Develop of a Text-To-Rap system

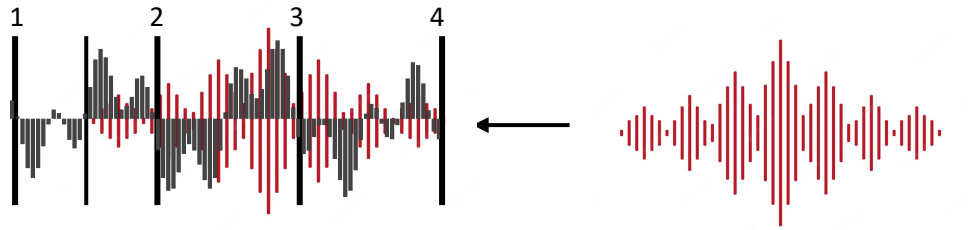
Deep Learning for Audio Signal Processing Final Project

Pablo Marcos Manchón

Table of contents

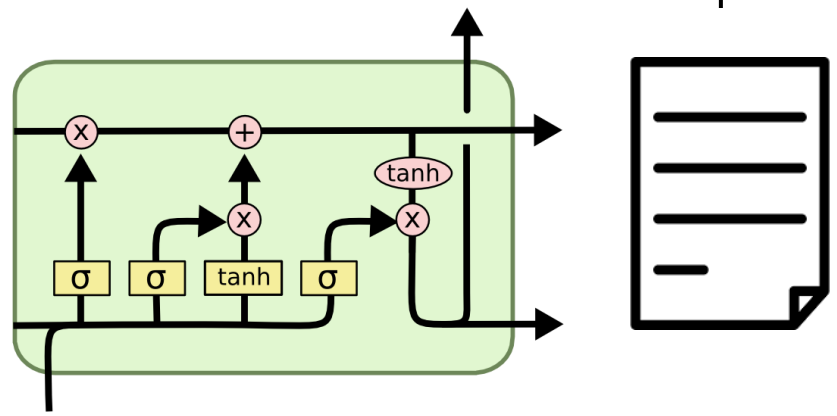
1. Previous work

- Idea to continue final work of the Neurocomputing course [1]. Development of a rap lyrics generation system.
- We downloaded a 30Mb corpus of text with song transcriptions.
- Use of an LSTM for text generation
- Use of google voice to read the song
- Basic adjustment of the voice, the first syllable of each sentence was aligned with the beat 0.5 of every 4 bars.
- Examples:



3. Join base + basic alignment

2. Text-to-Speech with google voice

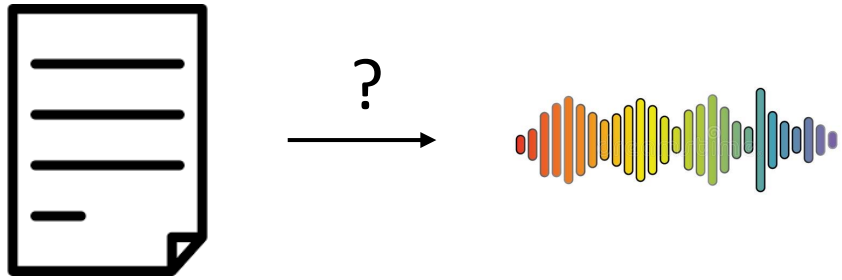


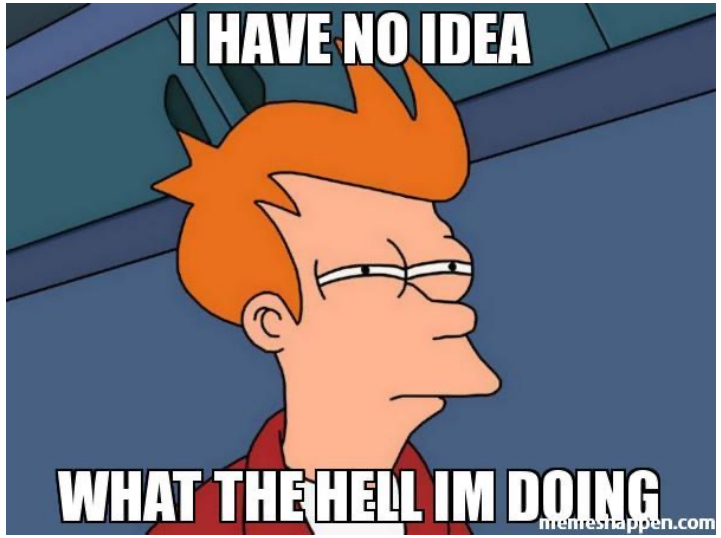
1. Text generation

[1] Use of a recurrent neuronal network to automatically generate rap music in Spanish. Jorge Arellano y Pablo Marcos. GitHub.

2. Objective

- Improving the Text-To-Speech system of the previous architecture.
- Try to develop a less robotic voice.
- Improve alignment of the voice with the base.
- Try to use deep learning techniques for audio whenever possible, even if there are simpler solutions.
- ~~Do not infringe any copyright~~.
- Limitations:
  - Colab available resources.
  - No prior data: need to create dataset from scratch.
  - No idea about TTS.
  - Do it during the December long weekend.





3. Data collection

- Use of Nach's discography. Key rapper in the evolution of the Spanish rap.
  - He usually sings alone
  - Style without major changes over the course of his career
  - No major variations in intonation, more rhythmic style, as if he were reciting poetry.
  - Very recognizable flow and timbre
  - Discography from 1994 - 2020
- For the dataset I downloaded 8 albums and 5 singles. A total of 117 songs. Around 9 hour of music.
- A bunch of challenges: Music and speech together, different speeds and emotions, other artists, parts without speech, different qualities, voice effects, etc.

El Idioma de los dioses (2011)

Me llaman (2014)

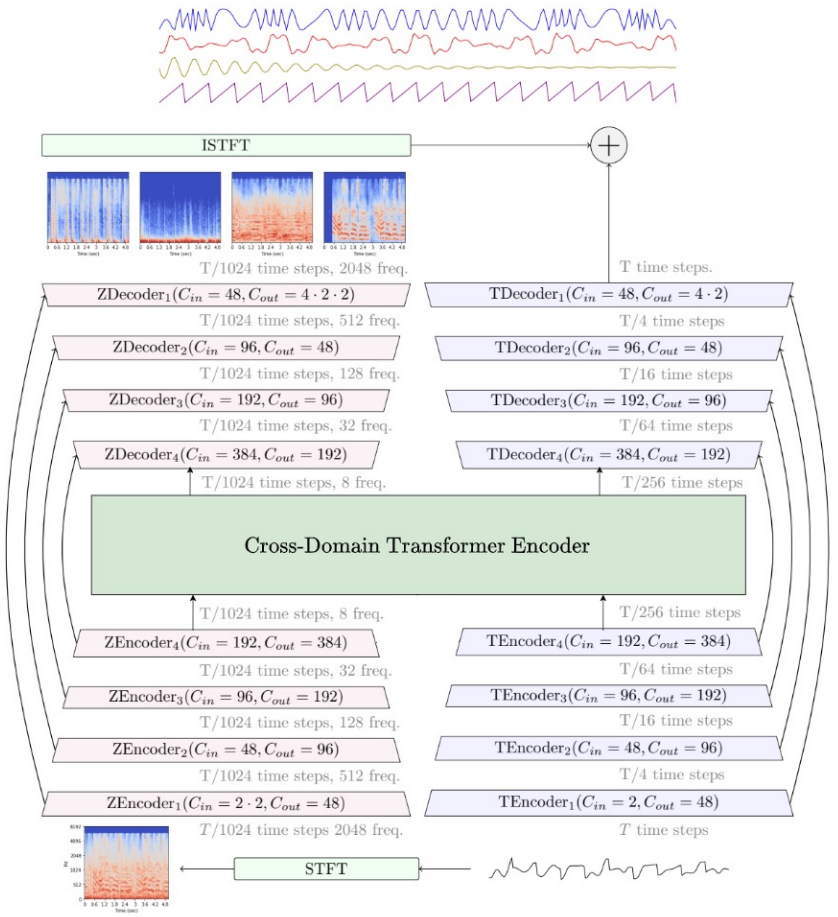
4. Data processing

4.1 Music Source Separation

- Objective: To separate the voice from the music and the rest of the elements of the songs.
- Use of Hybrid Transformer Demucs v4, released by Facebook Research in Nov 2022 (GitHub).
- Achieve state-of-the-art in source separation, 9.20 dB of SDR (Source-to-Distortion Ratio).

$$\text{SDR} := 10 \log_{10} \left(\frac{\|s_{\text{target}}\|^2}{\|e_{\text{interf}} + e_{\text{noise}} + e_{\text{artif}}\|^2} \right)$$

- I used it to separate the 117 songs into 4 parts: vocals, drums, bass, other.



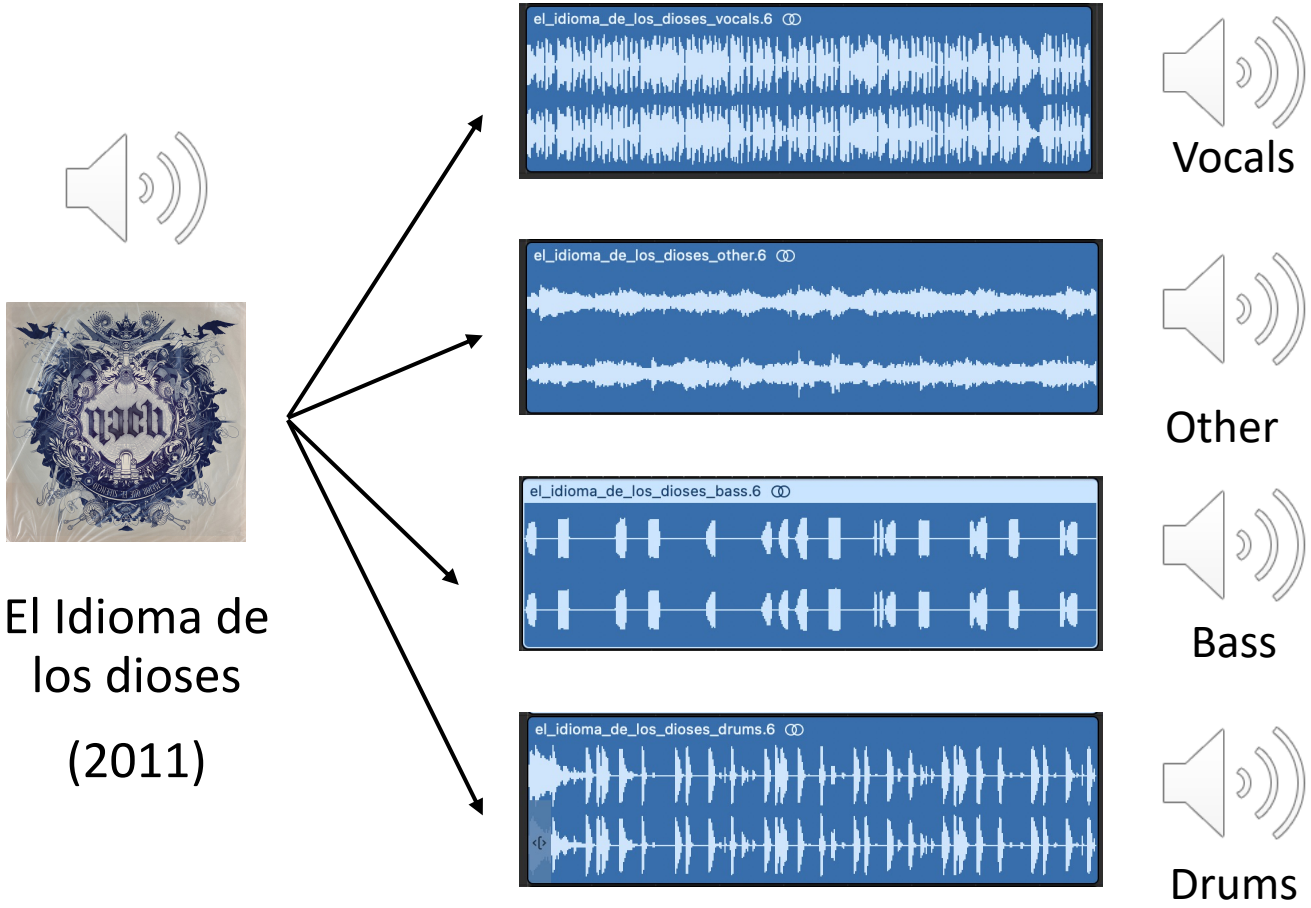
Hybrid Demucs architecture

Hybrid Spectrogram and Waveform Source Separation. Alexandre Défossez.
https://arxiv.org/pdf/2111.03600.pdf

4. Data processing

4.1 Music Source Separation - Example

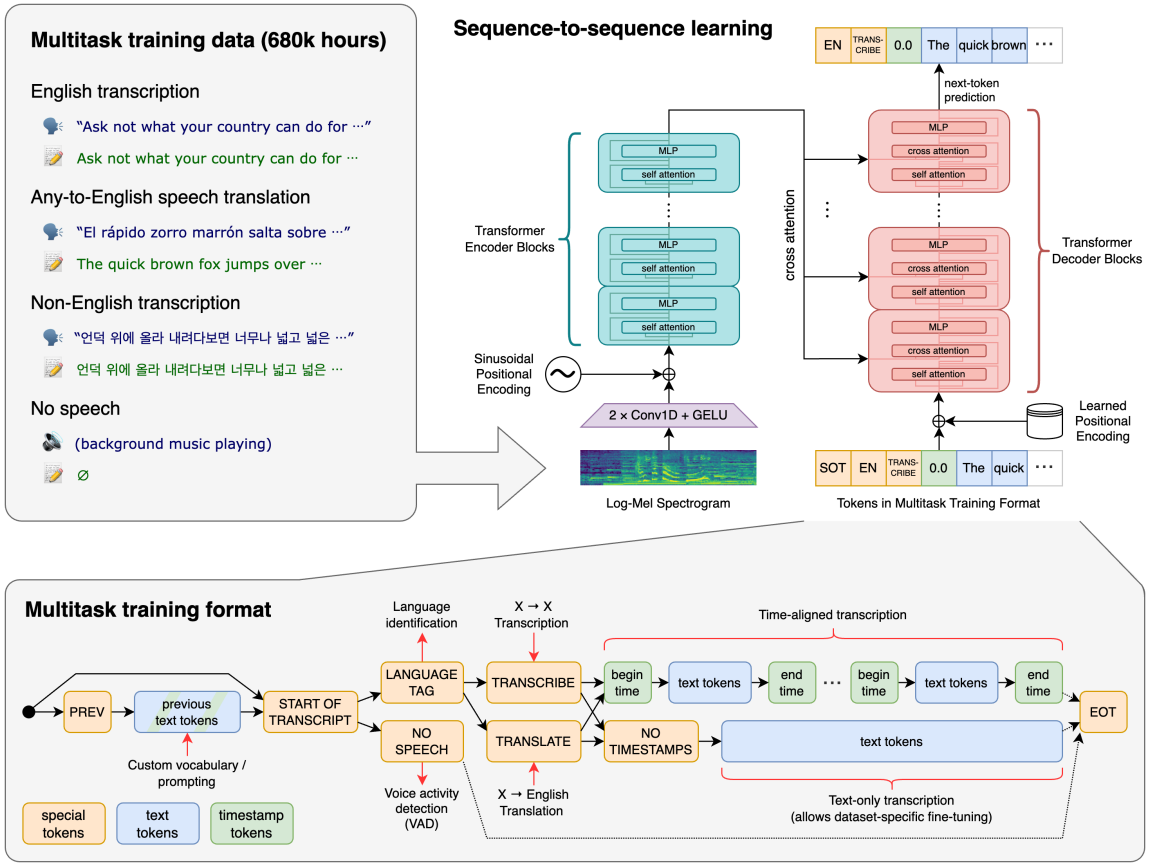
- Near-perfect results in almost all situations, even with loud background music
- Even with voices with effects: dubbed voices, chorus, reverb, etc.
- Issues: Some songs include vocals on the bases, multiple singers, etc.



4. Data processing

4.2 Speech Transcription

- Use Whisper to transcribe all the vocals tracks separated in the last stage.
- Model released by OpenAI in Sept. 2022 (GitHub)
- SOTA results for general purpose speech-to-text transcriptions
- I generated transcriptions for the 117 songs with timestamps at a sentence level
- Use of model “large” due to the complexity of the speech (fast, strange words, voice effects, etc).



Whisper architecture

Robust Speech Recognition via Large-Scale Weak Supervision. Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, Ilya Sutskever. https://cdn.openai.com/papers/whisper.pdf

4. Data processing

4.2 Speech Transcription - Example

El club de los olvidados.
Poesía difusa (2003)
Nach, Falsalarma

1.12 9.52
Oh, el gran Nach, Tito, Falsa Alarma, Mano con mano, es el club de los olvidados.

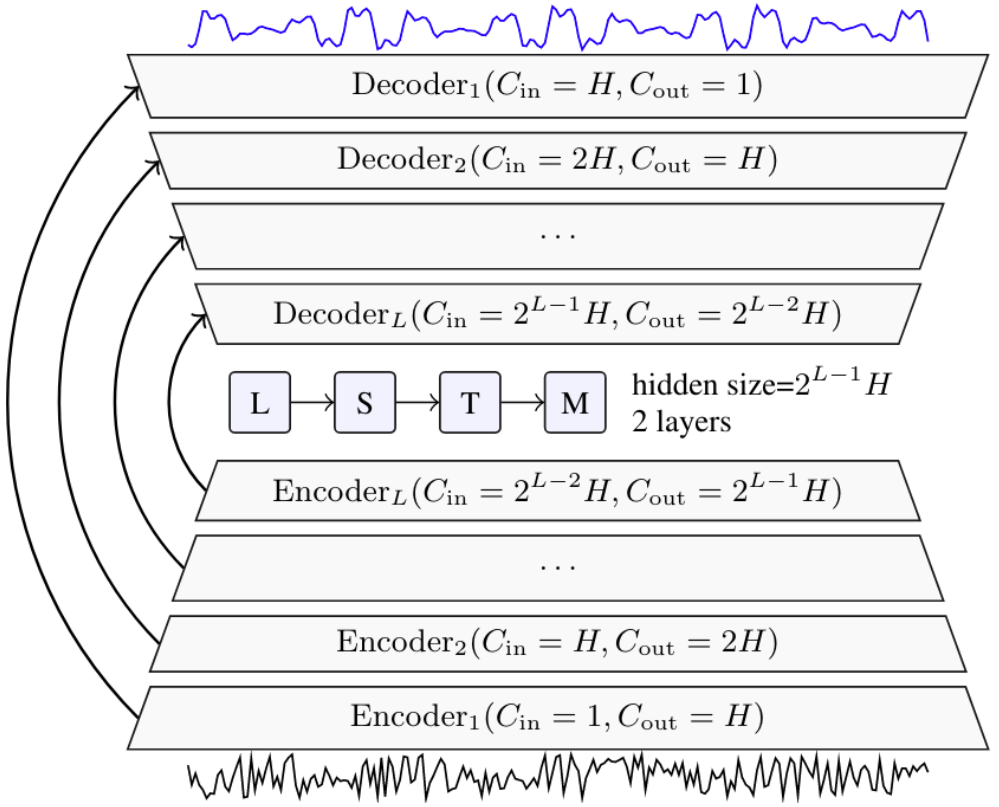
9.52 15.28
Alza la vista más allá del barrio, intenta abrir los ojos y sate el ruido diario, por favor.

15.28 20.44
Objetos materiales en tu armario a los que no les das valor, vida fácil la del ágil vencedor.

4. Data processing

4.3 Denoising

- Denoise the vocal audios to improve the results (better audio quality will imply better TTS system).
- Use of Real Time Speech Enhancement in the Waveform Domain, Interspeech 2020 (Github).
- **Step finally discarded.**
  - Not trained to remove noise like the one in our audio tracks (base voices).
  - Other denoisers have been tried without good results.
  - **Maybe, a speaker separator would be needed**
- Example of noisy audio:



Denoiser architecture

Real Time Speech Enhancement in the Waveform Domain.
Alexandre Defossez, Gabriel Synnaeve, Yossi Adi.
https://arxiv.org/abs/2006.12847

4. Data processing

4.4 Segment cutting

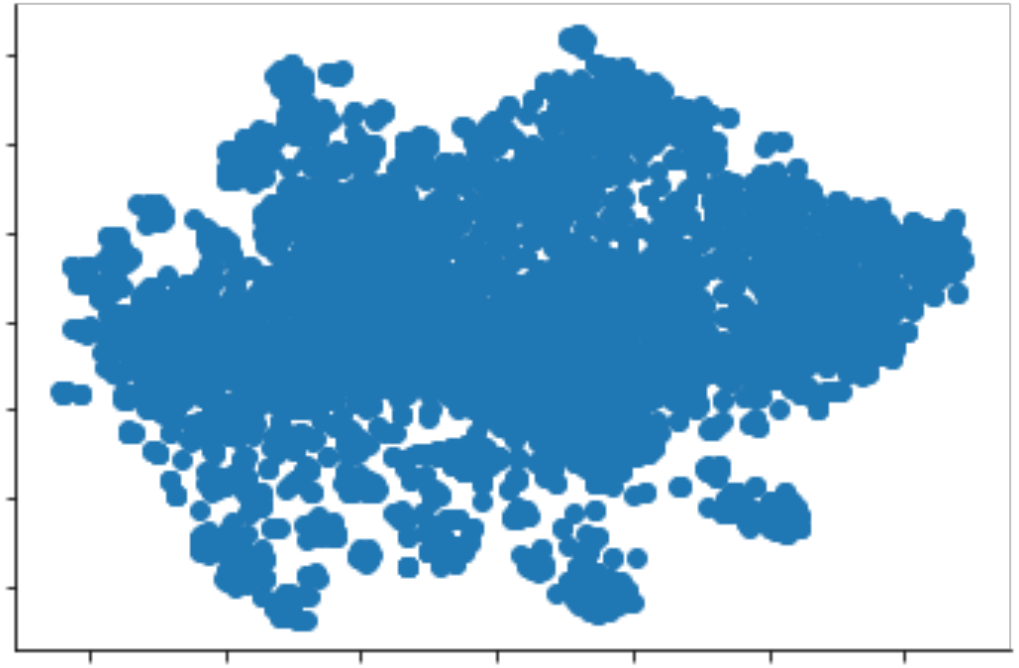
- Cut tracks with vocals of all songs into segments for further training of the system
- Use of timestamps per phrase provided by whisper
- Creation of a file associating each audio fragment with its text
- In total 9.055 sentences with 7.42 hours of speech
- Example:
  - Text: “Busco una calma inalcanzable, la atmósfera aquí no es fiable”
  - Filename: segment1.wav

Index	Text	Filename	Song
0	Busco una calma inalcanzable, la atmósfera aquí n	segment1.wav	53_nada_ni_nadie
1	quiero estar solo, si solo todo estará bien	segment2.wav	53_nada_ni_nadie
2	que nadie me hable, que no rompan este silencio e	segment3.wav	53_nada_ni_nadie
3	hoy quiero sentir el frío, vértigo que el mundo pare	segment4.wav	53_nada_ni_nadie
4	harto de fingir excusas, musa siento huir de mí, cos	segment5.wav	53_nada_ni_nadie
5	esta cicatriz de traumas, de sangra versos, desarm	segment6.wav	53_nada_ni_nadie
6	es mi verdad maldita, mitad genio, mitad flor march	segment7.wav	53_nada_ni_nadie
7	porque haga lo que haga el premio, no cambiará m	segment8.wav	53_nada_ni_nadie
8	es este sentimiento pésimo, que me tiene pálido co	segment9.wav	53_nada_ni_nadie
9	preguntan qué sucede y me limito a mirar serio mi	segment10.wav	53_nada_ni_nadie
10	me mira y sé que ve una decepción constante y si l	segment11.wav	53_nada_ni_nadie
11	quiero escapar a mi desierto sin ser visto salir de e	segment12.wav	53_nada_ni_nadie
13	nada ni nadie hoy me acompaña en este baile quier	segment14.wav	53_nada_ni_nadie
14	que nadie hable, me falta el aire por una vez que el	segment15.wav	53_nada_ni_nadie
15	nada ni nadie hoy me acompaña en este baile quier	segment16.wav	53_nada_ni_nadie
16	que nadie hable, me falta el aire por una vez que el	segment17.wav	53_nada_ni_nadie
17	me importa una mierda lo que el resto diga que se	segment18.wav	53_nada_ni_nadie
18	mi única enemiga es esta mente rota desde crío, al	segment19.wav	53_nada_ni_nadie
19	sonrío por compromiso y casi no veo a los míos, mi	segment20.wav	53_nada_ni_nadie
20	con mi rap estoy de luto, no disfruto es mi veneno,	segment21.wav	53_nada_ni_nadie
21	y si pierdo confianza atado a las circunstancias hag	segment22.wav	53_nada_ni_nadie
22	y con dios mantuve un pacto demasiado triste, él ja	segment23.wav	53_nada_ni_nadie
23	perdiste el norte, solo perdí al jugar con miedo, al s	segment24.wav	53_nada_ni_nadie

4. Data processing

4.5 Speaker identification

- The audio excerpts contain phrases from multiple singers
- Phrases with wrong cuts, including unvoiced parts
- Use of Deep Speaker architecture (GitHub) to eliminate erroneous segments.
- Calculation of a speaker vector for each segment (extracting it from an embedding layer of the architecture of size 512).
- Use of cosine similarity to measure the distance between segments.
- At a glance, we can see clusters in the embeddings.



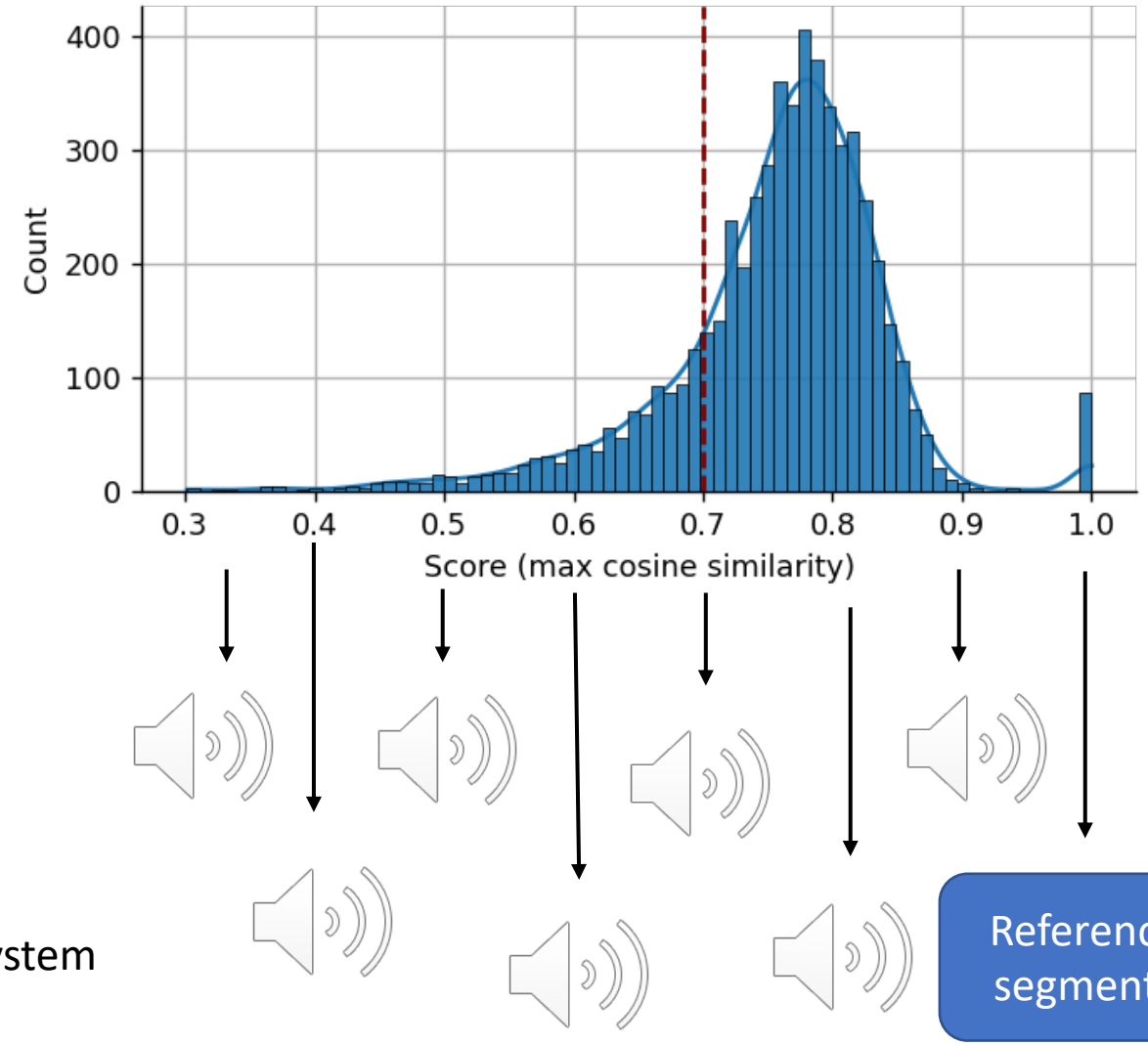
TSNE of the segment embeddings

Deep Speaker: an End-to-End Neural Speaker Embedding System
Baidu Inc. https://arxiv.org/pdf/1705.02304.pdf

4. Data processing

4.5 Speaker identification - Filtering

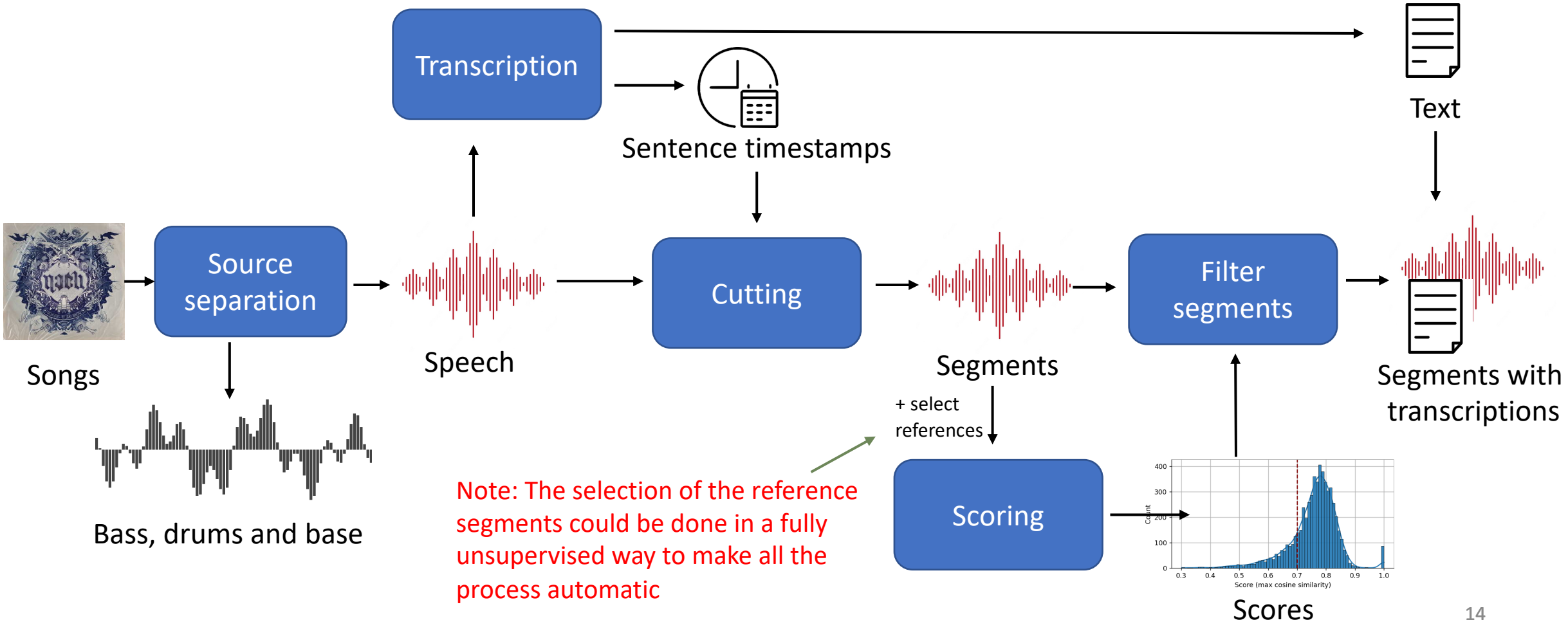
- I selected reference phrases randomly (chosen manually).
- The distances to the reference segments have been calculated.
- Filtered all those segments with distance less than 0.7
- Phrases with less than 3 words have also been filtered.
- If I were to repeat the experiment now I would be stricter, with a higher threshold and using more filter to detect too long sentences.
- After the filtering: 5.992 segments.



Deep Speaker: an End-to-End Neural Speaker Embedding System
Baidu Inc. https://arxiv.org/pdf/1705.02304.pdf

4. Data processing

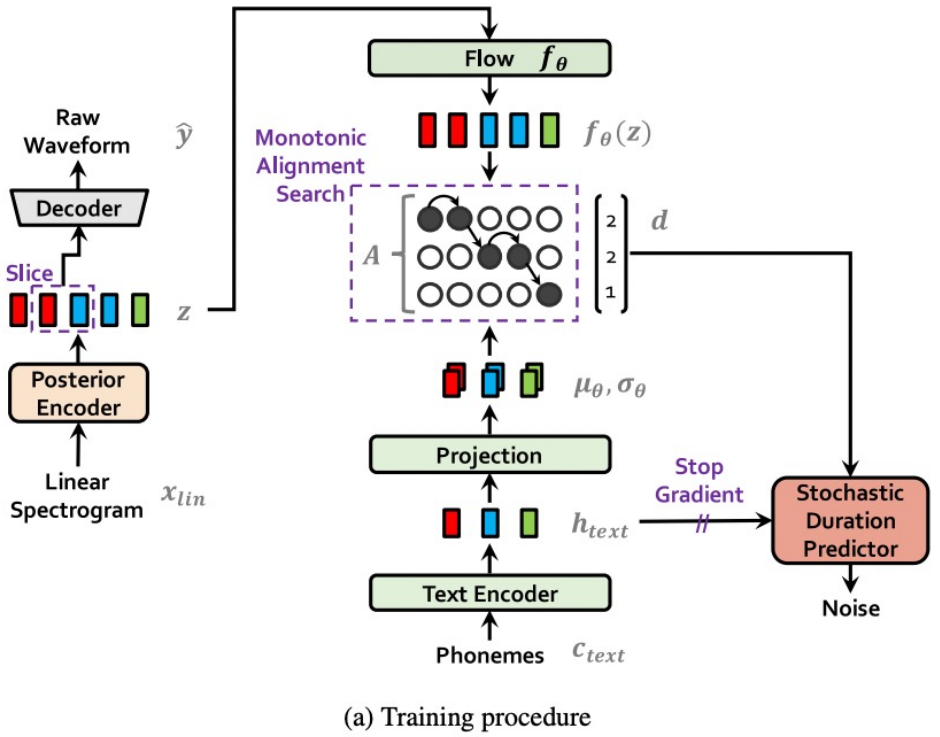
4.6 Recap – Weakly supervised pipeline



5. Training the TTS system

5.1 Model

- There are tutorials to train systems with a voice, but all the ones I found were based on English or other languages (in English or Chinese phonemes generally).
- Not enough computing resources to train a system from scratch
- After much searching, I found a pre-trained model in Spanish in the library coqui-ai/TTS (model with name *tts_models/es/css10/vits*)
- Pre-trained VITS model in Spanish based on a conditional variational autoencoder.
- Trained with chars not phonemes



(a) Training procedure

VITS model procedure (training)

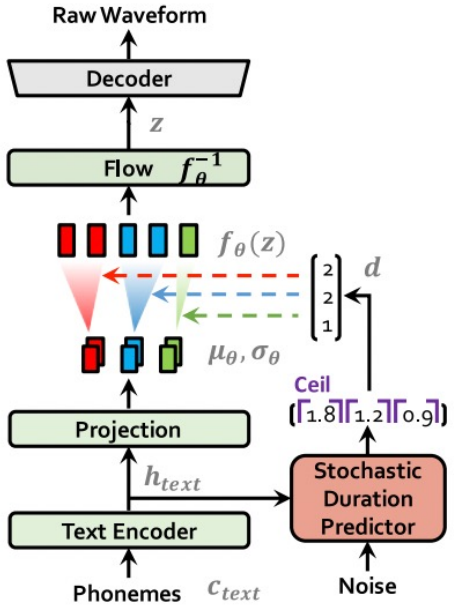
Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech.
Jaehyeon Kim, Jungil Kong and Juhee Son
https://arxiv.org/pdf/2106.06103.pdf

5. Training the TTS system

5.2 Fine-tune issues

Fine-tuning the model was a challenge:

- No documentation for fine-tuning. Understand how the framework was intended without help
- The Spanish model was not prepared for fine-tunning, the parameters used to create the model were not documented (I had to infer it from the PyTorch state dictionary).
- The discriminator network weights were not stored (only the part for inference was stored)
- Prepare the dataset to be in the same format (same text processing, speaker embedding managing, …)
- Adjust learning rates (there was multiple in the architecture) taking into account the missing part of the network
- Train for ~2 hours in Colab standard GPU



(b) Inference procedure

VITS model procedure (inference)

Model before fine-tunning

Model after fine-tunning

5. Training the TTS system

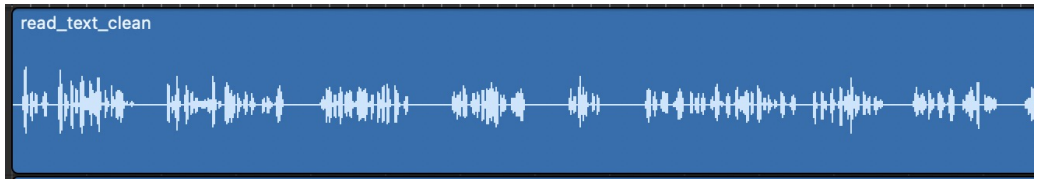
5.3 Things I learned and would change

- I would use fewer audios, but from the same domain and better quality. As the generation of phrases is done independently, between phrases the model still shouts or is quieter and gives an irregular feeling. It could be retrained again with only 5-6 songs with a same style.
- I would try to freeze layers for training, especially since the discriminator does not have pre-trained weights.
- The network is very sensitive to the length of segments, it would filter out segments that are too long (many erroneous).
- I would try to adjust the length of the sentences, the times given by whisper are not very accurate, when performing inference sometimes it cuts part of the first and last syllable as a result.

6. Aligning sentences

6.1 Testing the speech generation

- First test of the system:
  1. Take one of Nach's songs (Esclavos del destino)
  2. Separate base and voice
  3. Transcribe voice
  4. Read song with TTS
  5. Join with the base without any alignment.
- **Problem**: The network has learned to generate sentences separately, but generates them independently. It is necessary to adjust the spacing between sentences.
- The gaps between sentences can be seen in the waveform.

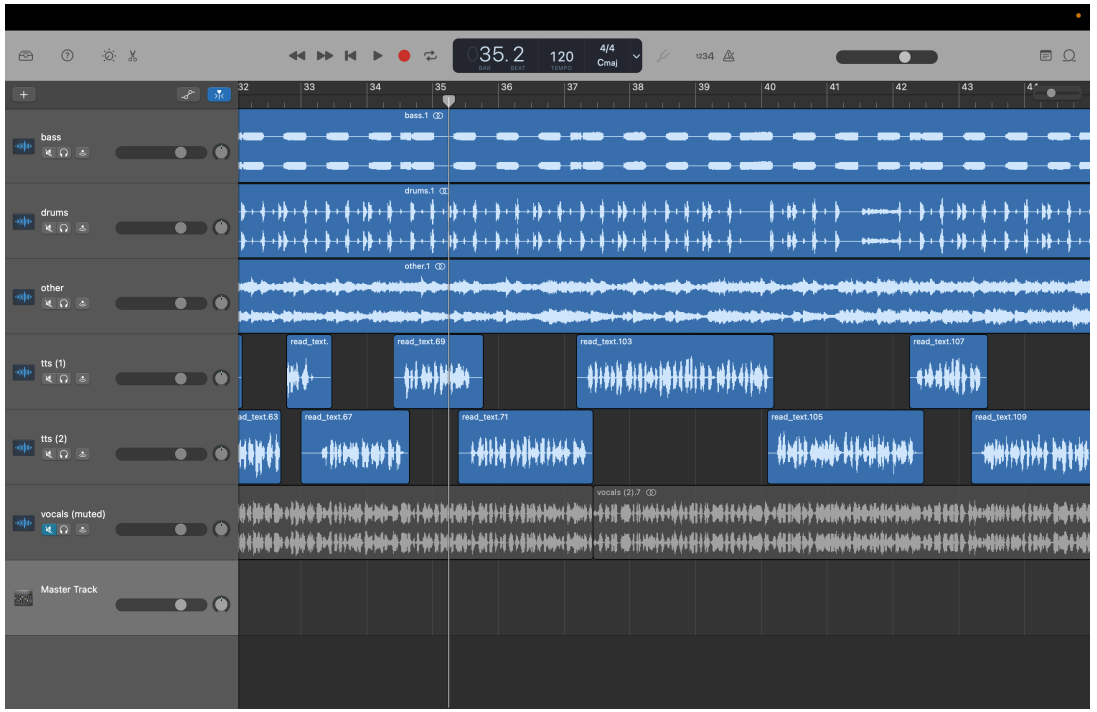


"Esclavos del desino" sing by TTS system without any aligment

6. Aligning sentences

6.2 Manually aligment

- The sentences are in rhythm. It is only necessary to adjust the spacing between them.
- To test how the result would look like, I cut the spaces manually
- No further transformations have been made (no warping, no echo, reverb, etc).
- Result:



“Esclavos del destino - Nach” but readed by TTS system and manually aligned. Cut.

7. Live test

https://colab.research.google.com/drive/1T2ucPAtvb8sfPxFz8j_YOri3e8RpYd2N#scrollTo=fKOb70c2BoUT

Don't forget to buy my disc at the exit!!

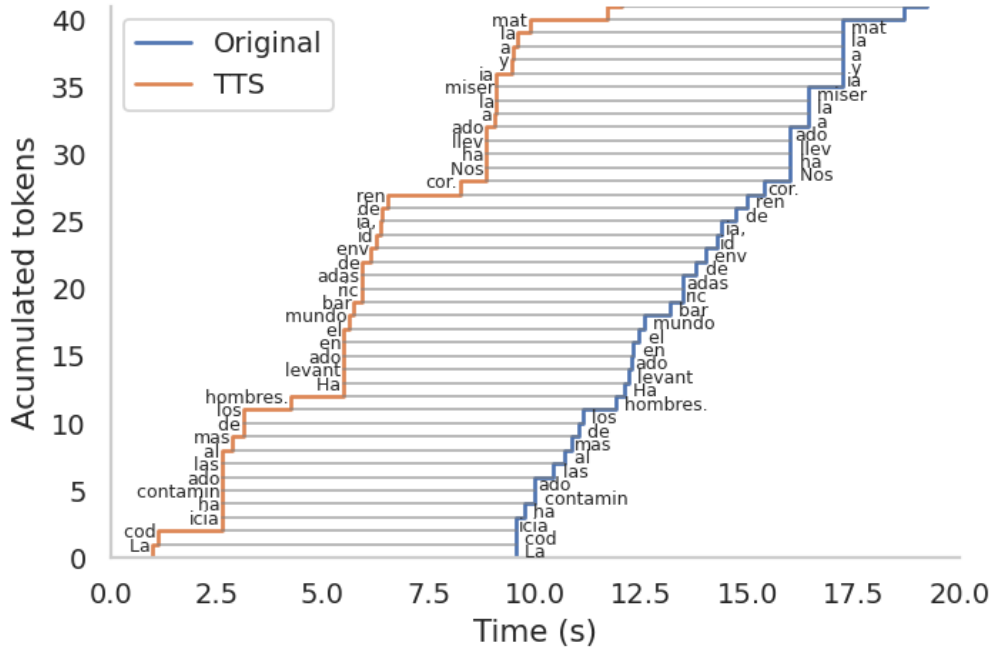
Diapositivas descartadas para no alargar más la presentación

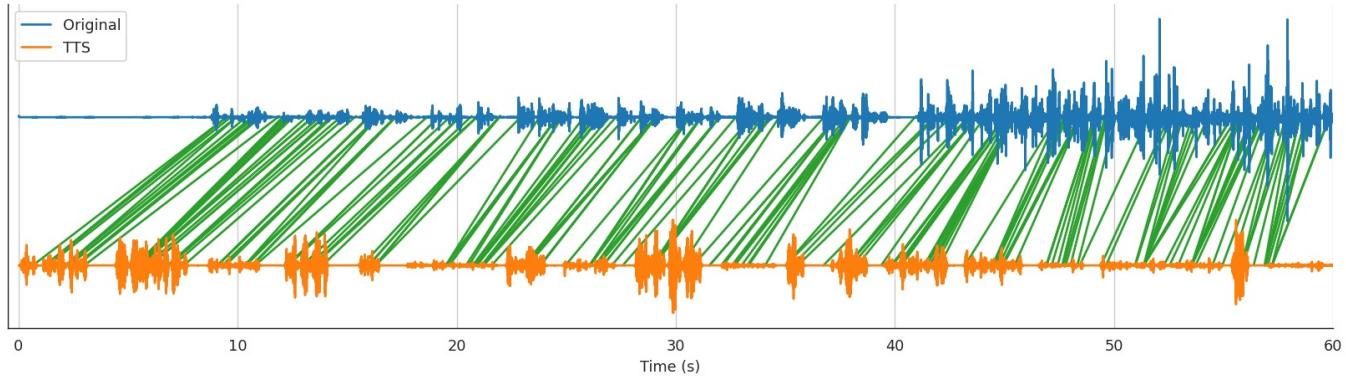
6. Aligning sentences

6.3 Use of whisper timestamps

Attempt to align automatically. Idea -> Copy the rhythm of a song and replace the voice.

1. Use a modified version of whisper when transcribing to obtain timestamps of tokens by extracting them from logits(Github).
2. Transcribes the original audio and the one read by the TTS.
3. Match both texts at the letter level by calculating substitutions, insertions, deletions(Myers's bit-vector algorithm)
4. Use the result to match tokens
5. Use the timestamps of the tokens to have a time relationship between the audio read by the TTS and the original voice.



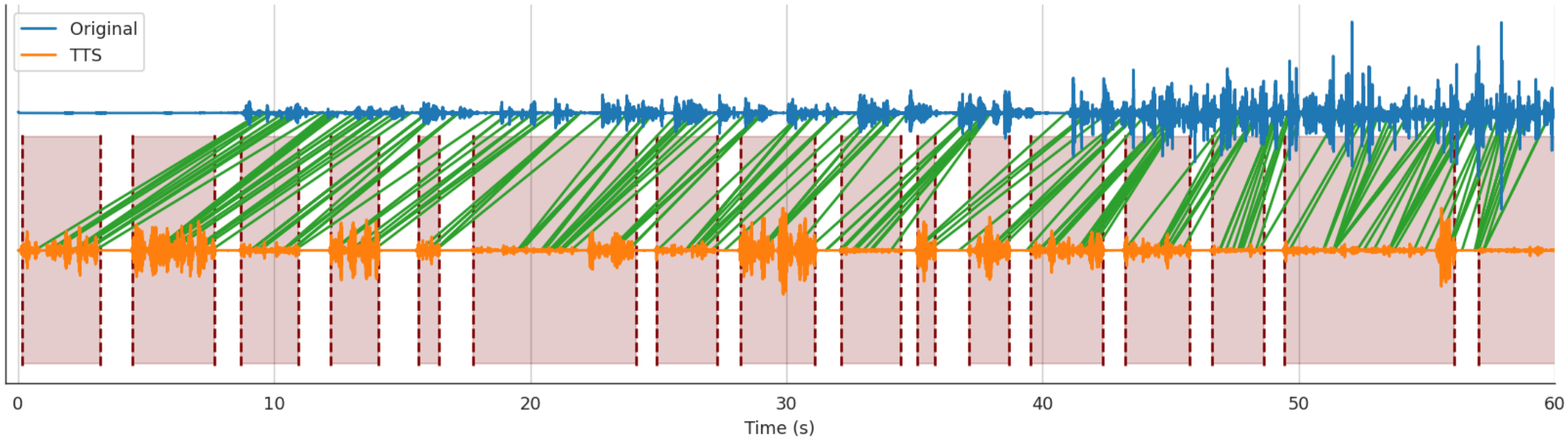


6. Aligning sentences

6.3 Shift blocks using landmarks

Align speech sentences using timestamp relations

1. Find speech windows (I did it using a window to look for the spaces and find the blocks carefully).

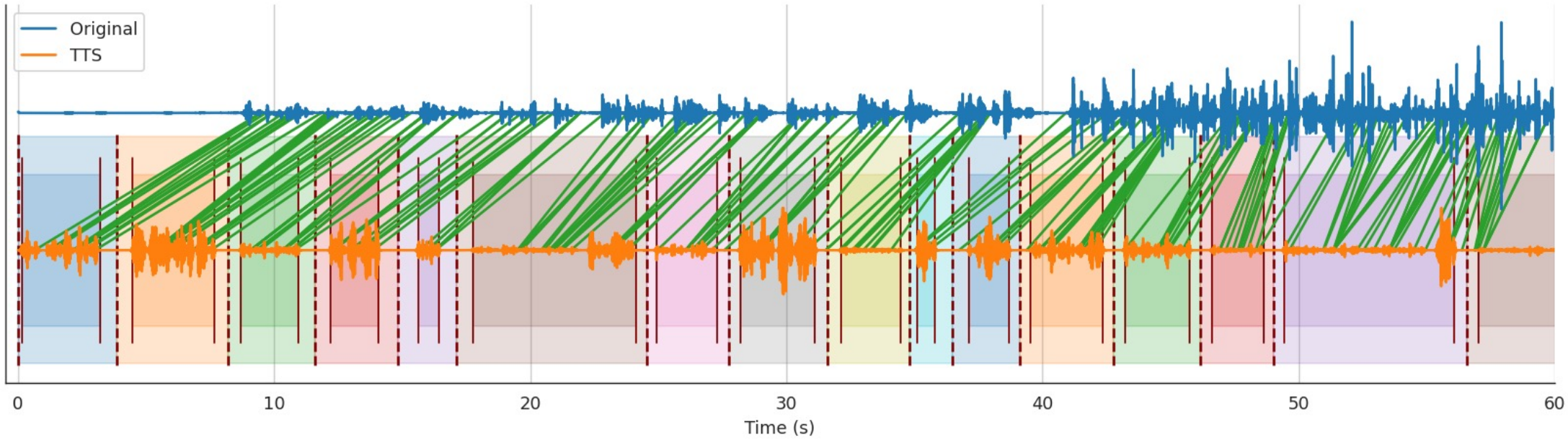


6. Aligning sentences

6.3 Shift blocks using landmarks

Align speech sentences using timestamp relations

2. Extend the widows to adjacent spaces to cover all the track

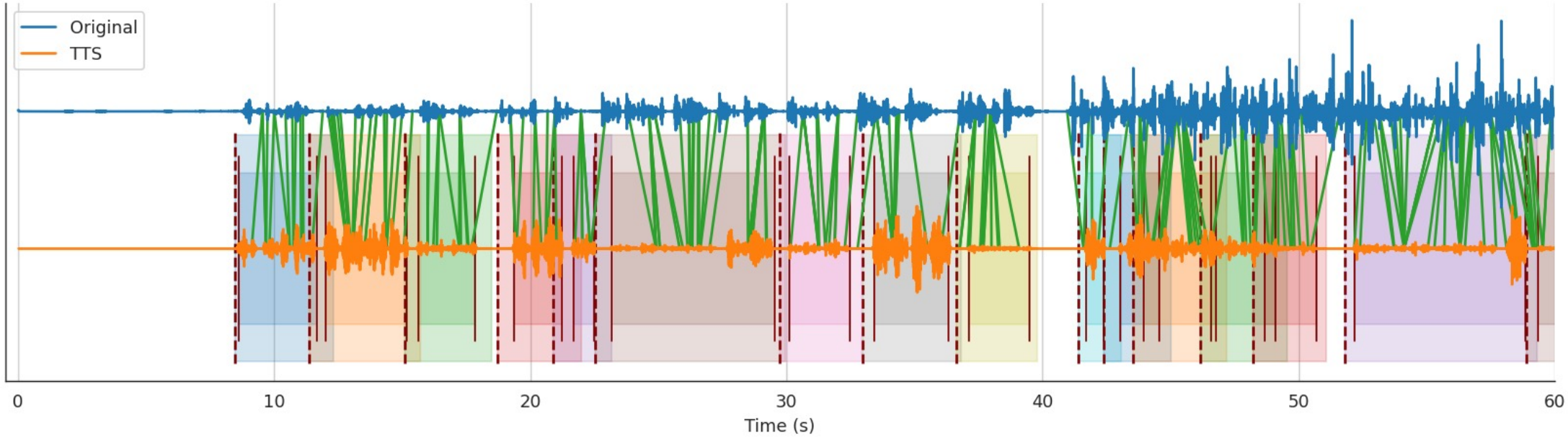


6. Aligning sentences

6.3 Shift blocks using landmarks

Align speech sentences using timestamp relations

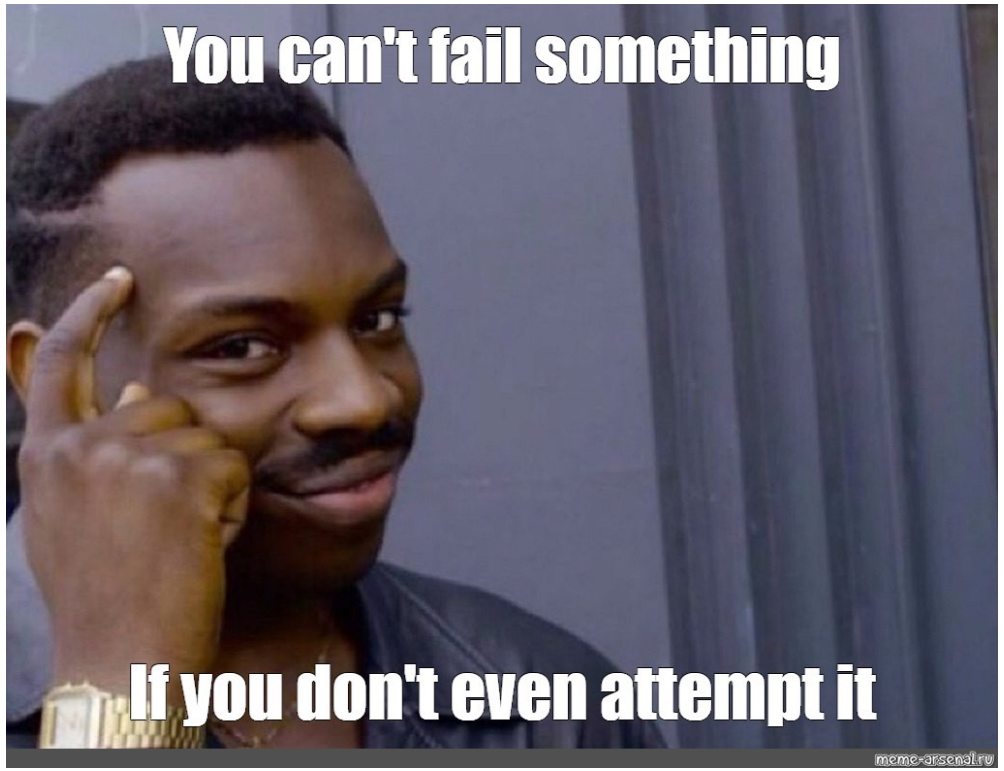
3. Calculate the shift of each block (For example, use the mean distance between landmarks mean(xi-yi) to minimize the square distance) and merge the shifted blocks in a wav file (this part was very tricky, I had to code an algorithm to merge the waves taking care of the overlapping).



6. Aligning sentences

6.3 Shift blocks using landmarks

- The results are **not very good** because **Whisper's token timestamps are not accurate** (it is not intended for this purpose).
- With good timestamps (force-alignment task) it could be possible to align the base with the voice with good results.
- I have not investigated this line further, as to use it you need the landmarks of someone singing over it.
- The algorithm of shifting blocks and joining them together can be very useful, we just need to calculate the displacement of each block. -> **We will infer it from the beats.**



6. Aligning sentences

6.3 Automatic alignment based on beats – Find beats

- First we need to find the base beats and then align the songs with the base.
- It's easy, since in rap there are usually very marked drums and they are always edited from studio loops.
- Generally there are never time changes, so the assumption that there is a global tempo is true.
- Use of the beats detector based on RNNs from the *madmon* library (Github).
- Example of a song fragment with the tempo detected by the model. Song Palabras (sing by the original Nach).

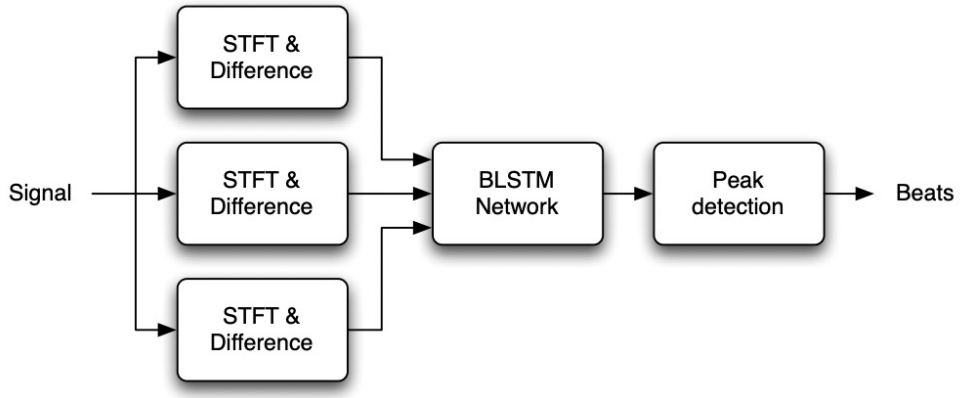


Figure 3: Basic signal flow of the presented beat detector / tracker

Sebastian Böck and Markus Schedl, "Enhanced Beat Tracking with Context-Aware Neural Networks", Proceedings of the 14th International Conference on Digital Audio Effects (DAFx), 2011. http://recherche.ircam.fr/pub/dafx11/Papers/31_e.pdf

6. Aligning sentences

6.3 Recap – Basic aligment based on beats