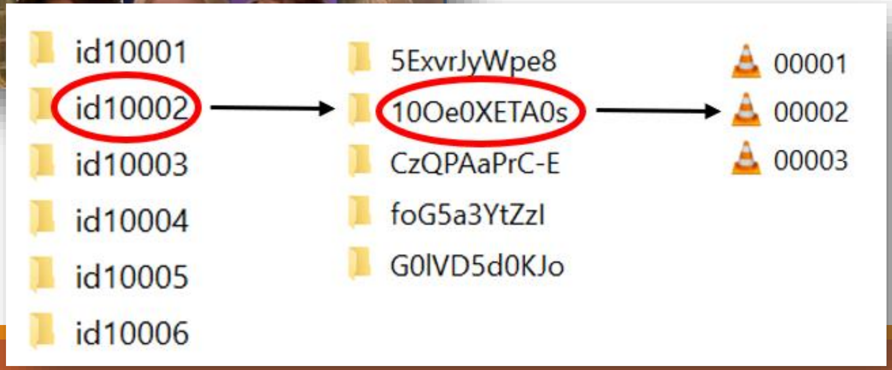
EVALUATING SPEECH SEPARATION THROUGH PRE-TRAINED DEEP NEURAL NETWORKS MODELS

JOSE IGNACIO POZUELO
VIJAY GURBANI
DAN PLUTH
AYUSH PANDA

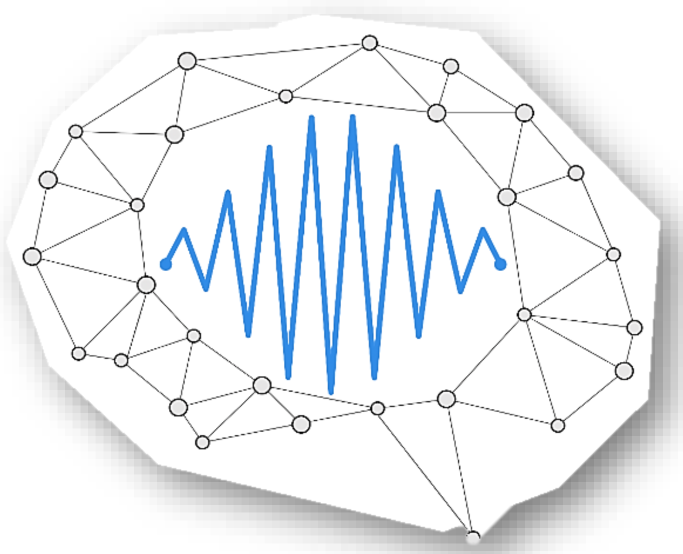
Introduction


VoxCeleb
amazon Transcribe
Google Cloud
Speech to Text

VoxCeleb dataset



SepFormer model



```
model = SepformerSeparation.from_hparams( source=' speechbrain/sepformer-whamr',
                                          savedir='pretrained_models/sepformer-whamr')
```

Separation Procedure

```
est_sources = model.separate_file('audio_mixture_path')
```

```
torchaudio.save('audio1_separation_path', est_sources[:, :, 0].detach().cpu(), 8000)
torchaudio.save('audio2_separation_path', est_sources[:, :, 1].detach().cpu(), 8000)
```

Analysis I

Audio Mixture Procedure

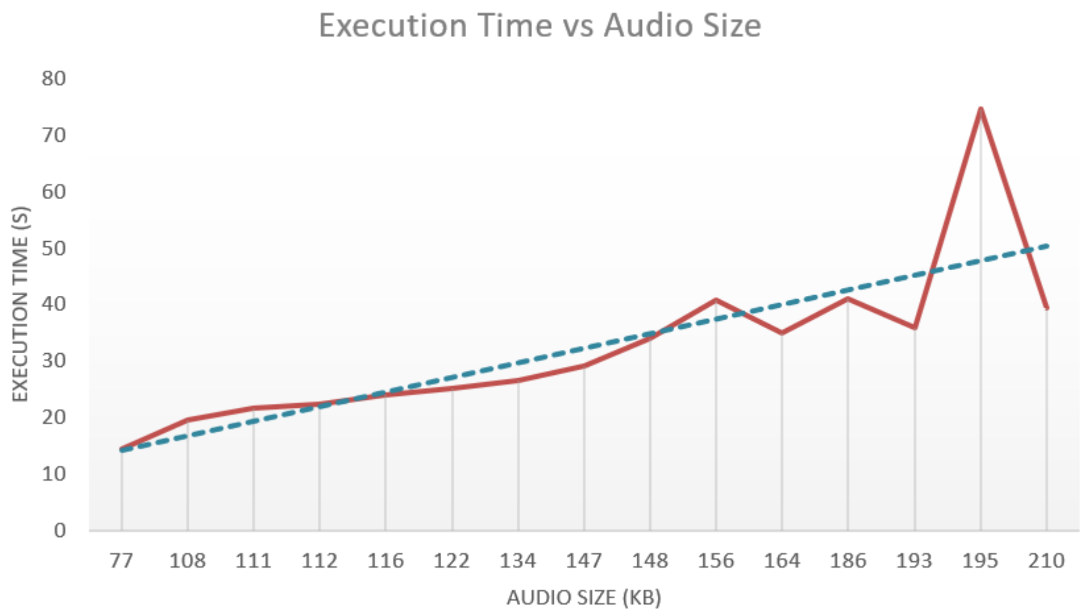
ffmpeg -i *'audio_path'*-af loudnorm=I=-25:LRA=7:TP=-2 *'audio_path_normalize'*

ffmpeg -y -i *'audio1_path'* -i *'audio2_path'* -ar 8000 -filter_complex *'filter parameters'* *'audio_mixture_path'*

Filter Parameters	Value
Delay	0 – 3 seconds (random)
Volume	1.7
Inputs	2
Duration	shortest / longest

15 Mixture Separation

delay | background noise | audio length | execution time



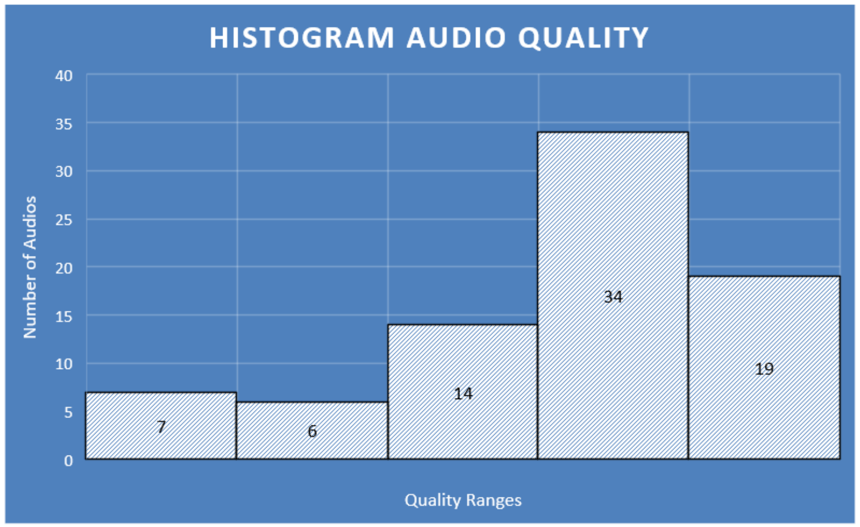
Introduced Delay → No effect

Length unrelated with after-separation quality

Runtime separation & Mixture size

80 Mixture Separation

5	Perfect audio separation
4	Good audio separation (low distortion)
3	Correct audio separation (medium distortion)
2	Bad audio separation (high distortion)
1	Audio separation failure



Different languages → Higher quality after-separation

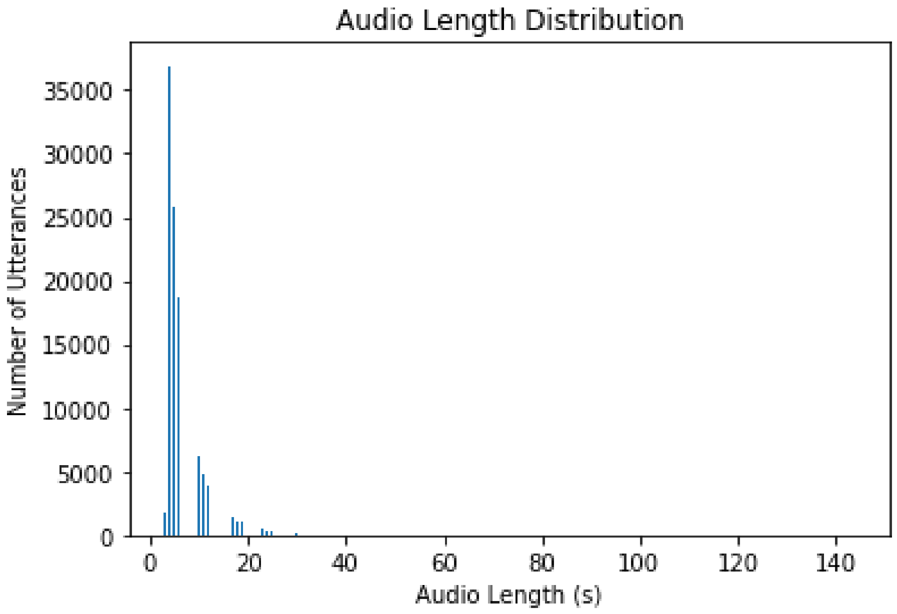
Audio quality seems to improve towards the end of the audio

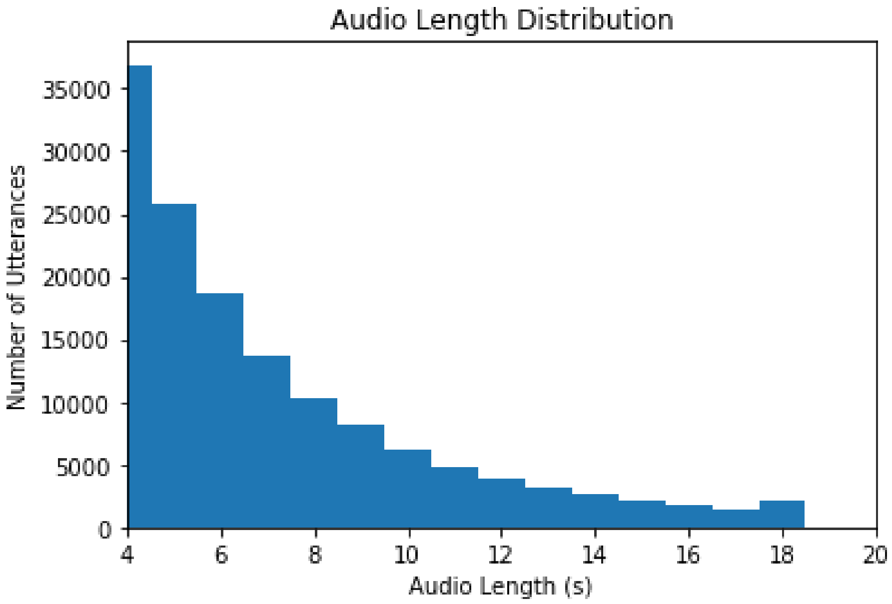
Great background music detection and elimination

AVERAGE QUALITY → 3.7

Analysis II

Utterances Selection





400 utterances → random choice (4 – 10 seconds)

Audio Mixture Procedure

Same configuration → Analysis I

Same length in the audios before mixing

No delay

↓

200 mixtures

Transcriptions

GROUND TRUTH

Google Cloud
Speech to Text

amazon Transcribe

Canonicalize!

```
config = speech.RecognitionConfig(
    encoding = speech.RecognitionConfig.AudioEncoding.LINEAR16,
    sample_rate_hertz = 16000,
    audio_channel_count = 1,
    language_code = “en-US”,
    enable_word_confidence = True,
    model = “video”,
)
```

```
aws transcribe start-transcription-job
    region us-east-2
    media MediaFileUri
    language-code en-US
    output-bucket-name transcription-bucket
    output-key
```

Word Error Rate (*WER*)

$$WER = \frac{S + D + I}{N}$$

- S is the number of substitutions,
- D is the number of deletions,
- I is the number of insertions,
- N is the number of words in the reference

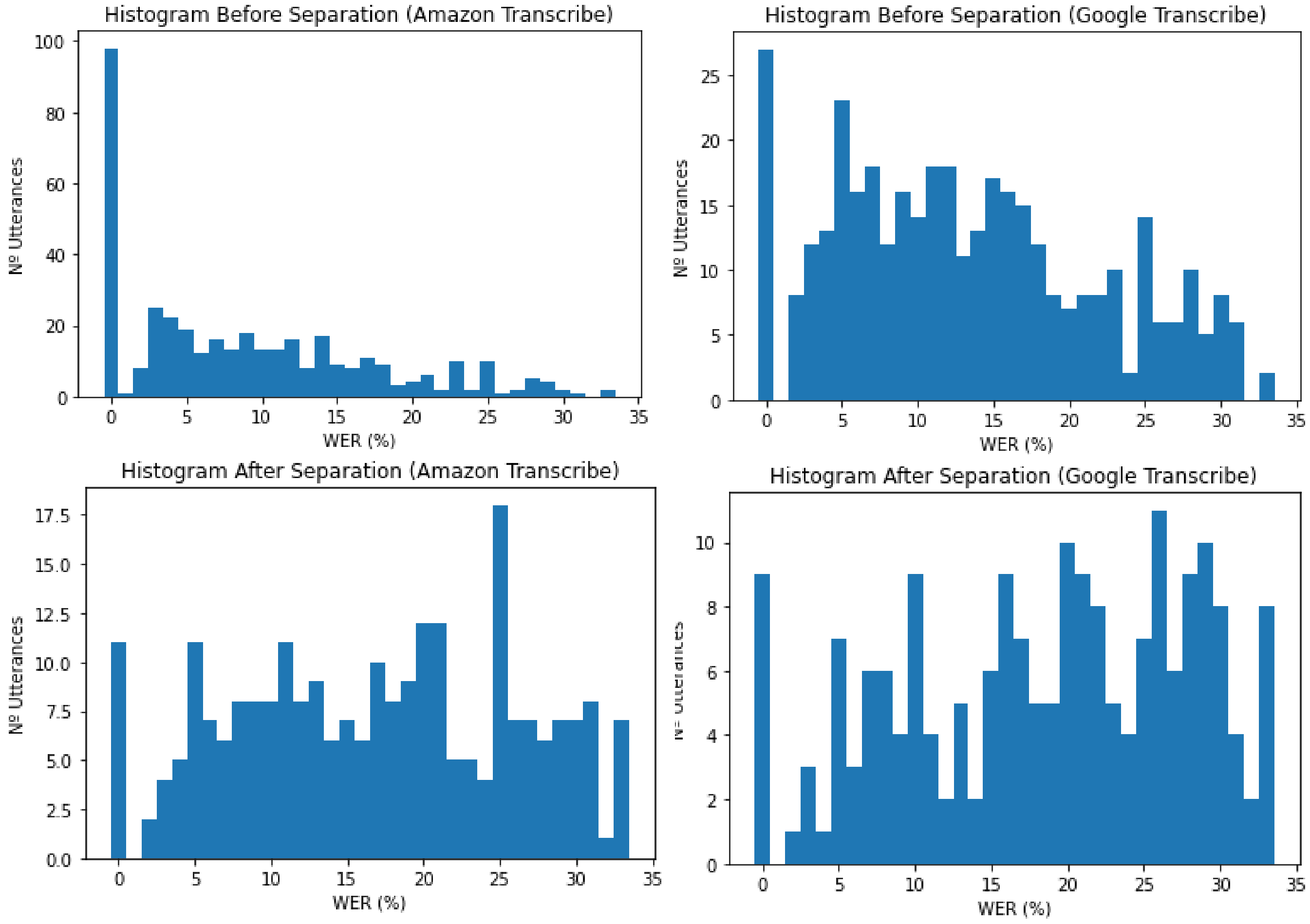
Substitution → "shipping" is transcribed as "sipping"

Deletion → "get it done" is transcribed as "get done"

Insertion → "hostess" is transcribed as "host is"

Word Error Rate (*WER*)

	Before	After
Human / Amazon	12.95 %	33.05 %
Human / Google	19.15 %	39.90 %



Word Error Rate (*WER*)

AFTER-SEPRATION (WER)	Amazon Transcribe	Google Transcribe
$WER \leq 15\ \%$	104	60
$15\ \% < WER < 35\ \%$	149	141
$WER \geq 35\ \%$	147	199

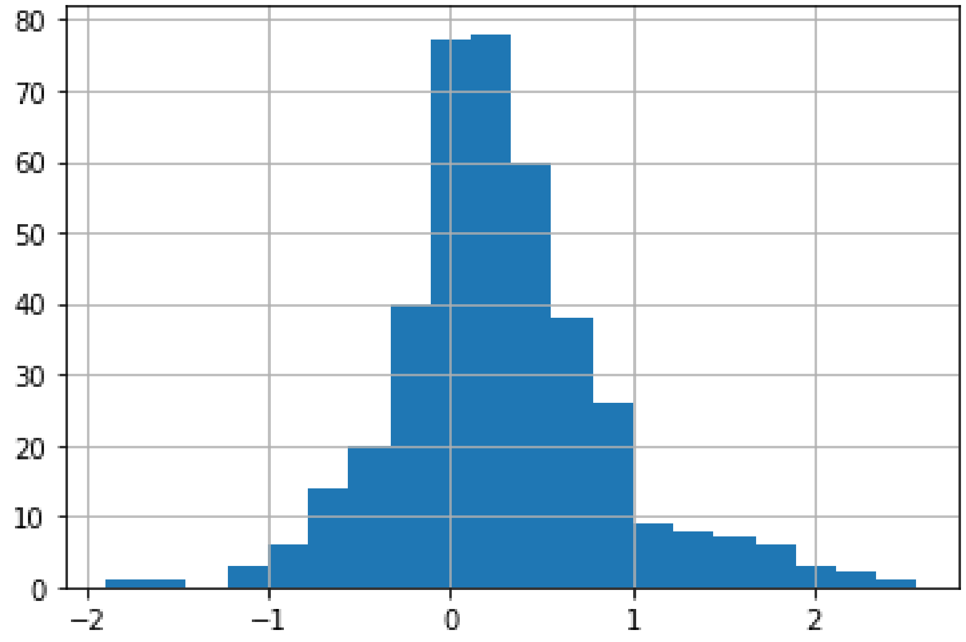
63 %

50 %

Mean Opinion Score Network (MOSNet)

MOSNet (before) → 3.1

MOSNet (after) → 2.8



Google Cloud
Speech to Text

amazon Transcribe

WER DECREASE

MOSNet INCREASE

3 transcripts

2 background noise
&
1 background music

Eliminated
After-Separation

Conclusions

The model works very well when separating male and female voices

The model detects noise and background music very accurately and it is often eliminated