Introducción a Docker  
UD 06. Caso práctico 05 - Bot de Telegram con Whisper AI

short line

Autor: Sergi García Barea

Actualizado Marzo 2023

Licencia

**Reconocimiento – NoComercial - CompartirIgual (BY-NC-SA)**: No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

📖 **Importante**

❕ **Atención**

💬 **Interesante**

[**1. Introducción 3**](#_dbh0n1vac4c8)

[**2. Pasos a previos para poder generar un bot de Telegram y obtener Ids de usuarios o grupos 3**](#_vyhbfp4t666x)

[**3. Poner en marcha bot de Telegram con Whisper AI 3**](#_orgfdxf9if1u)

[**4. Bibliografía 4**](#_72yv1gty1dke)

UD06. Caso práctico 05

# Introducción

Whisper AI es un software que permite transcribir audio a Texto. Este software fue creado por OpenAI (empresa creadora de ChatGPT) y liberado junto con varios modelos de procesamiento de lenguaje muy precisos. Actualmente, es una de las mejores soluciones “self-hosted” para convertir audio a texto.

Whisper CPP es una implementación de Whisper AI en C++ obteniendo mucho mejor rendimiento que la implementación original <https://github.com/ggerganov/whisper.cpp>

Utilizando esta última implementación, presentamos un bot de Telegram (creado por el autor del curso) que permite recibir mensajes de voz/audios (directamente al bot o en grupos donde el bot pueda leer) y transcribirlos. El software para ello está disponible en <https://github.com/sergarb1/telegram-bot-whisper-cpp>

# Pasos a previos para poder generar un bot de Telegram y obtener Ids de usuarios o grupos

En primer lugar, debes generar un nuevo bot de Telegram. Hay información de como hacerlo en Generar un nuevo bot en <https://atareao.es/tutorial/crea-tu-propio-bot-para-telegram/>

A grandes rasgos, los pasos son:

* Inicia una nueva conversación con el bot @BotFather.
* Escribe el comando /newbot. Establece un nombre y un nombre de usuario.
* Copia el Token que BotFather te dará.

También, para el siguiente paso, deberás obtener las IDs de usuarios o grupos permitidos. Se explica en detalle en <https://www.alphr.com/find-chat-id-telegram/>

A grandes rasgos, los pasos para obtener el ID de los usuarios/grupos permitidos:

* Inicia una nueva conversación con @RawDataBot.
* Envía un mensaje o reenvía un mensaje de cualquier usuario deseado de obtener el ID.
  + Si lo que se quiere es obtener el ID de un grupo, se debe meter al @RawDataBot en el grupo.
* Copia el valor del campo message.from.id.
* Debería ser un ID como:
  + Usuarios: 1234567890
  + Grupos (con un menos delante): -1234567890

# Poner en marcha bot de Telegram con Whisper AI

Obtenemos el repositorio con el comando (en el curso os lo proporcionamos como un “.zip”):

| git clone https://github.com/sergarb1/telegram-bot-whisper-cpp |
| --- |

Tras ello debemos editar el fichero “.env” de la siguiente forma:

* Establece tu TELEGRAM\_BOT\_TOKEN.
* Establece tus ALLOWED\_CHAT\_IDS (ID de usuarios/grupos separados por comas).
  + Establécelo en \* para permitir a todos los usuarios.
* Establece el WHISPER\_MODEL (base, tiny, small, medium, large).
  + Recomendado usar como máximo el “small” para estas pruebas.
* Establece AUDIO\_LANGUAGE (Si lo dejas en auto detectará el idioma).

Una vez realizadas estas operaciones, simplemente ejecuta:

| docker compose up --build -d |
| --- |

Y tendrás el bot funcionando perfectamente.

Para transcribir un audio, solo deberás:

* Enviarle una nota de voz/fichero de audio al bot desde un usuario autorizado.
* Mandar una nota de audio a un grupo donde esté el bot, tenga permiso de lectura del grupo y además sea un grupo autorizado. Cuidado, en este caso, la transcripción será enviada a todo el grupo.

# Bibliografía

[1] Docker Docs <https://docs.docker.com/>

[2] Docker Compose Docs <https://docs.docker.com/compose/>