



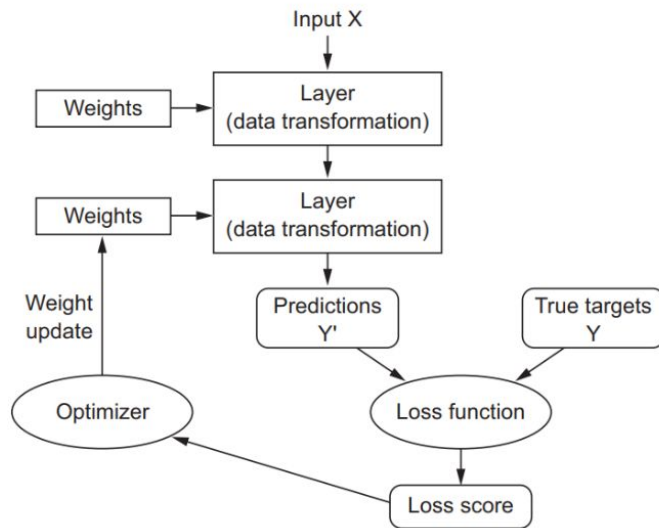
Análisis de Señales y Sistemas Digitales

Grupo 3 - 2022

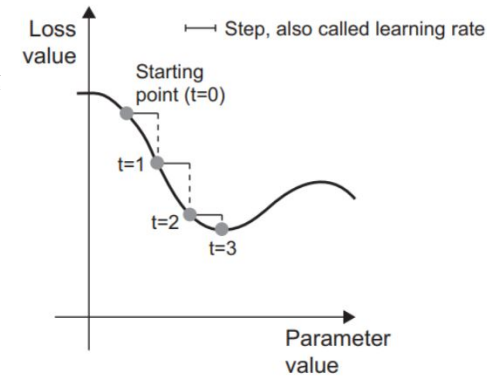
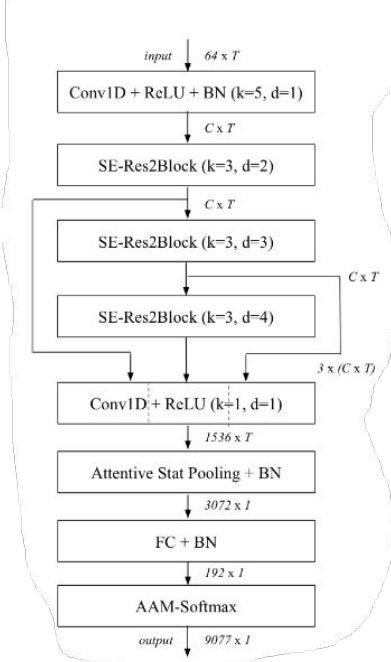
Recordemos el TP4....

Verificación por voz utilizando redes neuronales

Conocimientos previos: Esquema de una red neuronal

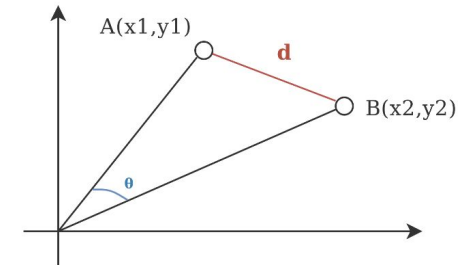


Neural Networks: ECAPA-TDNN



Cosine Similarity

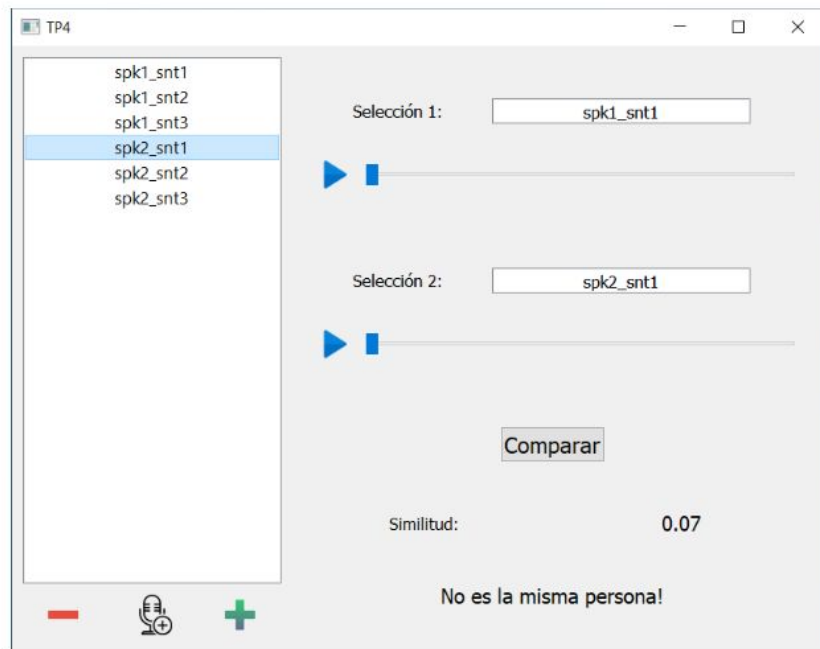
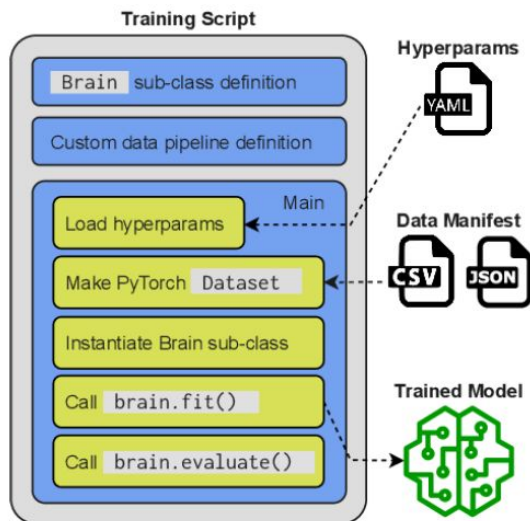
$$CDF(v_1, v_2) = \frac{v_1 \cdot v_2}{\|v_1\| \|v_2\|}$$



Recordemos el TP4....

Verificación por voz utilizando redes neuronales

Arquitectura



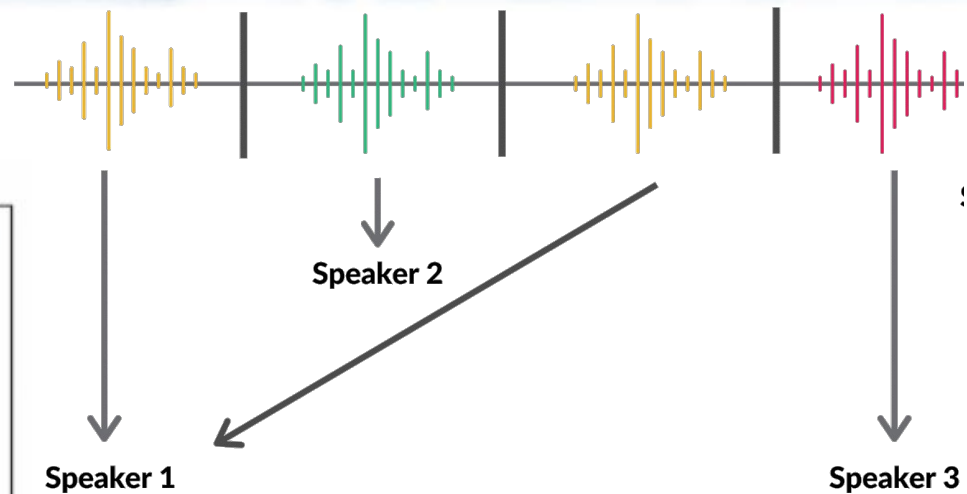
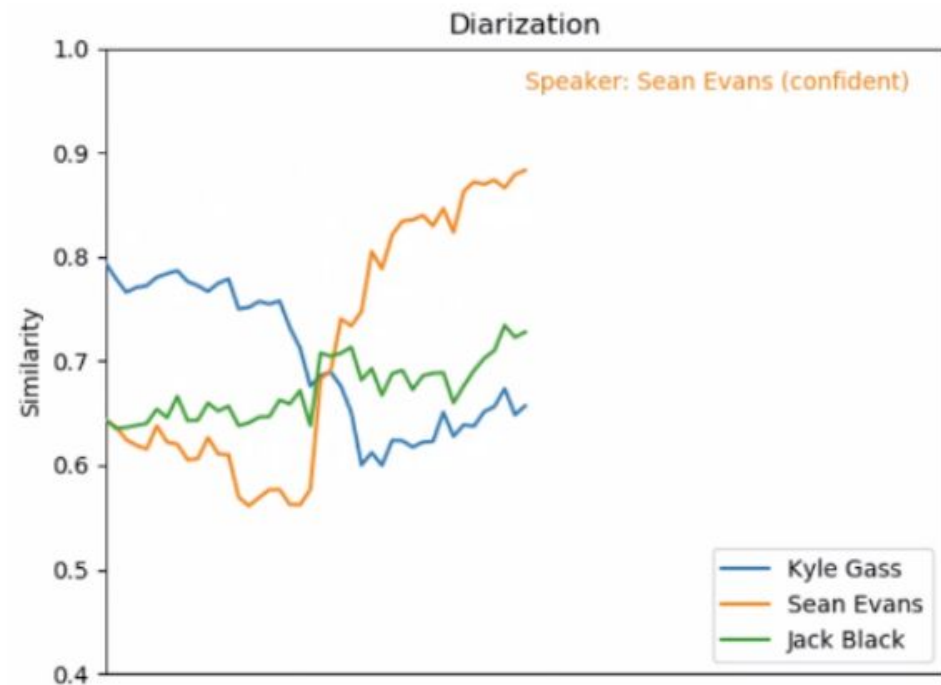
Recordemos el TP4....

¿ Y si no solo reconocemos que una voz sea correcta,
sino que también reconocemos a quien pertenece en
una conversación?

¿ Y si además transcribimos esa conversación?

Diarization + Speech to text

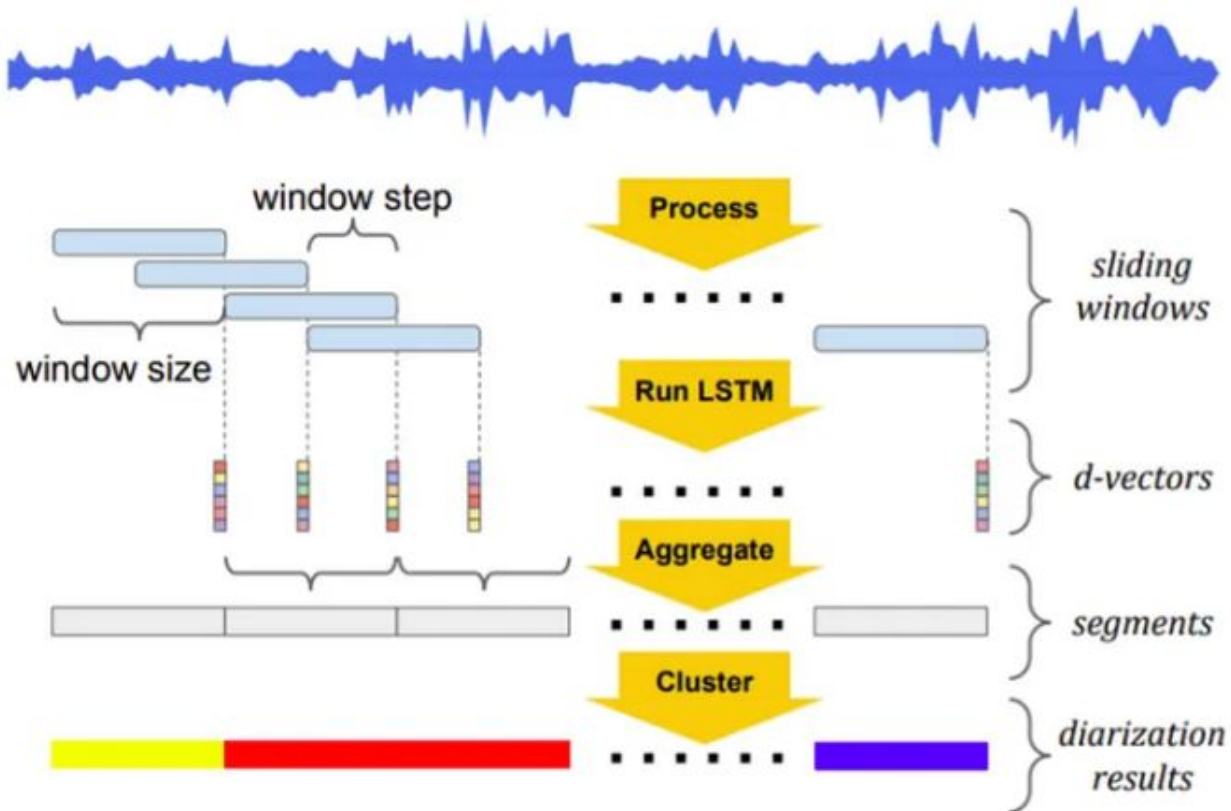
Diarization



Diarization

- **Speech Detection:** VAD (Voice Activity Detector) para separar palabras de silencio.
- **Speech Segmentation:** extraer segmentos que contengan a las diferentes personas que hablan. Se elige la duración de la ventana y el overlap.
- **Embedding Extraction:** se crea una red neuronal basada en los segmentos extraídos anteriormente. Un embedding es una representación vectorial de data de audio que utilizaremos (d-vector).
- **Clustering:** luego de crear los embeddings de los segmentos, se los agrupa por personas que hablan. Aquí también se colocan los labels correspondientes, y se indica la cantidad de personas que participan.

Diarization



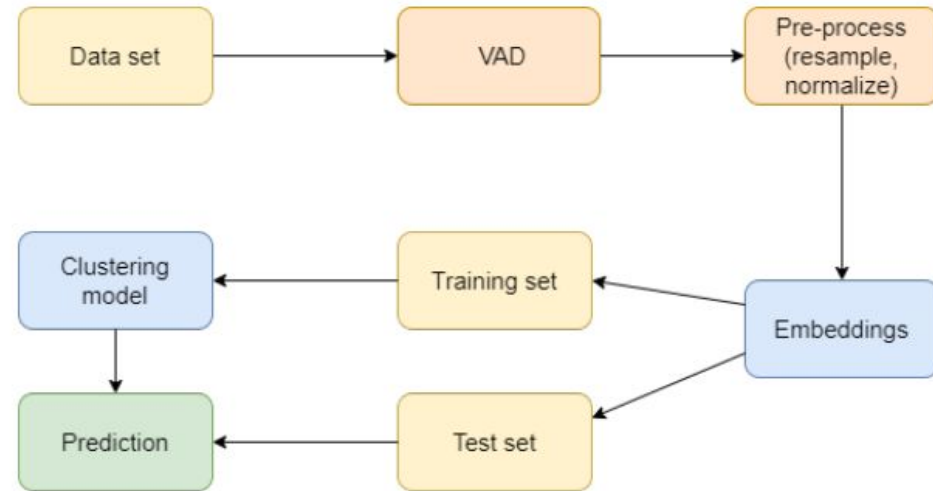
Diarization: Resemblyzer



- Ingresamos un audio 'audio.wav'.
- Una **VAD** interno para quitar las partes de silencio.
- Crea una instancia de **VoiceEncoder**.
- Se llama a la función **embed_utterance**, la cual segmenta el audio en ventanas, con un determinado overlap. La data del audio es **sampleada a 16KHz**. Finalmente se crea el d-vector. Cada d-vector corresponde a una ventana.

Diarization: Resemblyzer

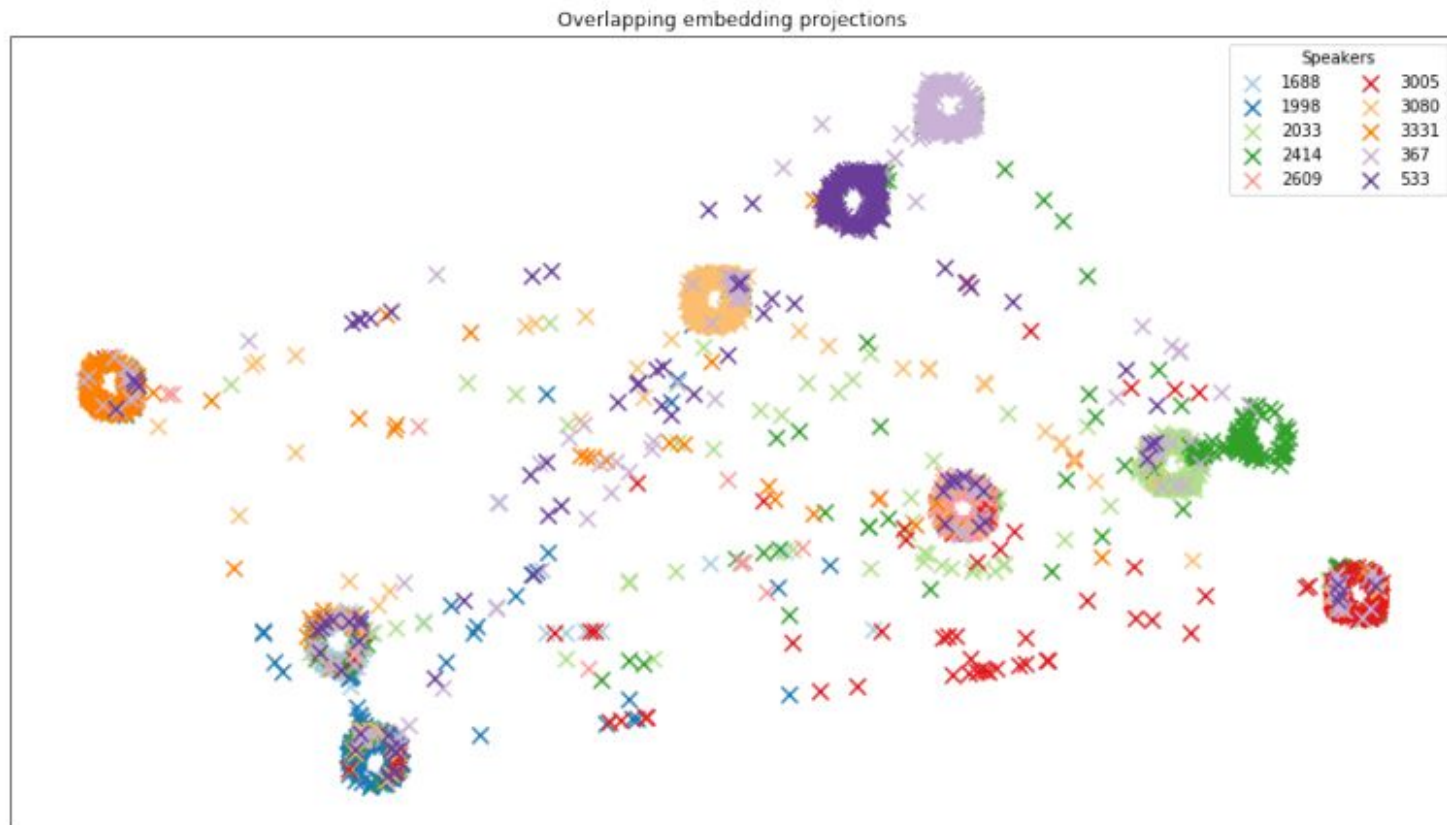
- Ingresamos un audio 'audio.wav'.
- Una un **VAD** interno para quitar las partes de silencio.
- Crea una instancia de **VoiceEncoder**.
- Se llama a la función **embed_utterance**, la cual segmenta el audio en ventanas, con un determinado overlap.
- A la **red** (supervisada) ya entrenada con muestras de audio, le ingresamos nuestros audios y los labels correspondientes. De este forma identificar quién está hablando (probabilidad).



Diarization: Resemblyzer: Dataset de entrenamiento



Diarization: Resemblyzer

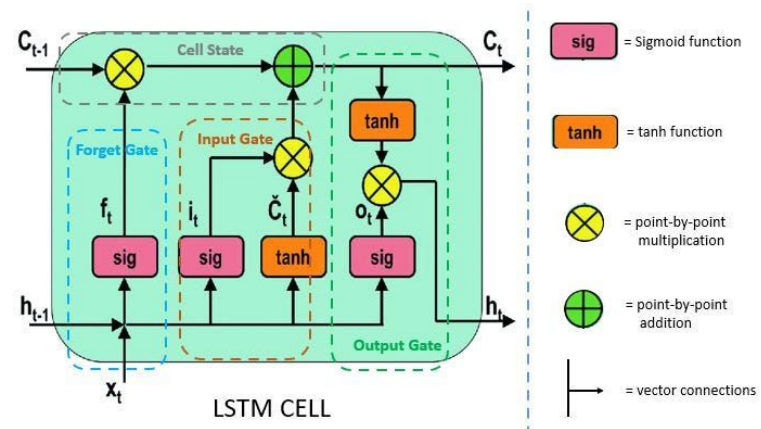
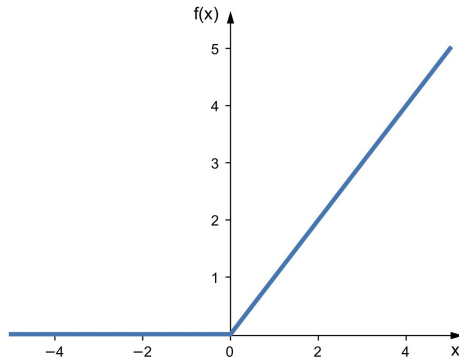


Diarization: Red Neuronal

- Se utilizaron 5, 600 speakers para el entrenamiento.
- El tipo de red es una LSTM (Long short-term memory) de 3 capas.

“Una Red LSTM es capaz de “recordar” un dato relevante en la secuencia y de preservarlo por varios instantes de tiempo. Por tanto, puede tener una memoria tanto de corto plazo (como las Redes Recurrentes básicas) como también de largo plazo”

- Utiliza como función de activación a la ReLU.



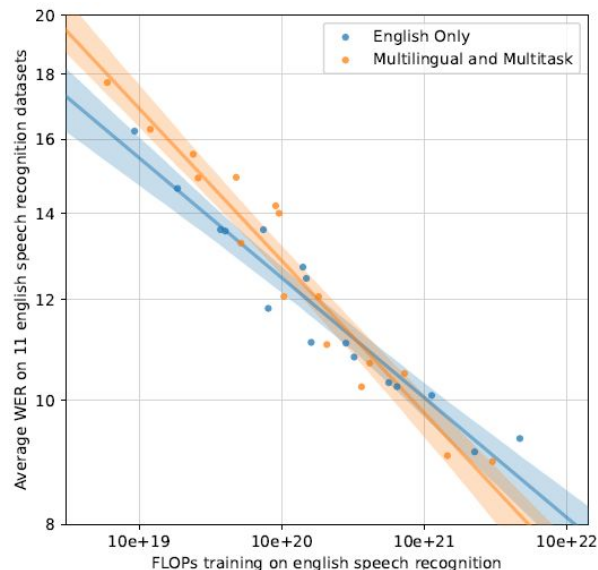
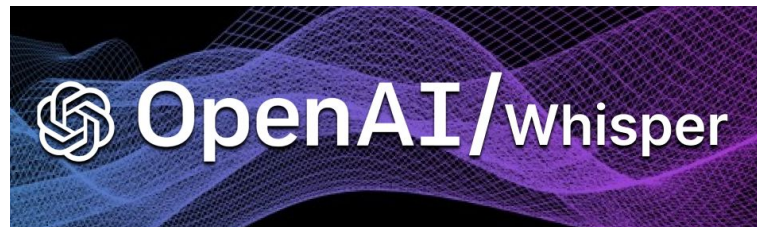
Diarization: procesamiento de datos

Una vez utilizada la red obtendremos probabilidades para cada speaker en cada instante. Luego se procesarán los datos:

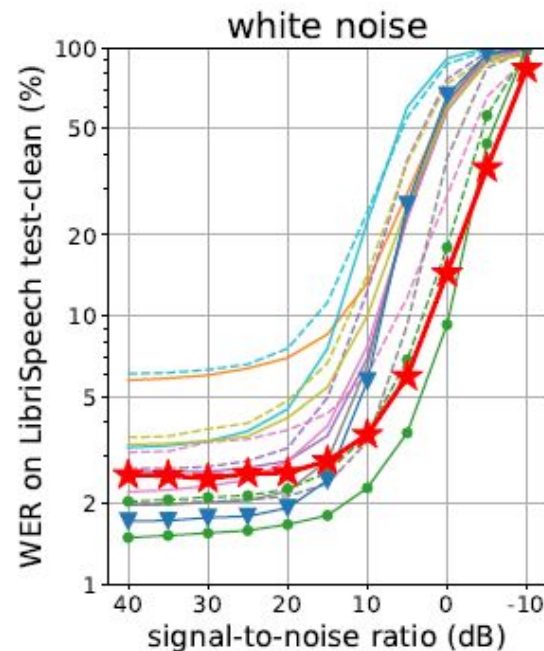
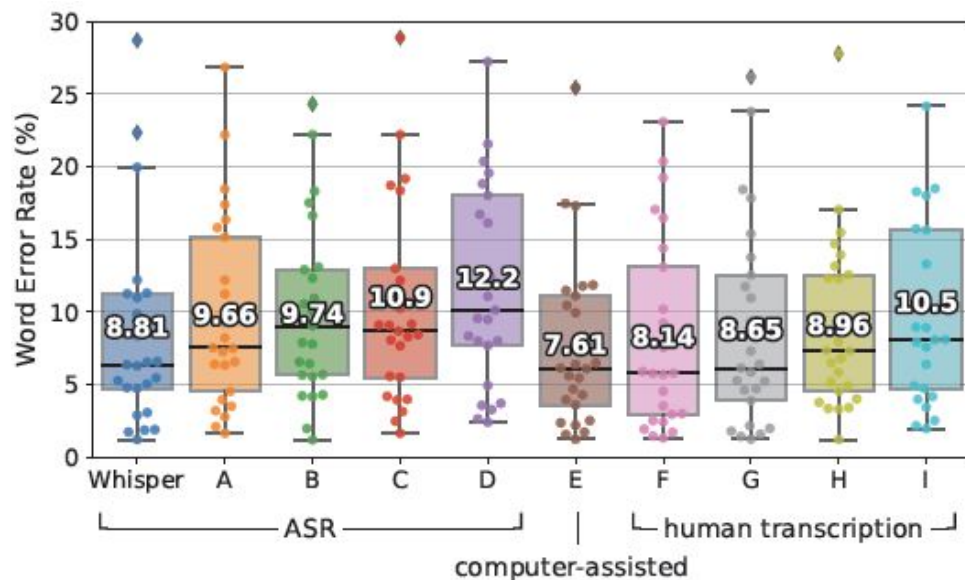
- Determinar el speaker predominante.
- Determinar umbral de ruido ambiente.
- Realizar correcciones leves (Si obtuve 1 valor de 5 diferente, corregirlo).
- Agrupar segmentos con el formato $[t_start, t_end, speaker, speech]$.
- Si $t < \Delta t$: eliminar segmento.
- Si hay 2 segmentos con $t_start_2 < t_end_1 + \Delta t$: unificar los segmentos.

Speech2Text: Whisper

- Librería Open Source
 - Acceso al código.
 - Acceso al paper de referencia.
- Innovaciones en el entrenamiento y la arquitectura de la red.
- Múltiples Lenguajes.
- Múltiples Funciones.



Speech2Text: Whisper

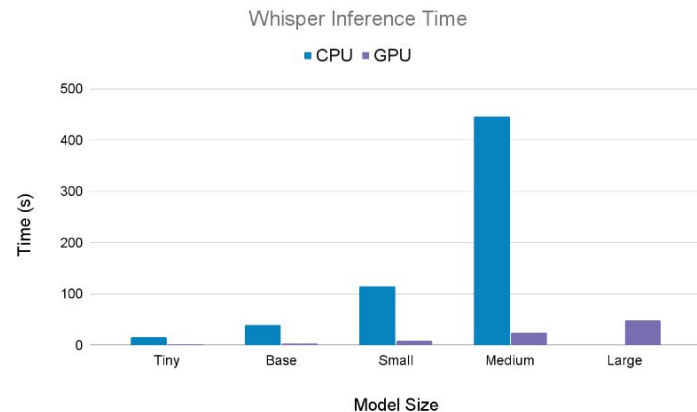


Speech2Text: Entrenamiento

- Entrenamiento Multi-Tasking y Multi-Lenguaje.
- > 650.000 horas de audio de entrenamiento.
- Datasets con diferentes características.
- Evaluación sobre dataset no utilizados (Zero-Shot)
- Modelos de varias complejidades
 - Distintos desempeños.
 - Distintos tiempos de procesamiento.

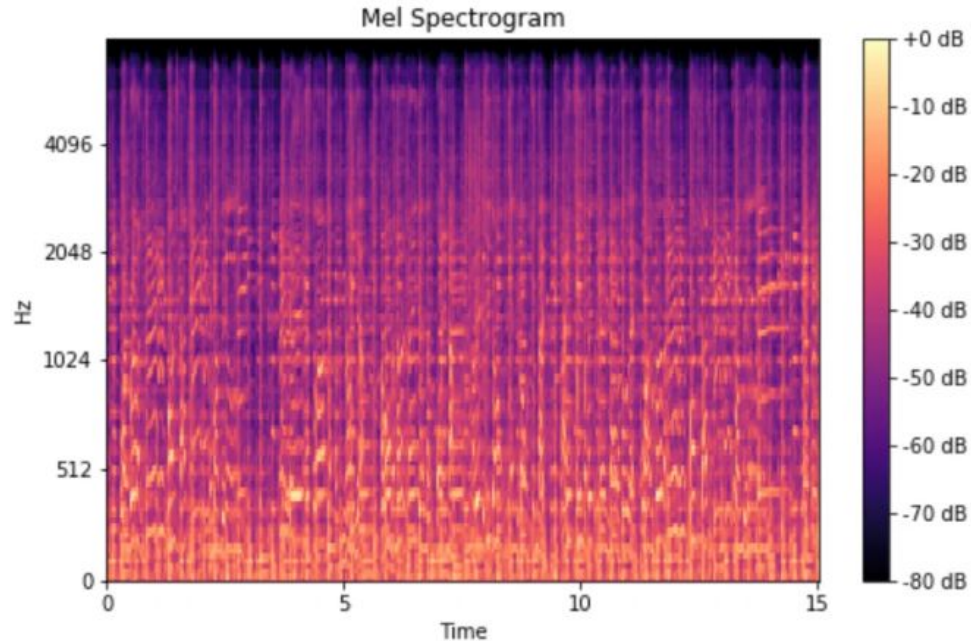
Model	Dutch	English	French	German	Italian
Whisper tiny	39.4	15.7	36.8	24.9	41.7
Whisper base	28.4	11.7	26.6	17.7	31.1
Whisper small	17.2	8.3	16.2	10.5	21.4
Whisper medium	11.7	6.8	8.9	7.4	16.0
Whisper large	10.2	6.3	8.9	6.6	14.3
Whisper large-v2	9.3	6.2	7.3	5.5	13.8

Table 10. WER (%) on MLS

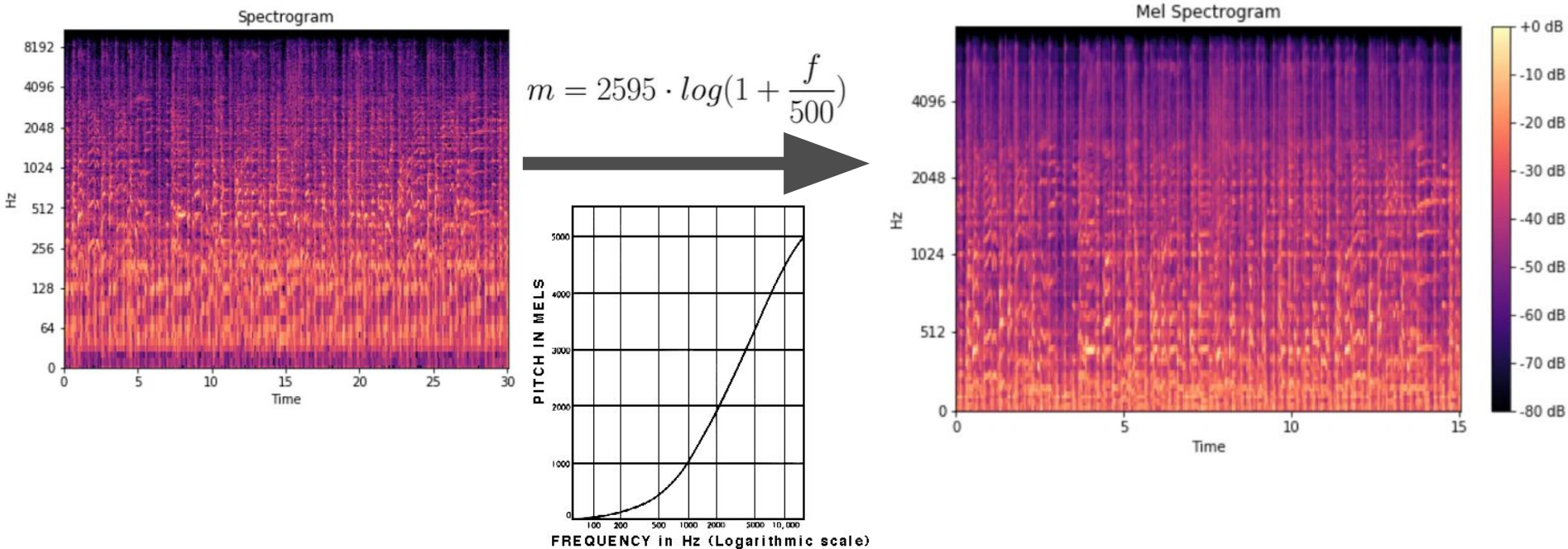


Speech2Text: Pre Procesamiento

- Cortamos el audio original en los recortes resultantes de la diarization.
- Resampleo del audio a 16KHz.
- Cada recorte del audio se divide en ventanas:
 - De 25ms de duración.
 - Superpuestas 10ms.
- Se genera una representación espectral en Log-Mel de cada ventana.



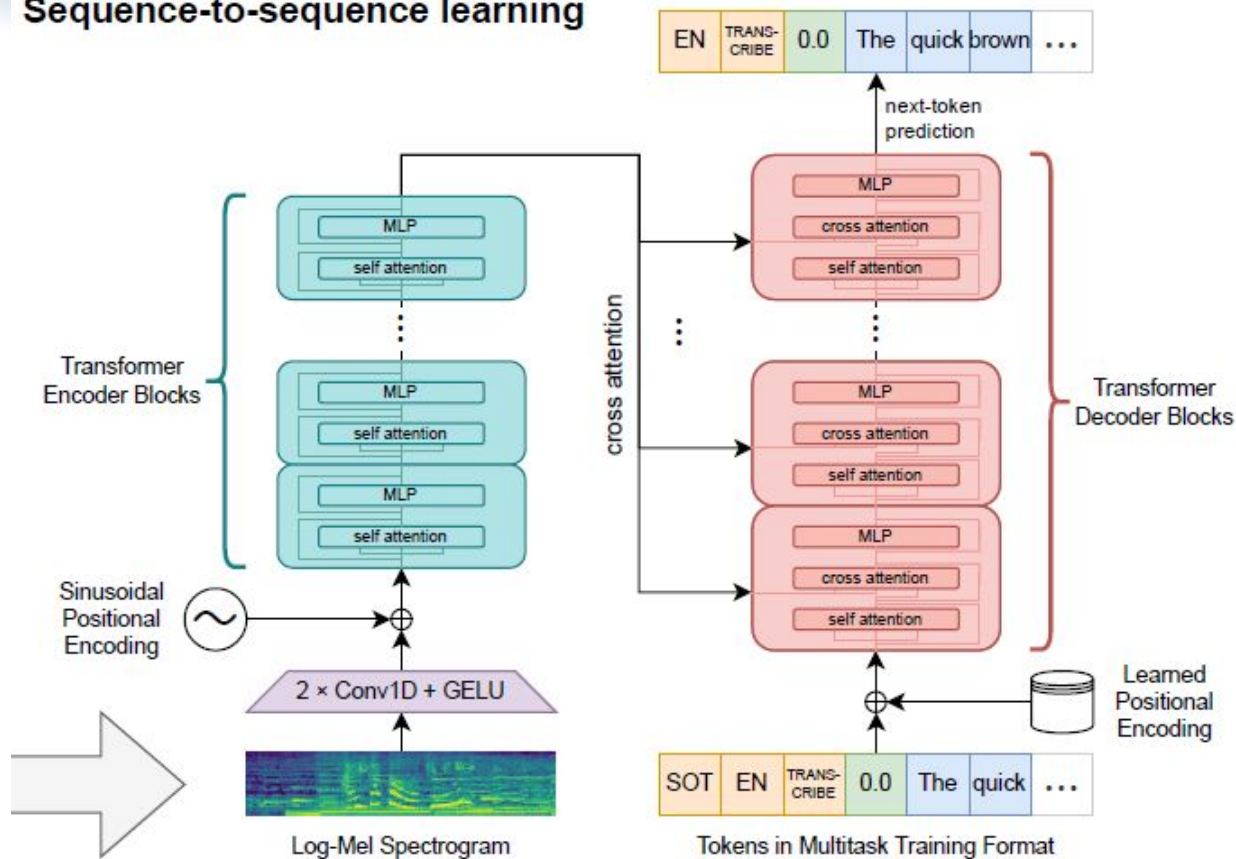
Speech2Text: Pre Procesamiento



Speech2Text: Arquitectura

- Capas Convolucionales.
- Información Posicional.
- Transformers:
 - MLP.
 - Mecanismos de atención cruzada y propia.
- Embebidos de observaciones previas

Sequence-to-sequence learning



Speech2Text: paso a paso

- Cortamos el audio original en las divisiones resultantes de la diarization.
- Transcribimos recorte por recorte.
 - Seleccionamos modelo.
 - Activamos o no GPU.
- Agregamos el resultado de la transcripción a la estructura con los timestamp de inicio y fin y el nombre del orador.
- La mostramos en la Interfaz.

Inicio	Fin	Orador	Diálogo
t0	t1	A	Hola
t2	t3	B	como andas?
t4	t5	A	Bien, vos?

Interfaz Gráfica

- Carga de Audio
- Carga de Oradores
- Selección de Modelo
- Des/Activación de GPU
- Transcripción
- Descarga Transcripción
- Gráfico de probabilidades de orador

The screenshot displays the 'Audio' tab of a graphical user interface. At the top, there are three tabs: 'Audio' (selected), 'Text', and 'Graph'. Below the tabs, the 'File:' dropdown menu shows 'dibu.wav'. To the right is a playback control bar with a play button, a progress indicator at '0:00 / 1:02', a volume icon, and a settings menu. Below this, there is a 'Name' input field, a 'time interval' slider set to '0:07 - 0:09', and an 'Add' button. A 'Speakers' list box contains two entries: 'fulano 0:04 - 0:06' and 'D 0:07 - 0:09'. Below the list is a 'Delete' button. To the right of the speakers list is a 'Model size:' dropdown menu set to 'base' and a checkbox labeled 'GPU' which is checked. At the bottom, there is a 'Process' button.

Audio Text Graph

File:

▶ 0:00 / 1:02 🔊 ⋮

time interval Add


Speakers

- fulano 0:04 - 0:06
- D 0:07 - 0:09


Delete Model size: ☒ GPU

Process

Aprendizajes Secundarios: Crowdfunding


 **Closed**


Error while using Resemblyzer #80
Waseem-786 opened this issue on Apr 2 · 2 comments

**FranBasili** commented on Apr 3

...

See the changes made in [#71](#) and make them manually in your PC or notebook. They have update the git repository but they haven't update it in pip




**CorentinJ** commented on May 6

Collaborator ...

I've made a push, please reopen the issue if you encounter any problem

<https://pypi.org/project/Resemblyzer/0.1.2/>



Bibliografía

- **Resemblyzer:** <https://github.com/resemble-ai/Resemblyzer>
- **Whisper:** <https://github.com/openai/whisper>
- **Repositorio:** <https://github.com/KevinWahle/ASSD-TPF>

¿Preguntas?

