UWB-NOTSOFAR: A System for Distant Meeting Transcription with a Single Device

Jan Lehečka, Zbyněk Zajíc, Marie Kunešová

NTIS – New Technologies for the Information Society, Faculty of Applied Sciences, Pilsen, Czech Republic

{jlehecka,zzajic,mkunes}@ntis.zcu.cz

Abstract

This technical description paper presents our enhanced system for the CHiME-2024 Challenge, Task 2 - NOTSOFAR-1, Track 1 (single-channel). Building on the baseline system, we implemented several improvements to tackle the challenge of overlapping speech recognition and speaker diarization. We integrate Overlapped Speech Detection (OSD) to differentiate between overlapping and non-overlapping speech segments. Non-overlapping segments were transcribed end-to-end by the ASR model while the overlapping segments were separated into streams and transcribed separately. Finally, all results were fused together producing the final hypotheses.

Index Terms: meeting transcription, overlapped speech detection, diarization, ASR

1. Introduction

Meeting recordings are known to be challenging data for automatic transcription (i.e., what words were spoken) and diarization (i.e., who spoke when) tasks. This paper describes our system used to transcribe meetings for the NOTSOFAR-1 challenge [1]. Our goal was to generate accurate transcriptions with correct assignment to the individual speakers in a computationally efficient way. We denote our system `UWB-NOTSOFAR` (UWB for the University of West Bohemia).

2. System Overview

Our system is based on a baseline system provided for the challenge by [1]. We used the same source separation at the beginning of the process and modified the next steps. In general, we made the following changes: (1) we used OSD (Overlapped Speech Detection) to break the task into two subtasks – (a) transcribe segments with speech overlaps by combining the information from separated streams and (b) transcribe segments without speech overlaps using pure ASR (Automatic Speech Recognition) and the original audio stream; (2) We improved the diarization by using speaker embeddings and (3) we fine-tuned the Whisper model on the train data from the challenge. In the following sections, we describe all the components of our system in detail.

3. Source Separation

Source separation was performed using the baseline system[1], using a slightly edited pipeline that only performs CSS. The output of this step was three separate audio streams for each input file.

The processing time of the evaluation data was 11m56s (716s) on a desktop PC (Intel Core i7-12700F; RTX 3060; Linux Mint 21.1).

4. Overlapped Speech Detection and Voice Activity Detection

Overlapped Speech Detection (OSD) and Voice Activity Detection (VAD) predictions were obtained using a single-task version of the wav2vec2-based approach proposed in [2]. Specifically, we used:

- wav2vec 2.0 [3] fine-tuned for audio frame classification,
- separate models for OSD and VAD,
- pre-trained wav2vec2 model for both tasks: ClTRUS[2],
- data for fine-tuning: the NOTSOFAR-1 training set `240415.1_train`, split into overlapping segments of 20s (10s overlap between neighboring segments), plus an equal number (19314) of randomly chosen speech segments from the training set of the AMI Meeting Corpus[3] [4] ("headset mix" recordings),
- VAD model: fine-tuned for 2 epochs, threshold 0.15,
- OSD model: fine-tuned for 8 epochs, threshold 0.1,
- all other parameters were as in the original paper.

VAD was performed 4 times for each input file: once for the original file and once for each of the three separated streams. OSD was performed only on the original file.

Intervals with detected overlapped speech were removed from the original (unseparated) audio files by overwriting them with silence (all zeroes). The intervals were also relabeled in the corresponding VAD. These removals were only done to the full audio file, not the separated streams.

The outputs of this step were:

- a list of intervals with overlapped speech in the original input file,
- an audio file with removed overlapped speech, plus the corresponding VAD,
- VAD for three source-separated streams.

Processing time of the evaluation data (real-time):

- preprocessing: 2m52s (splitting WAVs into shorter segments),
- VAD + OSD prediction: 37m42s (4x VAD, 1x OSD),

[1] `https://github.com/microsoft/NOTSOFAR1-CHALLENGE`

[2] `https://huggingface.co/fav-kky/wav2vec2-base-cs-80k-ClTRUS`

[3] `https://groups.inf.ed.ac.uk/ami/corpus`

- removing overlaps: 2m13s.

Total time for the evaluation data was 42m47s (2567s) on a desktop PC (Intel Core i7-12700F; RTX 3060; Linux Mint 21.1)

5. Diarization

For each input file, we performed 4 runs of speaker diarization using the NeMo toolkit[4]: once for the audio file with removed overlaps and once for each of the three separated streams.

NeMo settings: We used only the clustering diarizer step (without MSDD), with the `titanet_large` speaker model, configuration from official repository[5] and our own external VAD (obtained in the previous step). All other settings were as in the inference notebook[6].

The output of this step was 4 RTTM files for each input audio.

To enhance the diarization performance, we did the following post-processing. For each speaker in the original record `ch0.wav` (without overlapped speech), we compute an embedding `speechbrain/spkrec-ecapa-voxceleb`[5], the same for each stream. Each speaker from streams is mapped to the nearest speaker from the original recording (most similar embeddings from the original recording to each embedding from all streams) using normalized cosine similarity (norm similarity = the best similarity - second best similarity).

Total time for the evaluation data was 34m47s (2087s) on a desktop PC (Intel Core i7-12700F; RTX 3060; Linux Mint 21.1). In more detail: 1741s for NeMo + 308s to create the embeddings + 38s for mapping.

6. Automatic Speech Recognition

We applied Automatic Speech Recognition (ASR) on all four streams (original and three separated streams) per meeting. We used the `Whisper-large-v3` model[7] [6] additionally fine-tuned on the NOTSOFAR-train dataset. We used only speech segments without overlapping speech for the training. We fine-tuned the model for 2 epochs with a learning rate 1×10^{-7}. During the inference, we also kept word-level timestamps.

The total processing time on the eval dataset was 8835s (2h 27min) on a desktop PC (Intel Core i7-7800X) running on a GPU (NVIDIA GeForce RTX 2080 Ti).

The outputs of this step were the transcribed words with timestamps for each stream and each meeting record.

7. Fusion of Stream Results

In the final step, we combined all the information from the previous steps. Specifically, for each meeting record and microphone, we did the following steps:

1. First, we processed the ASR output from the original record stream (ch0.wav). We discarded all words falling into intervals with overlapped speech (output of OSD model) as the ASR model can't deal with overlapped speech reliably. Then, we assigned a speaker for each remaining word based on the RTTM intervals from the diarization process. We discarded all words outside the RTTM intervals, as there shouldn't be any speech.
2. Then, we inspected the intervals with overlapped speech. We found all words decoded from the three separated streams and assigned speakers based on the RTTM intervals from the diarization process. We used only RTTM intervals with normalized similarity > 0.1.
3. To filter out hallucinated words, we discarded all words with zero duration.
4. Finally, we collected all words assigned to each speaker, cleaned duplicate words (same words with some overlap), squeezed overlapping words, sorted them by their timestamps, and segmented them on pauses longer than 0.5s or on end-of-sentence tokens (".?!").

For meetings with multiple parallel recordings from different microphones, the process was performed *independently* for each microphone, as required by the rules of the single-channel track of the challenge.

The total processing time on the eval dataset was 41s on a desktop PC (Intel Core i7-7800X).

The output of this step was the file containing the final hypotheses submitted for evaluation.

8. Conclusion

Table 1: *Processing time of system's components. Note that processing times were not measured on exactly the same hardware, so the total sum is only an approximation.*

component	machine	GPU	time [hours]
source sep.	i7-12700	RTX 3060	0.20
OSD+VAD	i7-12700	RTX 3060	0.71
Diarization	i7-12700	RTX 3060	0.58
ASR	i7-7800X	RTX 2080 Ti	2.45
fusion	i7-7800X	-	0.01
TOTAL			3.96

The total processing time of transcribing the whole evaluation dataset was almost 4 hours (3:57:26). We tabulate the times of individual components in Tab. 1. Given that the total duration of all processed files in the evaluation dataset was 16.7 hours, our system works at a speed of about 4x faster than real-time on a common desktop PC with a low-end GPU.

9. References

[1] A. Vinnikov, A. Ivry, A. Hurvitz, I. Abramovski, S. Koubi, I. Gurvich, S. Peer, X. Xiao, B. M. Elizalde, N. Kanda *et al.*, "Notsofar-1 challenge: New datasets, baseline, and tasks for distant meeting transcription," *arXiv preprint arXiv:2401.08887*, 2024.

[2] M. Kunešová and Z. Zajíc, "Multitask detection of speaker changes, overlapping speech and voice activity using wav2vec 2.0," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[3] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[4] `https://github.com/NVIDIA/NeMo`

[5] `https://raw.githubusercontent.com/NVIDIA/NeMo/main/examples/speaker_tasks/diarization/conf/inference/diar_infer_meeting.yaml`

[6] `https://github.com/NVIDIA/NeMo/blob/main/tutorials/speaker_tasks/Speaker_Diarization_Inference.ipynb`

[7] `https://huggingface.co/openai/whisper-large-v3`

[4] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The AMI meeting corpus: A pre-announcement," in *International workshop on machine learning for multimodal interaction*. Springer, 2005, pp. 28–39.

[5] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong *et al.*, "Speechbrain: A general-purpose speech toolkit," *arXiv preprint arXiv:2106.04624*, 2021.

[6] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*. PMLR, 2023, pp. 28 492–28 518.