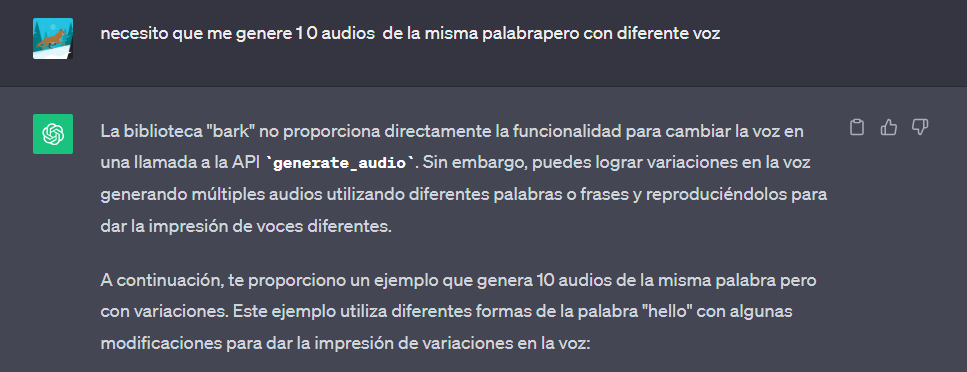
Texto

Descripción generada automáticamente



Texto

Descripción generada automáticamente



Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico, Sitio web

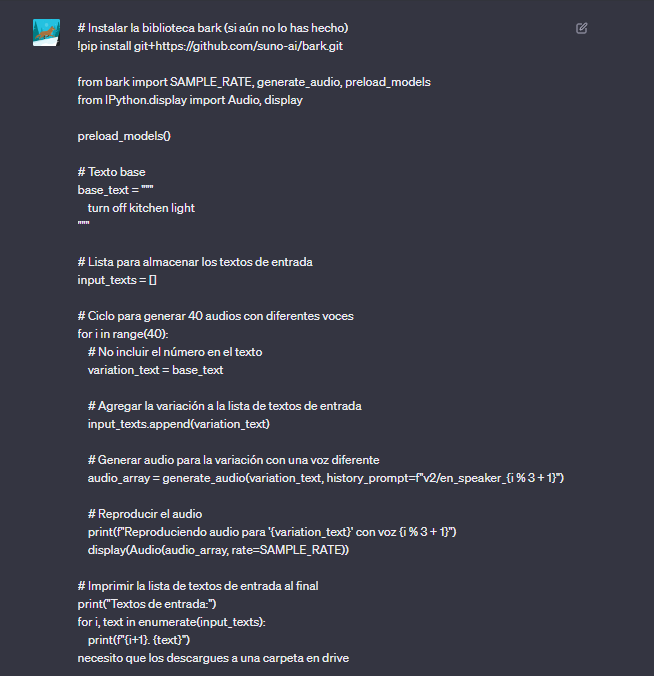
Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente



Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente



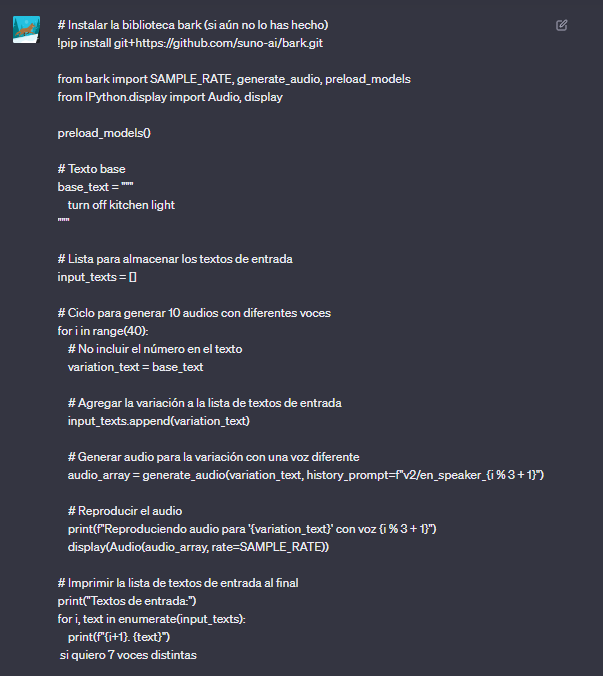
Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente



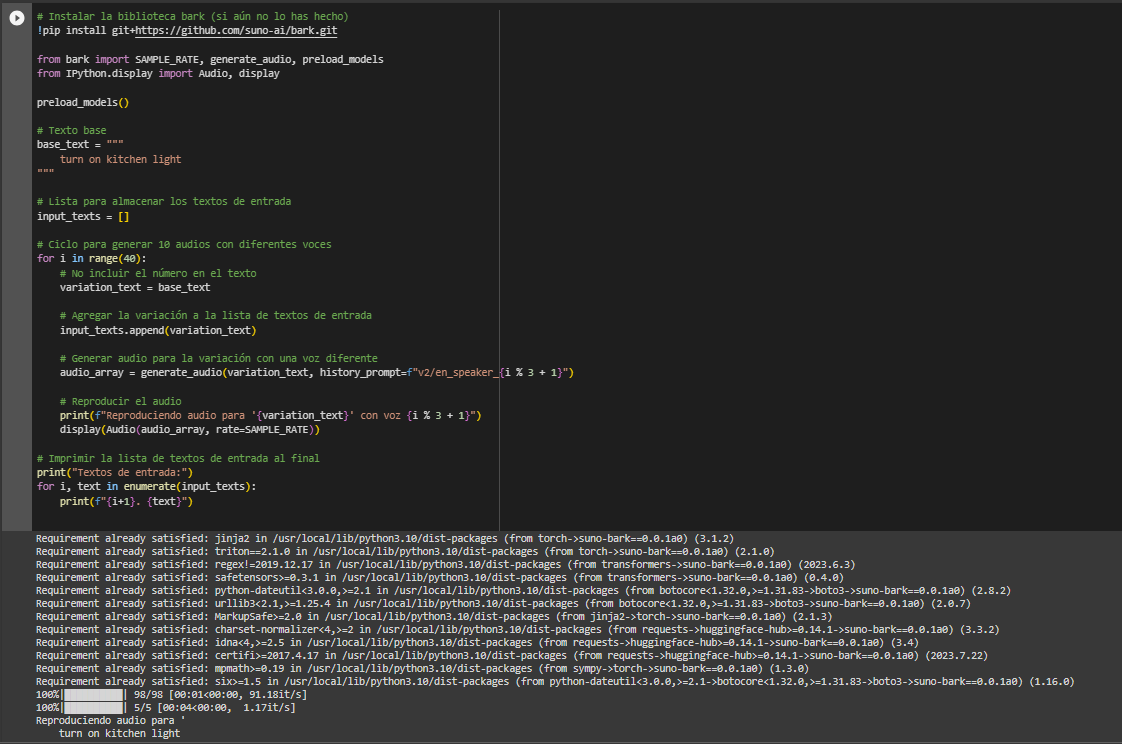


Texto

Descripción generada automáticamente

Con estos promps se logró crear un código aceptable para lograr un dataset para la nuestra aplicación de control por voz

Resultado :

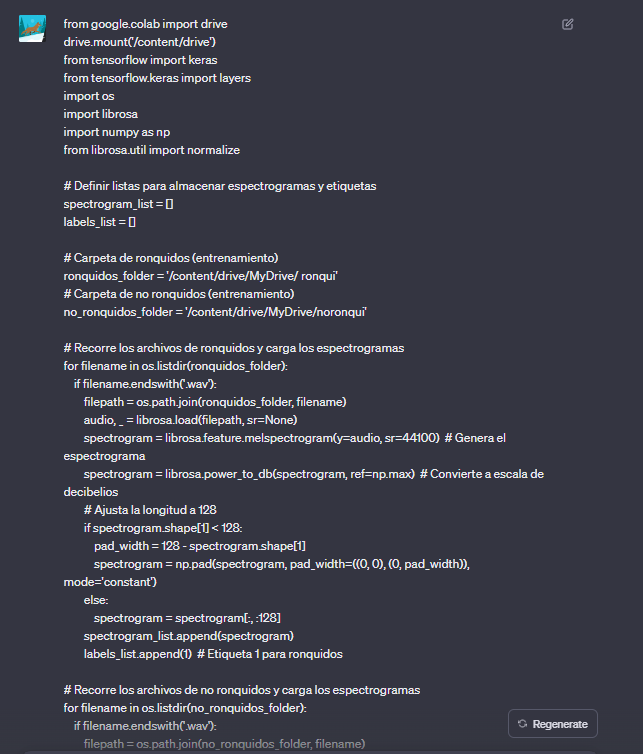


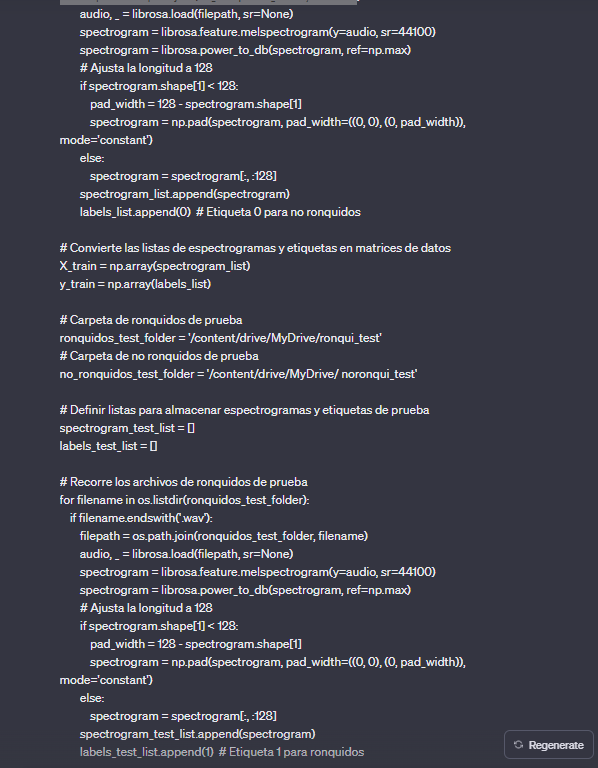
Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

* Fase 2:

El enfoque adoptado para el entrenamiento del modelo se fundamentó en la utilización de espectrogramas derivados de los audios. Este proceso permitió abordar el problema como una tarea de procesamiento de imágenes. Inicialmente, se partió de un código existente, el cual fue modificado y personalizado conforme a las particularidades del proyecto. La adaptación y refinamiento de este código fueron facilitados mediante la colaboración con ChatGPT para obtener orientación específica.





Texto

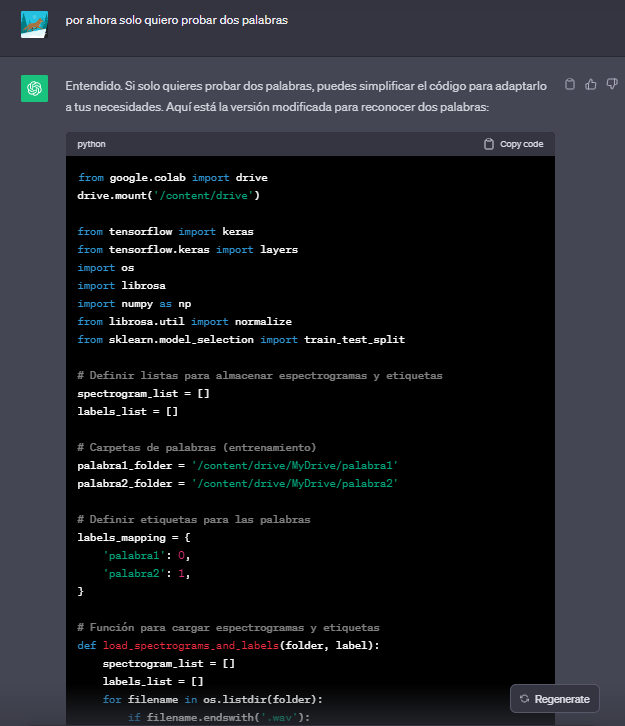
Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente



Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

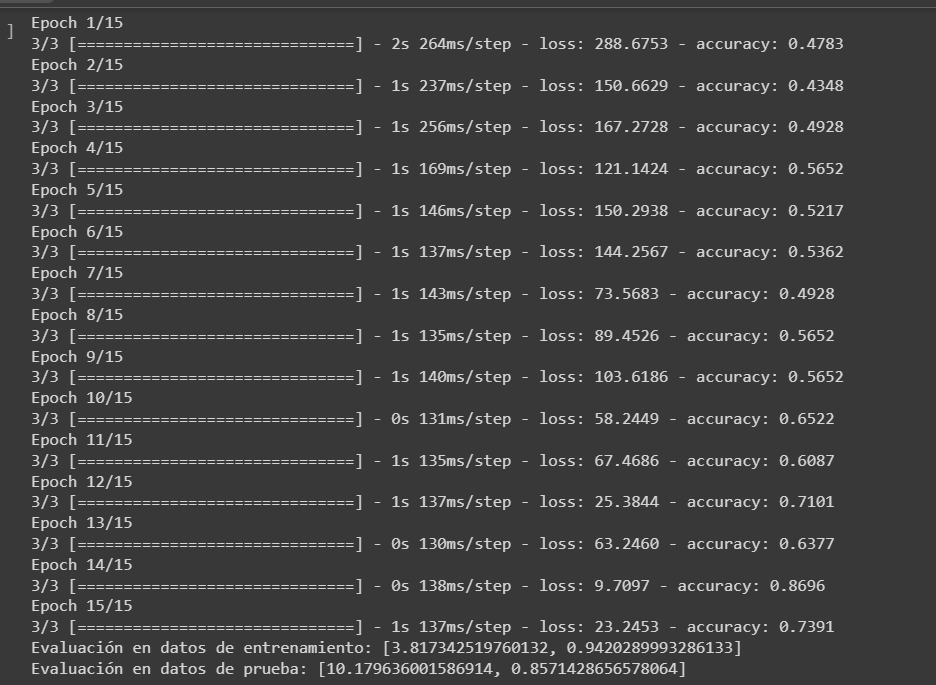
Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

* Resultado del código:



En cada "Epoch", el modelo pasa por el conjunto de datos de entrenamiento y ajusta sus pesos y parámetros para mejorar su rendimiento en una tarea específica. Cada línea contiene información sobre la pérdida (loss) y la precisión (accuracy) del modelo en un paso de entrenamiento específico

* "Epoch 1/15" indica que el entrenamiento se divide en 15 épocas (o iteraciones completas a través del conjunto de datos).
* "2/2" se refiere al número de lotes (batches) de datos que se procesaron en esa época. En este caso, parece que se procesaron 2 lotes de datos.
* "loss" es la función de pérdida, que mide cuán lejos está el modelo de predecir el resultado correcto. En la primera época, la pérdida es 157.4106, lo que significa que el modelo tiene un alto error en sus predicciones en ese momento.
* "accuracy" es la precisión del modelo en ese paso de entrenamiento. En la primera época, la precisión es 0.4833, lo que significa que el modelo tiene una precisión del 48.33 porciento en sus predicciones en ese momento.
* El registro continúa a lo largo de las 15 épocas, y puedes observar cómo la pérdida disminuye y la precisión aumenta a medida que el modelo se entrena. Al final, se proporciona una "Evaluación del modelo" que incluye una pérdida de 1.6156 y una precisión de aproximada- mente 0.7857 en un conjunto de datos de evaluación. Esto indica que el modelo ha mejorado su rendimiento durante el entrenamiento y tiene una precisión del 78.57 porciento en la tarea en la que fue entrenado.

Texto

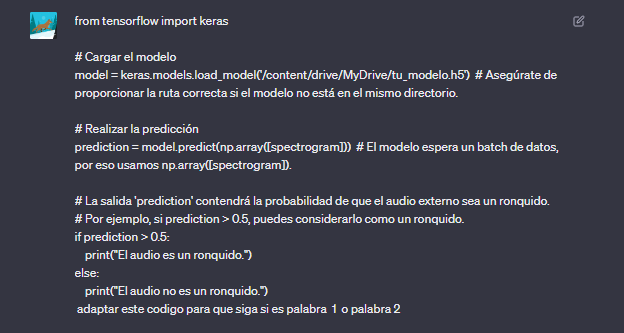
Descripción generada automáticamente

Texto

Descripción generada automáticamente

la arquitectura de una red neuronal convolucional (CNN) en Keras, una biblioteca de aprendizaje profundo

* Model: Esto indica que se está describiendo la arquitec- tura de un modelo de red neuronal.
* Layer (type): Esta sección muestra el tipo de capa en la red y sus configuraciones. En este caso, la capa es una capa convolucional (Conv2D) seguida de una capa de agrupación máxima (MaxPooling2D).
* Output Shape: Indica las dimensiones de la salida de la capa. Por ejemplo, "(None, 126, 126, 32)" significa que la capa de convolución produce un tensor tridimensional de forma 126x126x32.
* Param : Representa el número de parámetros en la capa. En este caso, la capa Conv2D tiene 320 parámetros y la capa Dense tiene 127,009 parámetros. Los parámetros son los valores que la red neuronal aprende durante el proceso de entrenamiento.
* Total params: Muestra el número total de parámetros en el modelo, que en este caso es 127,329.
* Trainable params: Indica el número de parámetros que se pueden ajustar durante el entrenamiento de la red.
* Non-trainable params: Muestra el número



Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

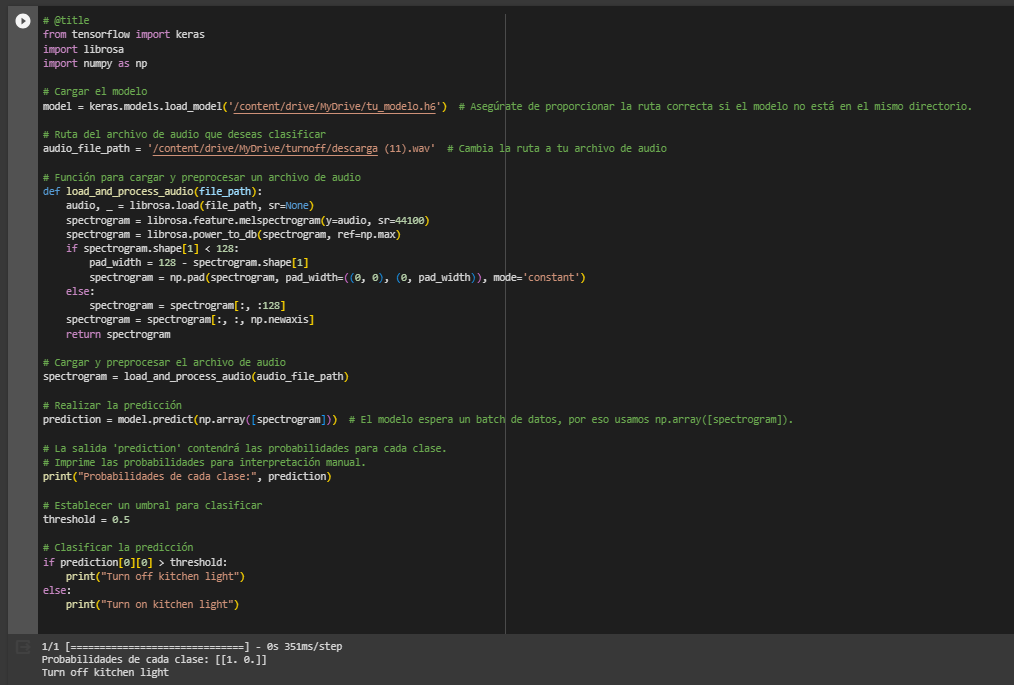
Texto

Descripción generada automáticamente

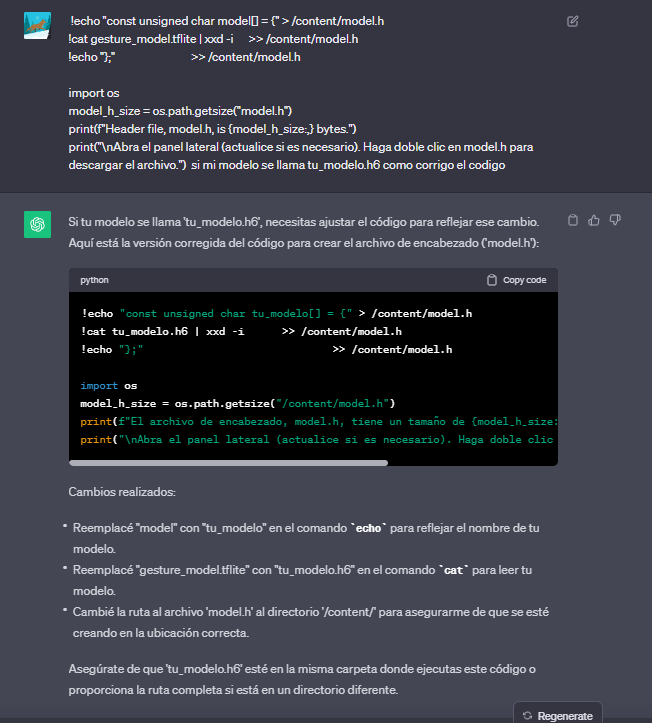
Texto

Descripción generada automáticamente

Resultado:



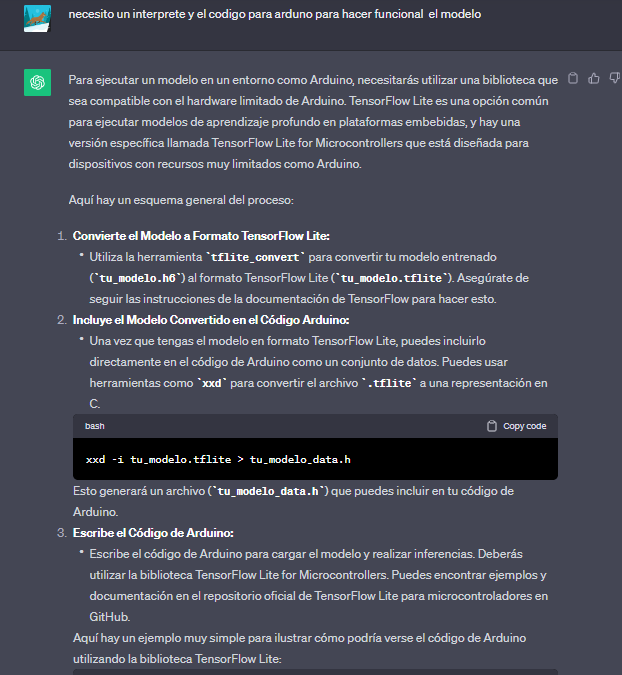
* Fase 3: La fase culminante del proyecto se centra en la transición exitosa del modelo estadístico entrenado hacia una implementación práctica utilizando TensorFlow Lite en el entorno de programación C++.



Resultado

Texto

Descripción generada automáticamente



Texto

Descripción generada automáticamente

Esta ultima fase no esta Funcional ya que faltan varias pruebas

**4. Conclusiones:**

Este proyecto ha sido ejecutado de manera integral con el respaldo fundamental de herramientas de inteligencia artificial como ChatGPT y Hugging Face. A través de esta experiencia, se ha evidenciado claramente que la inteligencia artificial es una herramienta sumamente valiosa, y su eficacia se maximiza cuando se formula las preguntas adecuadas. Algunas conclusiones destacadas incluyen:

* Colaboración Esencial con la Inteligencia Artificial:

La colaboración activa con herramientas como ChatGPT y Hugging Face ha sido esencial para orientar y personalizar el proyecto en todas sus fases, desde la concepción hasta la implementación.

* Eficiencia en la Generación y Adaptación de Códigos:

La inteligencia artificial ha demostrado su eficiencia en la generación de datos, la adaptación de códigos existentes y la resolución de desafíos específicos, acelerando significativamente el proceso de desarrollo.

* Poder de la Formulación de Preguntas:

La habilidad humana de formular preguntas pertinentes y específicas a la inteligencia artificial ha sido clave para obtener respuestas útiles y adaptar las herramientas según las necesidades del proyecto.

* Demostración de Utilidad en Diversas Fases:

La utilidad de la inteligencia artificial se ha demostrado a lo largo de todas las etapas del proyecto, desde la creación del dataset hasta la implementación en la plataforma hardware, destacando su versatilidad.

**Lecciones Aprendidas**:

* Sinergia entre Habilidades Humanas e Inteligencia Artificial: La combinación de la intuición y la creatividad humana con las capacidades de la inteligencia artificial genera resultados más potentes y adaptados a necesidades específicas.
* Aprovechamiento Eficiente de Herramientas Existentes: La capacidad para integrar y aprovechar herramientas preexistentes, como ChatGPT y Hugging Face, demuestra la importancia de la colaboración entre humanos y máquinas.
* Aplicación en Proyectos Futuros: La experiencia obtenida indica el potencial de replicar y ampliar este enfoque en futuros proyectos, aprovechando la eficiencia y versatilidad de la inteligencia artificial.

Este proyecto, concebido y ejecutado con el apoyo integral de la inteligencia artificial, subraya la importancia de formular preguntas adecuadas para maximizar el potencial de estas herramientas. La combinación de habilidades humanas y capacidades de la inteligencia artificial ha resultado en un desarrollo exitoso y eficiente.