Computer Speech and Language 95 (2026) 101841

Contents lists available at ScienceDirect

Computer Speech & Language

journal homepage: www.elsevier.com/locate/csl

DiCoW: Diarization-conditioned Whisper for target speaker automatic speech recognition

Alexander Polok [a,*,1], Dominik Klement [a,b,1], Martin Kocour [a,1], Jiangyu Han [a], Federico Landini [a], Bolaji Yusuf [a], Matthew Wiesner [b,c], Sanjeev Khudanpur [b,c], Jan Černocký [a], Lukáš Burget [a]

[a] *Speech@FIT, Brno University of Technology, Czechia*
[b] *CLSP, Johns Hopkins University, USA*
[c] *HLTCOE, Johns Hopkins University, USA*

ARTICLE INFO

Dataset link: https://github.com/BUTSpeechFIT/TS-ASR-Whisper.

Keywords:
Diarization-conditioned Whisper
Target-speaker ASR
Speaker diarization
Long-form ASR
Whisper adaptation

ABSTRACT

Speaker-attributed automatic speech recognition (ASR) in multi-speaker environments remains a significant challenge, particularly when systems conditioned on speaker embeddings fail to generalize to unseen speakers. In this work, we propose Diarization-Conditioned Whisper (DiCoW), a novel approach to target-speaker ASR that leverages speaker diarization outputs as conditioning information. DiCoW extends the pre-trained Whisper model by integrating diarization labels directly, eliminating reliance on speaker embeddings and reducing the need for extensive speaker-specific training data. Our method introduces frame-level diarization-dependent transformations (FDDT) and query-key biasing (QKb) techniques to refine the model's focus on target speakers while effectively handling overlapping speech. By leveraging diarization outputs as conditioning signals, DiCoW simplifies the workflow for multi-speaker ASR, improves generalization to unseen speakers and enables more reliable transcription in real-world multi-speaker recordings. Additionally, we explore the integration of a connectionist temporal classification (CTC) head to Whisper and demonstrate its ability to improve transcription efficiency through hybrid decoding. Notably, we show that our approach is not limited to Whisper; it also provides similar benefits when applied to the Branchformer model. We validate DiCoW on real-world datasets, including AMI and NOTSOFAR-1 from CHiME-8 challenge, as well as synthetic benchmarks such as Libri2Mix and LibriCSS, enabling direct comparisons with previous methods. Results demonstrate that DiCoW enhances the model's target-speaker ASR capabilities while maintaining Whisper's accuracy and robustness on single-speaker data.

1. Introduction

The rapid development of deep learning techniques and vast increases in available training data and computing resources have made low-error-rate ASR systems on single speaker recordings viable at reasonable latencies (Li et al., 2022). Consequently, the research community has focused (Watanabe et al., 2020; Yu et al., 2022; Cornell et al., 2023, 2024b) on the more challenging task of multi-speaker ASR, whose goal is the accurate transcription of multiple speakers in a recording, including speaker-attributed ASR,

* Corresponding author.
E-mail address: ipoloka@fit.vut.cz (A. Polok).
[1] These authors contributed equally.

https://doi.org/10.1016/j.csl.2025.101841
Received 23 December 2024; Received in revised form 18 April 2025; Accepted 29 May 2025
Available online 13 June 2025

whose goal is also to assign a speaker label to each spoken word. In this work, we focus on speaker-attributed ASR by adapting a single-speaker ASR system using speaker diarization information to produce transcripts for all speakers.

Speaker-attributed ASR-systems that combine diarization and speech recognition usually operate in one of three ways: (a) running ASR and diarization independently, capturing the respective word and timing information from each, and using this timing information to assign words to speakers throughout a long conversation (Bhandari et al., 2024); (b) using a fully cascaded pipeline where various orderings of diarization, speaker extraction or source separation are combined with ASR (Yoshioka et al., 2019; Raj et al., 2021); and (c) using target speaker ASR (TS-ASR), where, in lieu of any source extraction or separation, the original audio is directly input along with speaker conditioning, so that the system transcribes the speech belonging to the conditioned speaker (Kanda et al., 2019).

The conventional approach to TS-ASR is to extract speaker embeddings corresponding to the target speaker and to have these embeddings as an auxiliary input to the ASR system (Karafiát et al., 2011; Huang et al., 2023). Although the use of speaker embeddings from a pre-trained speaker embedding extractor (Dehak et al., 2010; Snyder et al., 2018; Wang et al., 2023) can give clues to the ASR about what information in its input to utilize and what information to ignore, it implicitly requires learning to map speaker embeddings to ASR speech embeddings. Learning a robust mapping generalizing to new speaker sets at test time, especially when the speaker embedding extractor is trained independently, requires having a large number of speakers in the multi-speaker ASR training set. While this is manageable for simulated multi-speaker data (for which a large number of speakers can be obtained), data scarcity is arguably the most significant challenge for "in-the-wild" multi-speaker ASR; it is therefore, imperative to develop systems that can be efficiently trained on the order of tens of hours of real conversational data.

In this paper, we propose Diarization-Conditioned Whisper (DiCoW), a semi-end-to-end approach to speaker-attributed ASR where we condition Whisper (Radford et al., 2023) on diarization outputs, unlike prior approaches relying on speaker embeddings or specific modules to model speaker information. We use Whisper as the base ASR system in order to take advantage of its large-scale pretraining, multi-domain robustness, and long-form capabilities. However, we also present results with another "generic" single-speaker system, showcasing that our proposed method can attain strong results in combination with different ASR models. By directly using diarization labels, there is no need for the model to learn to map speaker and ASR embedding subspaces. This is accomplished by means of speaker conditioning in the form of time-speaker activity probabilities.

To this end, we propose a pair of methods for incorporating speaker activity information into Whisper.

The first, termed Query-Key Biasing, produces a mask for each target speaker from the diarization outputs. This mask is then used to modify the attention scores: the scores for frames that do not correspond to the desired speaker are attenuated, while those belonging to the target speaker are kept intact. This allows Whisper to focus on the target speaker's ASR in the presence of large regions of silence and non-target speaker speech. We note that the attention scores are modified with trainable parameters so that the attenuation of non-target-speaker positions is only enforced at the beginning of training, and the model is still afforded the flexibility to learn how much non-target information to keep.

In the second method, named Frame-Level Diarization Dependent Transformations (FDDT), the model is provided with more fine-grained access to the diarization output. Specifically, for each target speaker, an external diarizer categorizes speech frames into silence, target speaker, non-target speaker and overlapped speech. For each of the four categories and each Whisper encoder layer, a trainable affine transformation is introduced to transform the input frames belonging to the given category before they are fed into the next encoder layer. Thus, the model is equipped to learn how to handle the different regions of speech.

We experimentally validate our methods by fine-tuning Whisper on various datasets: NOTSOFAR-1 (Vinnikov et al., 2024), AMI (Mccowan et al., 2005), and Libri2Mix (Cosentino et al., 2020) using ground-truth speaker segmentation information. At inference time, we utilize speaker diarization labels generated by an end-to-end speaker diarization system with vector clustering (Kinoshita et al., 2021b; Bredin, 2023). We also evaluate our system on LibriCSS (Chen et al., 2020) without fine-tuning it on this dataset.

Our experiments on both real and synthetic datasets show that, without considerably degrading its single-speaker recognition performance, our proposed methods imbue Whisper with strong speaker-attributed ASR capabilities across datasets, even when automatic diarization is used for conditioning.

The rest of the paper is organized as follows: Section 2 provides coverage of related works. Section 3 presents background information on Whisper and our modifications to reduce its hallucination tendencies. Section 4 describes the methods we propose to enable diarization-conditioned Whisper to perform target-speaker ASR. Section 5 outlines the setup of our experiments, including datasets, metrics, and training details. Section 6 reports the results of our experiments. Section 7 discusses the strengths and weaknesses of the proposed systems. Finally, Section 8 concludes the paper with a summary of our findings.

2. Related works

Diarization-based ASR: The integration of ASR and diarization has been explored using different techniques. These include adding speaker role tokens during ASR decoding (Shafey et al., 2019), clustering speaker embeddings, and mapping them to ASR tokens (Kanda et al., 2022) — which requires deriving word timings for the tokens after decoding — and jointly producing ASR tokens, speaker tags, and timings (Cornell et al., 2024a). While training a single model from scratch to perform both tasks at once can exploit the synergies between "what is said" and "who said it", large training corpora are necessary, which are either only available to large companies or of synthetic nature in academic settings. In contrast, we focus in this work on leveraging existing pre-trained models to reduce the training burden.

Multi-speaker extensions of Whisper: Recent works extended Whisper for multi-speaker ASR. In Ma et al. (2024), the model is prompt-tuned with a target speaker embedding so that the model recognizes only the speech of that speaker. Alternatively, in Meng et al. (2024), Whisper is activated using speech from the target speaker instead of an embedding so that smaller modifications to the original architecture are required compared to Ma et al. (2024). In Guo et al. (2024), a "speaker-querying" module is added to produce speaker prompts that are used as inputs to the decoder. While similar in motivation, our work differs from Ma et al. (2024), Meng et al. (2024) and Guo et al. (2024) in that instead of using speaker embeddings or enrollment speech, we directly utilize speaker activities. This simplifies the interaction with external modules and elides any need for selecting and processing enrollment speech.

Lastly, this work extends our previous systems presented in the BUT/JHU CHiME-8 NOTSOFAR-1 Challenge (Polok et al., 2024) and Target Speaker ASR with Whisper (Polok et al., 2025). Specifically, we adhere to the NOTSOFAR-1 challenge conditions and build upon the experimental setup of those works. In this paper, we provide a more comprehensive analysis of various approaches for conditioning Whisper on diarization outputs—namely, input masking, Query-Key biasing (QKb), and Frame-Level Diarization Dependent Transformations (FDDT) methods. We perform new ablations to evaluate the behavior of these techniques immediately after initialization. Furthermore, we introduce a novel Co-Attention mechanism that enables interaction between previously independent TS-ASR branches, enhancing performance in overlapping speech scenarios. We also propose a vectorized joint CTC/attention decoding strategy for Whisper, demonstrating its effectiveness across conditions. In addition, we analyze how different diarization models impact the final system performance and show that our method generalizes well to other ASR backbones. Finally, we confirm that our system maintains strong performance even in single-speaker settings, ensuring its versatility beyond multi-speaker use cases.

3. Long-form modeling with whisper

OpenAI's Whisper (Radford et al., 2023) is an attention-based encoder–decoder model for automatic speech recognition and speech translation. The widely used model is trained on an order of magnitude more data than other open-source models, which was found to be the key to achieving state-of-the-art performance on a wide range of ASR benchmarks and popularized a number of useful features for ASR. Several speech processing tasks usually need to be performed in the real scenarios, and it might be relevant to perform them jointly, so Whisper is designed to be prompted with token-based control sequences in order to perform ASR and additionally return voice activity detection (VAD) or language identification (LID) decisions, among other complementary information.

Whisper incorporates previous text conditioning, where prior transcriptions are fed as context to the decoder. This feature enables effective processing of long-form audio, such as meetings, lectures, or podcasts, by maintaining context across extended recordings. Long-form audio processing is particularly relevant for multi-speaker ASR, which requires handling continuous dialogues rather than isolated utterances. However, extending Whisper to support multi-speaker scenarios introduces new challenges, such as managing overlapping speech and multi-speaker outputs. Leveraging Whisper as a foundation benefits from its extensive pre-training on large-scale data, reducing the need for additional task-specific training data.

In this work, we propose several extensions to Whisper that address these challenges and enable its application to multi-speaker ASR. The Whisper model is described in the rest of this section, and our proposed methods are detailed in Section 4.

3.1. Whisper

Whisper is available in variants ranging from 38M to 1.54B parameters and has been trained on up to 5 million hours of weakly (pseudo) labeled data. It employs an encoder–decoder Transformer (Vaswani et al., 2017) architecture, processing the log-Mel spectrogram as input $\mathbf{X} \in \mathbb{R}^{F\times T}$, where $T = 3000$ corresponds to 30 s of audio. Shorter audio segments are padded with zero signal, while longer ones are processed sequentially. The number of Mel frequency bins, F, is 80 in earlier versions and 128 in later versions.

The spectrogram is passed through two 1-dimensional convolutional layers that increase the feature dimension to d_m and downsample the sequence by a factor of two. The encoder layers transform these features into hidden representations $\mathbf{H} \in \mathbb{R}^{d_m\times T/2}$, which the decoder uses autoregressively to generate text tokens $\hat{y}$, conditioned on task-specific special tokens g. The process is formally defined as:

$$\mathbf{H} = \text{AudioEncoder}_{\phi_e}(\mathbf{X}), \quad \hat{y}_t = \text{TextDecoder}_{\phi_d}(g, \hat{y}_{1:t-1}, \mathbf{H}), \tag{1}$$

where ϕ_e and ϕ_d denote the encoder and decoder parameters, respectively.

Whisper incorporates two primary task-specific tokens: ⟨|transcribe|⟩ for transcription and ⟨|translate|⟩ for translation tasks. Additionally, the token ⟨|notimestamps|⟩ can be used to suppress the decoding of timestamps. The inclusion of language-specific tokens, such as ⟨|en|⟩ for English, enables Whisper to condition decoding for specific languages and tasks. Furthermore, Whisper supports previous text conditioning by allowing an optional sequence of tokens from prior decoding as input, facilitating context-aware transcription or translation.

In this study, we utilize the large-v3-turbo variant of Whisper in all our Whisper experiments. This version builds on prior work on distilling Whisper models (Gandhi et al., 2023) and reduces the number of decoder layers from 32 (large-v3) to 4 without significantly harming the model's performance. This architectural modification significantly reduces autoregressive decoding time, making it more practical for real-world applications. The large-v3-turbo model was trained on a mixture of 1 million hours of weakly labeled audio and 4 million hours of pseudo-labeled audio derived from the large-v3 model. In this work, we turn Whisper into a TS-ASR system by extending and adapting it to condition on diarization information in order to decode speakers of interest.

Note that the following two sections slightly deviate from the main idea of conditioning Whisper on diarization information for TS-ASR. However, they present orthogonal extensions of Whisper that we propose to enhance its capabilities. Specifically, the introduction of the CTC head serves as a general-purpose enhancement to Whisper.

3.2. CTC head for whisper

Building on the idea of a hybrid CTC/attention architecture (Hori et al., 2017; Watanabe et al., 2017), we propose to incorporate a connectionist temporal classification (CTC) (Graves et al., 2006) head into Whisper. To the best of our knowledge, we are the first to add a CTC head to an already pretrained and well-performing attention-based encoder–decoder (AED) model such as Whisper. Unlike prior work that trains CTC and AED jointly from scratch, our approach introduces the CTC objective only during fine-tuning. This encourages monotonic alignment between the input audio and predicted tokens and allows us to leverage intermediate encoder representations for efficient single-pass decoding or more accurate transcription when combined with the autoregressive decoder (Hori et al., 2017). Additionally, this setup naturally supports self-speculative decoding (Leviathan et al., 2023).

The proposed CTC head first applies a Transformer layer, followed by two 1-dimensional convolutional layers with a stride of 2, reducing the sequence length from 1500 to 375 (approximately matching the maximum sequence length in the decoder). This subsampling helps to reduce the computational overhead caused by the final linear projection, which maps the hidden representations from dimensionality d_m to vocabulary size $V \approx 50k$.

Formally, given the encoder's hidden representations $\mathbf{H} \in \mathbb{R}^{d_m \times T/2}$, the CTC head computes:

$$\mathbf{Z} = \mathrm{Linear}(\mathrm{Conv}(\mathrm{SelfAttention}(\mathbf{H}))), \tag{2}$$

where $\mathbf{Z} \in \mathbb{R}^{V \times T/8}$ represents the output logits.

3.3. Joint CTC/attention decoding with whisper

We observed improved convergence when the CTC head does not generate timestamp tokens, which led us to modify the ESPNet (Watanabe et al., 2018) CTC prefix scoring implementation.[2] Specifically, we adjusted the scoring procedure to retain current states without restoring the next tokens when the autoregressive decoder prefers timestamp tokens. This modification facilitates joint CTC/Attention decoding, wherein the CTC head operates over a distinct subset of labels, diverging from those used by the autoregressive attention-based decoder.

The decoding objective in this setup is defined as a combination of the sequence probabilities from the CTC and attention-based models. Let C be a sequence, $p_{\mathrm{ctc}}(C|\mathbf{X})$ be the sequence probability given by the CTC model, and $p_{\mathrm{att}}(C|\mathbf{X})$ be the sequence probability given by the attention-based model. The decoding objective is then formulated as:

$$\hat{C} = \arg\max_{C \in \mathcal{U}^*} \left(\lambda \log p_{\mathrm{ctc}}(C|\mathbf{X}) + (1-\lambda) \log p_{\mathrm{att}}(C|\mathbf{X}) \right), \tag{3}$$

where λ is a weight parameter controlling the balance between the CTC and attention model outputs (Hori et al., 2017) and $\mathcal{U}^*$ is the set of sequences, given by a beam of the top-k most likely hypotheses. Additionally, we streamlined the implementation by vectorizing key operations, improving computational efficiency, and enabling support for batched beam decoding. Our implementation is made publicly available.[3]

4. Diarization-conditioned whisper

In this section, we introduce the proposed approaches for conditioning on speaker activity. While Whisper serves as the base single-speaker ASR model for our explanations, these methods are generalizable and can be applied to other models, as shown in Section 6.6.

We begin by defining conditioning masks derived from the speech activity of each speaker, which serves as the foundation for all our proposed systems. We then describe three distinct mechanisms for utilizing these masks:

1. **Input masking**: directly masks the input audio based on speaker activity.
2. **Query-Key biasing (QKb)**: selectively biases the attention weights using information from the masks.
3. **Frame-Level Diarization Dependent Transformations (FDDT)**: incorporate the full masks to condition encoder representations in a more comprehensive manner.

It is important to highlight the growing parameter footprint associated with each diarization-conditioning method. Input masking introduces no additional parameters, making it the most lightweight approach. QKb increases the model size only slightly, adding $2N(2d_{\mathrm{model}}-1)$ parameters, where N is the number of layers to which it is applied. In contrast, FDDT is the most expressive but also the most parameter-intensive approach, potentially introducing up to $4N(d_{\mathrm{model}}^2+d_{\mathrm{model}})$ new parameters. This progressive increase highlights a trade-off between model complexity and potential performance gains.

[2] https://github.com/espnet/espnet/blob/master/espnet/nets/ctc_prefix_score.py.

[3] https://github.com/BUTSpeechFIT/TS-ASR-Whisper.

4.1. Silence, target, non-target, and overlap masks

Let $\mathbf{D} \in [0,1]^{S\times T}$, where S is the number of speakers in the recording, and T is the number of frames, represent the diarization output, with $d(s,t)$ denoting the probability that speaker s is active in time frame t. The dependency on the number of speakers in $\mathbf{D}$ can be a limiting factor for easily incorporating this mask into the model. To address this, we treat each speaker independently. Let s_k represent the target speaker. We define a distribution over the following mutually exclusive events for a frame at time t:

- $\mathcal{S}$: Time frame t represents silence.
- $\mathcal{T}$: The target speaker, s_k, is the only active speaker in time frame t.
- $\mathcal{N}$: One or more non-target speakers, $s \neq s_k$ are active and the target speaker, s_k, is not active at time frame t.
- $\mathcal{O}$: The target speaker s_k is active while at least one non-target speaker $s \neq s_k$ is also active at time frame t, denoting an overlap.

The probabilities of these events occurring at time frame t can be calculated as:

$$p^t_{\mathcal{S}} = \prod_{s=1}^{S}(1 - d(s,t)) \tag{4}$$

$$p^t_{\mathcal{T}} = d(s_k,t) \prod_{\substack{s=1\\ s\neq s_k}}^{S}(1 - d(s,t)) \tag{5}$$

$$p^t_{\mathcal{N}} = \left(1 - p^t_{\mathcal{S}}\right) - d\left(s_k,t\right) \tag{6}$$

$$p^t_{\mathcal{O}} = d(s_k,t) - p^t_{\mathcal{T}} \tag{7}$$

This definition allows us to use a fixed-sized STNO (Silence, Target, Non-target, Overlap) mask $\mathbf{M}^t = \begin{bmatrix} p^t_{\mathcal{S}} & p^t_{\mathcal{T}} & p^t_{\mathcal{N}} & p^t_{\mathcal{O}} \end{bmatrix}^\top$. Note that the mask is speaker-dependent, so decoding each target speaker involves using a different STNO mask, which results in a different transcript.

4.2. Input masking

Having the STNO mask, a straightforward way to perform target speaker ASR is to mask the signal by multiplying each frame by the probability that it is target speech or that it involves overlap with the target speaker. Hence, if the target speaker is not active, the audio signal is set to 0 (i.e., silence). We add $p^t_{\mathcal{T}}$ and $p^t_{\mathcal{O}}$ to ensure that both target speech and overlapping speech are preserved in the masked signal. Similar to source separation approaches, this method has limitations. It can introduce artifacts because we are creating a modified version of the input signal, and errors in diarization can propagate through the system, potentially affecting the model's performance.

4.3. Query-key biasing conditioning

An alternative approach to steer the model's attention away from non-target segments is to integrate target-speaker masks with the model's internal representations by incorporating them into encoder/decoder attention masks. Compared to the input masking method, attention masking does not introduce artificial silence in the audio signal, reducing the chance of artifacts and potentially leading to better performance.

For simplicity, let us assume a single attention head. Let $\mathbf{W}_q, \mathbf{W}_k \in \mathbb{R}^{d_m\times d_m}$ denote the query and key projection matrices and $\mathbf{q}_i, \mathbf{k}_j \in \mathbb{R}^{d_m}$ the query and key, respectively. The unnormalized attention score between $\mathbf{q}_i, \mathbf{k}_j$ is defined as:

$$a_{ij} = \frac{(\mathbf{W}_q\mathbf{q}_i)^T(\mathbf{W}_k\mathbf{k}_j)}{\sqrt{d_m}}. \tag{8}$$

To obtain normalized attention weights, the softmax function is applied across the j-dimension of a_{ij}

If we assume that acoustic information is aligned across time, masking out non-target speaker frames forces the model to ignore information irrelevant to the target speaker transcript (i.e., other speakers, silence, etc.). However, "hard" attention masking leaves the model no chance for unmasking and possibly attending to non-target frames, which makes adaptation and speaker-tracking learning impossible.

As a solution, we decided to bias the encoder self-attention and the decoder cross-attention by extending queries and keys, and initializing corresponding projections in the following way:

$$\hat{\mathbf{q}}_i = \begin{bmatrix}\mathbf{q}_i\\ 1\end{bmatrix}, \hat{\mathbf{k}}_j = \begin{bmatrix}\mathbf{k_j}\\ -c\end{bmatrix}, \hat{\mathbf{W}}_q = \begin{bmatrix}\mathbf{W}_q & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}, \hat{\mathbf{W}}_k = \begin{bmatrix}\mathbf{W}_k & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}, \tag{9}$$

where $c \in \mathbb{R}^+_0$ is a bias factor set to 0 if k_j corresponds to a target speaker frame, and to a predefined constant otherwise. We name this approach query-key biasing conditioning (QKb).

It is easy to observe that, after initialization, if k_j represents a target speaker frame, a_{ij} remains intact. On the other hand, if k_j represents a non-target speaker frame, the calculation of a_{ij} changes as:

$$\hat{\mathbf{W}}_q\hat{\mathbf{q}}_i = \begin{bmatrix}\mathbf{W}_q\mathbf{q}_i \\ 1\end{bmatrix}, \quad \hat{\mathbf{W}}_k\hat{\mathbf{k}}_j = \begin{bmatrix}\mathbf{W}_k\mathbf{k}_j \\ -c\end{bmatrix}, \tag{10}$$

$$\begin{bmatrix}(\mathbf{W}_q\mathbf{q}_i)^T & 1\end{bmatrix}\begin{bmatrix}\mathbf{W}_k\mathbf{k}_j \\ -c\end{bmatrix} = (\mathbf{W}_q\mathbf{q}_i)^T(\mathbf{W}_k\mathbf{k}_j) - c. \tag{11}$$

It is important to note that fine-tuning the Whisper model with extended queries and keys changes the extended attention projection matrices $\hat{\mathbf{W}}_q$ and $\hat{\mathbf{W}}_k$, which controls the level of attention biasing.

4.3.1. Shifted positional embeddings

Masking parts of the input using the proposed QK biasing technique within an utterance prevents the Whisper decoder from cross-attending to the entire sequence of encoder embeddings. Hence, the decoder does not have a chance to see all the positional embeddings, creating discontinuities in the attended sequence. Such breaks in the continuity of positional embeddings can disrupt the model immediately after initialization and cause instabilities during training, particularly in its early stages. To mitigate this, an ad-hoc solution that seems to help is to shift the positional embeddings on target speaker frames and repeat the previous positional embedding on non-target frames (e.g., for a sequence like TTTNNTT, the positional embeddings represent frames 1 233 345) instead of applying the original sequence of positional embeddings. This ensures continuity in positional embeddings for the frames the decoder attends to, aligning with how Whisper was originally trained. Since the decoder is not expected to attend to non-target frames, their positional embeddings should not influence the transcription. Repeating the last positional embedding on non-target frames is therefore expected to perform comparably to alternative strategies such as interpolation of positional embeddings or fixed non-target embeddings.

4.4. Frame-level diarization dependent transformations

The third and most sophisticated approach to diarization conditioning is frame-level diarization dependent transformations (FDDT) depicted in Fig. 1. It modifies the frame-by-frame model inputs based on the diarization outputs and unlike the previous methods, it uses all four STNO masks.

Let $\mathbf{Z}^l \in \mathbb{R}^{d_m \times T}$ represent the frame-by-frame inputs to the lth (Transformer) layer. We transform these hidden representations by applying four affine STNO layer- and class-specific transformations: $\mathbf{W}^l_{\mathcal{S}}, \mathbf{W}^l_{\mathcal{T}}, \mathbf{W}^l_{\mathcal{N}}, \mathbf{W}^l_{\mathcal{O}} \in \mathbb{R}^{d_m \times d_m}$ together with biases $\mathbf{b}^l_{\mathcal{S}}, \mathbf{b}^l_{\mathcal{T}}, \mathbf{b}^l_{\mathcal{N}}, \mathbf{b}^l_{\mathcal{O}} \in \mathbb{R}^{d_m}$ to obtain new speaker-specific hidden representations $\hat{\mathbf{Z}}^l = [\hat{\mathbf{z}}^l_1, \ldots, \hat{\mathbf{z}}^l_T]$ as:

$$\begin{aligned}\hat{\mathbf{z}}^l_t &= \left(\mathbf{W}^l_{\mathcal{S}}\mathbf{z}^l_t + \mathbf{b}^l_{\mathcal{S}}\right) p^t_{\mathcal{S}} + \left(\mathbf{W}^l_{\mathcal{T}}\mathbf{z}^l_t + \mathbf{b}^l_{\mathcal{T}}\right) p^t_{\mathcal{T}} \\ &+ \left(\mathbf{W}^l_{\mathcal{N}}\mathbf{z}^l_t + \mathbf{b}^l_{\mathcal{N}}\right) p^t_{\mathcal{N}} + \left(\mathbf{W}^l_{\mathcal{O}}\mathbf{z}^l_t + \mathbf{b}^l_{\mathcal{O}}\right) p^t_{\mathcal{O}}.\end{aligned} \tag{12}$$

In other words, the hidden representations $\mathbf{z}^l_t$ are transformed using a convex combination of the four STNO class-specific affine transformations, weighted by the corresponding STNO class probabilities (4)–(7). When using a hard STNO mask (i.e., one-hot encoding), Eq. (12) simplifies to selecting and applying one of the four class-specific transformations for each frame. Note that the same transformation is applied to all frames with identical STNO masks.

The matrices $\mathbf{W}^l_{\mathcal{S}}, \mathbf{W}^l_{\mathcal{T}}, \mathbf{W}^l_{\mathcal{N}}, \mathbf{W}^l_{\mathcal{O}}$ are designed to transform the hidden representations into a space where speaker distinction is more effective, or where certain components of the signal can be suppressed. These transformations are essential for the model to correctly identify and isolate the target speaker while handling other sources of noise.

In prior work, we have shown that random initialization could disrupt the model's already learned internal representations (Polok et al., 2025), leading to a significant drop in performance. To mitigate this risk, we employ a suppressive initialization strategy. In this strategy, we initialize $\mathbf{W}^0_{\mathcal{S}}$ and $\mathbf{W}^0_{\mathcal{N}}$ as zero matrices, which effectively suppresses the influence of other speakers; while the other parameters $\mathbf{b}^l_{\mathcal{S}}, \mathbf{b}^l_{\mathcal{T}}, \mathbf{b}^l_{\mathcal{N}}, \mathbf{b}^l_{\mathcal{O}}$ (set to zero vectors), and $\mathbf{W}^l_{\mathcal{T}}, \mathbf{W}^l_{\mathcal{O}}$ (set to identity matrices)—are initialized to maintain the original flow of information. This initialization strategy was motivated by our earlier study, where we analyzed various initialization schemes. As shown in Table 4 of this paper, suppressive initialization (FDDT init.) yields better performance immediately after initialization compared to identity initialization (Whisper), further confirming its effectiveness.

4.5. Co-attention module for speaker interaction in TS-ASR

So far, we have discussed diarization conditioning approaches for TS-ASR, where each speaker is processed independently by a separate TS-ASR instance. While this parallel setup is efficient and works well for clearly separated speech, it can struggle in complex conversational scenarios. The key limitation is the lack of interaction or information sharing between speaker streams, which becomes especially problematic when speaker turns are ambiguous or overlapped. To address this, we introduce a Co-Attention module designed to explicitly model inter-speaker dependencies and promote interaction between the speaker-specific representations. Originally introduced for speaker diarization (Horiguchi et al., 2022) and later extended to speaker identification (Mošner et al., 2024), Co-Attention enables the model to compare, contextualize, and coordinate information across multiple speaker channels.

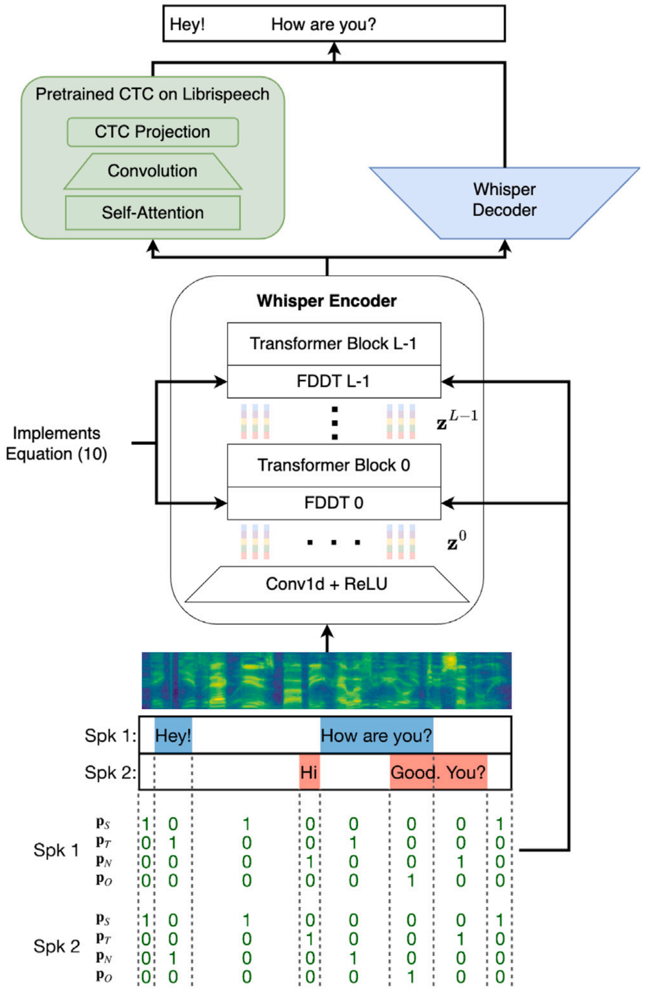


Fig. 1. Proposed diarization-conditioned whisper model with STNO mask example.

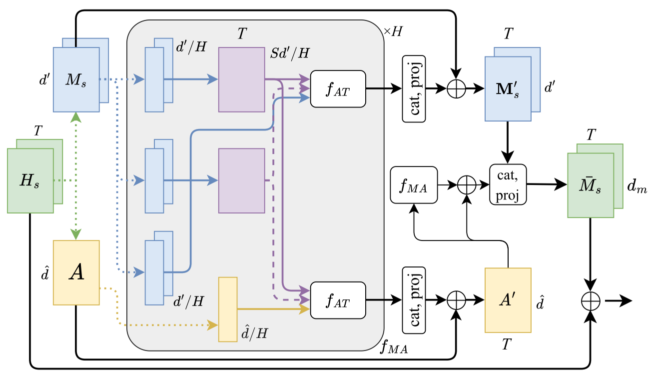


Fig. 2. Scheme of Co-Attention module. Dotted lines depict affine transformation followed by layer normalization. The gray rectangle represents the multi-headed attention derived in (13). Inputs to scaled-dot product f_{AT} attention are queries, keys and values from top to bottom.

Fig. 2 illustrates the architecture of the Co-Attention module, which is structured into three stages: *summarization*, *temporal alignment*, and *contextualization*. The core idea is to first compute a global summary of the speech context by averaging representations across all speaker streams. This is followed by a shared temporal attention mechanism that aligns speaker representations in time. Finally, each speaker stream is enriched with information from both the global summary and its own channel. This design allows the model to encode salient or dominant features from one speaker into the shared representation, effectively signaling to other streams that this information has already been captured. In addition, shared temporal attention promotes alignment across speaker streams.

We follow the Co-Attention formulation introduced by Mošner et al. (2024), adapted for TS-ASR. We denote multi-head attention (MHA) as:

$$f_{\mathrm{MA}}(\mathbf{X},\mathbf{Y};\theta_{\mathrm{MA}}) := \mathbf{W}_O \left[\mathop{\Big\|}_{h=1}^{H} f_{\mathrm{AT}}\left(\mathbf{W}_Q^{(h)}\mathbf{X}, \mathbf{W}_K^{(h)}\mathbf{X}, \mathbf{W}_V^{(h)}\mathbf{Y}\right)\right], \tag{13}$$

where $f_{\mathrm{AT}}(\mathbf{Q},\mathbf{K},\mathbf{V})$ denotes scaled dot-product attention (Vaswani et al., 2017), and $\mathop{\|}_{h=1}^{H}$ represents concatenation along the head dimension. The parameter set θ_{MA} includes all learnable weights: $\{\mathbf{W}_Q^{(h)}, \mathbf{W}_K^{(h)}, \mathbf{W}_V^{(h)}\}_{h=1}^{H} \cup \{\mathbf{W}_O\}$.

Summarization. Given encoder outputs $\{\mathbf{H}_s\}_{s=1}^{S}$, with $\mathbf{H}_s \in \mathbb{R}^{d_m \times T}$ for each speaker s, we compute the global summary as:

$$\mathbf{A} = \mathrm{LN}\left(\mathbf{W}_A \left[\frac{1}{S}\sum_{s=1}^{S}\mathbf{H}_s\right]\right), \quad \mathbf{W}_A \in \mathbb{R}^{\hat{d}\times d_m}, \tag{14}$$

where $\mathrm{LN}(\cdot)$ denotes layer normalization. Each speaker's embedding is also projected individually:

$$\mathbf{M}_s = \mathrm{LN}\left(\mathbf{W}_M \mathbf{H}_s\right) \in \mathbb{R}^{d'\times T}, \tag{15}$$

and all speaker embeddings are stacked together:

$$\mathbf{M} = \mathop{\Big\|}_{s=1}^{S} \mathbf{M}_s \in \mathbb{R}^{Sd'\times T}. \tag{16}$$

Temporal alignment. We apply shared temporal attention over all speaker channels using block-diagonal shared projection matrices. The speaker-specific and global summaries are enriched as:

$$\mathbf{M}'_s = \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{M},\mathbf{M}_s;\theta) + \mathbf{M}_s\right), \tag{17}$$

$$\mathbf{A}' = \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{M},\mathbf{A};\xi) + \mathbf{A}\right), \tag{18}$$

where θ and ξ share $\mathbf{W}_Q^{(h)}$ and $\mathbf{W}_K^{(h)}$, enforcing identical attention weights for temporal alignment across speaker streams.[4]

Contextualization. To refine the global summary, we apply self-attention:

$$\bar{\mathbf{A}} = \mathrm{LN}\left(f_{\mathrm{MA}}(\mathbf{A}',\mathbf{A}';\omega) + \mathbf{A}'\right). \tag{19}$$

Then, for each speaker, we fuse the co-attended embedding with the contextualized global summary:

$$\bar{\mathbf{M}}_s = \mathbf{W}_F\left(\mathbf{M}'_s \,\|\, \bar{\mathbf{A}}\right), \quad \mathbf{W}_F \in \mathbb{R}^{d_m\times(d'+\hat{d})}. \tag{20}$$

Finally, the output of the Co-Attention module is added residually:

$$\hat{\mathbf{H}}_s = \mathbf{H}_s + \bar{\mathbf{M}}_s. \tag{21}$$

5. Experimental setup

This section details our experimental setup, including datasets, evaluation metrics, training procedure, hyperparameters, and lastly, the diarization system. All our models were implemented in HuggingFace Transformers library (Wolf et al., 2020). We used Whisper-large-v3-turbo[5] as it is a faster version of large-v3, and we observed almost no performance degradation.

5.1. Evaluation datasets

To validate the proposed method, we utilized two publicly available multi-speaker datasets: AMI (Mccowan et al., 2005) and NOTSOFAR-1 (Vinnikov et al., 2024). Both have realistic interactions in a far-field setting, presenting English-spoken meetings in a challenging scenario. Statistics about the sets can be found in Tables 1 and 2. To be able to compare with other existing methods, we also utilized Libri2Mix (Cosentino et al., 2020) and LibriCSS (Chen et al., 2020).

Given the current state of technology and the availability of public datasets, we believe results should be reported on real and not synthetic datasets.

[4] In both (17) and (18), $\mathbf{M}$ is used as both query and key source. The shared key/query projections are block-diagonal matrices built from per-speaker components in $\mathbb{R}^{d'/H\times d'}$.

[5] https://huggingface.co/openai/whisper-large-v3-turbo.

Table 1
Numbers of files, minimum and maximum numbers of speakers per recording, and numbers of hours per partition.

Dataset	Train			Development			Test		
	#files	#spk	# h	#files	#spk	# h	#files	#spk	# h
AMI	136	3–5	80.67	18	4	9.67	16	3–4	9.06
NOTSOFAR-1	526	4–8	54.27	117	4–6	12.17	160	3–7	16.67
Libri2Mix	13 900	2	56.37	3000	2	7.6	3000	2	7.01
LibriCSS	–	–	–	7	8	1.0	55	8	9.09

Table 2
Percentage of silence, speech with a single speaker, and overlap for each set.

Dataset	Train			Development			Test		
	%sil	%1-spk	%ov	%sil	%1-spk	%ov	%sil	%1-spk	%ov
AMI	16.5	72.3	11.2	22.0	61.1	16.9	14.7	67.9	17.4
NOTSOFAR-1	8.1	65.4	26.5	17.7	68.8	13.7	8.0	66.6	25.2
Libri2Mix	5.5	33.9	60.6	8.4	42.6	49.0	8.1	42.8	49.1
LibriCSS	–	–	–	6.2	84.2	9.6	6.7	83.7	9.6

5.2. Evaluation metrics

Word error rate (WER) is normally used to evaluate single-speaker ASR. It calculates the error of a hypothesis with respect to the reference as the sum of substitutions, insertions, and deletions over the number of words in the reference annotation. However, with recordings having more than one speaker, different recognizers are usually evaluated with different metrics depending on how the hypotheses and references are mapped. In order to be able to compare with relevant previous works, we considered:

- Concatenated minimum-permutation WER (cpWER), where for each speaker, all their utterances are concatenated, and the best permutation between hypothesis and reference is used to calculate the standard WER. This metric takes into account speaker-attributed ASR errors.
- Time-constrained minimum-permutation WER (tcpWER), where the evaluation is like cp-WER but also considering the temporal alignments of the words.
- Optimal reference combination WER (ORC-WER), which does not consider the speaker labels and can be used to evaluate speaker-agnostic systems.
- Time-constrained optimal reference combination WER (tcORC-WER), where the evaluation optimally matches hypotheses and references without considering speaker labels but ensures that temporal alignments are respected.

To avoid cluttering tables with notation, we do not write ‘WER’ in the headers below. For a more thorough analysis and comparison of the metrics, we refer the reader to Neumann et al. (2023). For time-constrained metrics, we use a collar of 5 s. The diarization error rate (DER), used to evaluate the diarization system, is calculated using collar 0 s.

5.3. Training details

The training is divided into three consecutive phases:

1. **CTC preheat** - pre-train only the CTC-related parameters on LibriSpeech 960 h (Panayotov et al., 2015) with the rest of the model being frozen. CTC is trained without timestamps as we believe that timestamp prediction should not occur in the encoder and can rather be heuristically derived from the CTC frame-by-frame predictions.
2. **FDDT preheat** - pre-train CTC and FDDT-related parameters on the target multi-speaker dataset. We train FDDT parameters with 100× higher learning rate than the rest (i.e., 2×10^{-5}) to improve the model convergence.
3. **Full fine-tuning** - fine-tune all the parameters on the target multi-speaker dataset until convergence.

We train all the models with an overall batch size of 64 samples using AdamW (Loshchilov and Hutter, 2019) optimizer with weight decay 1×10^{-6}. We warm up the learning rate for 5k steps (i.e., 5k per batch model updates) and then use a linear decay scheduler for the rest of the training. The peak learning rate is set to 2×10^{-7}. The CTC loss weight is set to $\lambda = 0.3$ (Watanabe et al., 2017). We evaluate the model on the development set at intervals of $\min(1 \text{ epoch}, 500 \text{ steps})$, monitor the development tcpWER, and use early stopping with a patience of 5 evaluation steps. The maximum number of training steps is set to 50k, and we select the final model based on the best development tcpWER. Please note that CTC Preheat training is independent of FDDT preheat and full fine-tuning. For CTC Preheat, we use only 1k warm-up steps with a peak learning rate of 2×10^{-4}, monitoring WER on Librispeech dev-clean and dev-other sets. Other hyperparameters, including the batch size and optimizer settings, remain unchanged.

We do not enforce true casing or lower casing. Instead, we compute the cross-entropy loss with both lowercase and uppercase labels and select the smaller loss value. We empirically chose the initial value of the QK biasing constant c (9)–(11) to be 50 as we observed that higher values result in fewer hallucinations at the beginning of training, while also making it easier for the model to approximate a hard attention mask by producing very low attention scores without having to scale up the query and key projection weights.

5.4. Diarization system

We utilized DiariZen (Han et al., 2025)[6] — a framework with local end-to-end neural diarization (EEND) followed by speaker embedding clustering (Kinoshita et al., 2021b,a) using pyannote (Bredin, 2023; Plaquet and Bredin, 2023). The EEND module combines WavLM (Chen et al., 2022) with Conformer (Gulati et al., 2020) layers and is trained using the powerset loss (Plaquet and Bredin, 2023). The EEND operates on 8-second-long overlapping segments, and for each speaker found in the segment, a speaker embedding is extracted using a ResNet34-based embedding extractor (Wang et al., 2023). These are, in turn, clustered by spectral clustering (Park et al., 2019) to obtain the inter-segment mapping between speakers and produce a single output for each recording. The hard decisions made by the diarization system are used to provide the corresponding inputs for the following ASR system. Note that we use the same model on all evaluation sets. This model is compliant with the CHiME-8 challenge rules (Vinnikov et al., 2024) for the NOTSOFAR-1 track. Fine-tuning the model on each specific dataset could lead to further improvements.

6. Experiments

This section presents a comparison of the proposed methods. We first evaluate our best configuration against prior work using both ground-truth and system diarization. To remain consistent with previous literature, Table 3 reports results using non-time-constrained metrics (cpWER and ORC-WER) for AMI, Libri2Mix, and LibriCSS—datasets where these metrics are commonly used. Here, cpWER measures speaker-attributed transcription quality, while ORC-WER assesses overall transcription accuracy, independent of speaker attribution.

However, these metrics overlook errors in utterance timing, which we consider a critical aspect of TS-ASR performance. To address this, Table 4 and those that follow additionally report time-constrained metrics (tcpWER and tcORC-WER), aligned with the NOTSOFAR-1 challenge setup.

As evaluation protocols vary across datasets and prior work, we report both metric types for completeness. For future comparisons, we encourage the use of time-constrained metrics, which better reflect the practical demands of TS-ASR. In cases where ORC-WER could not be directly computed, we approximate it by increasing the time collar in tcORC-WER until convergence, providing a conservative upper bound.

First, we focus on comparing the proposed variants of diarization conditioning, presenting results across all datasets using ground-truth diarization, followed by an analysis of the impact of real diarization on the results. Next, we analyze the effect of joint CTC training and decoding on the NOTSOFAR-1 dataset. We also report the impact of target-speaker training on single-speaker evaluation datasets. Finally, we show the proposed diarization conditioning when using a different architecture to showcase that the method can be applied to other ASR models.

6.1. Comparison to baselines

In this section, we begin by analyzing the performance of our best-performing models (Table 3), and comparing them to state-of-the-art results reported in the literature. Specifically, we use the best-performing variants of our approach: the multi-domain (MD) FDDT model for AMI-sdm, NOTSOFAR-1, and LibriCSS, and the single-domain (SD) fine-tuned version with Co-Attention for Libri2Mix. These configurations were selected to highlight our most effective methods on each dataset. For a more detailed comparison of all proposed methods, refer to Table 4.

Even though comparisons across datasets are inherently challenging, as different studies report results on different datasets, our models demonstrate consistently strong performance. On AMI-sdm, to the best of our knowledge, we achieve state-of-the-art results with ground-truth diarization. Notably, the degradation when switching to real diarization is relatively small for ORC-WER. However, for cpWER, the impact is significantly larger, indicating that errors related to speaker label assignment (linked to the confusion errors in DER) have a more substantial effect than inaccuracies in segment boundaries.

Our proposed method also achieves the best results to date on the Libri2Mix and LibriCSS datasets for both real- and ground-truth diarization. However, while these results are noteworthy, we consider evaluations on artificial datasets like Libri2Mix to be of limited practical relevance.

[6] https://github.com/BUTSpeechFIT/DiariZen.

Table 3
Comparison of the proposed system alongside various multi-speaker ASR systems. The top section includes systems where no additional information about speaker identity or segmentation is provided. Our results are obtained using a diarization system. The bottom section features models that directly or indirectly utilize ground-truth (oracle) diarization information. Proposed ORC WER results marked with ⋆ were approximated by increasing the collar for tcORC WER. Results marked with † are evaluated on utterance groups, where the model is not penalized for speaker confusions outside the window of the utterance group, which significantly reduces cpWER.

	AMI-sdm		NOTSOFAR-1		Libri2Mix				LibriCSS	
	Test		Eval-small		Test-clean		Test-both		Test	
	cp	ORC	tcp	tcORC	cp	ORC	cp	ORC	cp	ORC
Kanda et al. (2021)	21.2†									
Raj et al. (2023)		44.6†								16.9
Fazel-Zarandi and Hsu (2023)						7.8				
Vinnikov et al. (2024)			41.4	35.5						
Cornell et al. (2024a)	24.5									
Niu et al. (2024)			22.2	17.7						
Ours (real diar.)	23.6	18.0⋆	33.5	22.6	6.0	6.0	15.0	14.9	8.5	6.5⋆
Cornell et al. (2024a)	21.1									
Ma et al. (2024)					12.0		26.4			
Zhang and Qian (2023)							23.5			
Meng et al. (2024)					4.7					
Guo et al. (2024)	22.0						14.6			
Ours (oracle diar.)	17.2	16.5⋆	19.7	19.1	4.4	4.4	10.9	10.9	5.6	5.5⋆

6.2. Input masking vs. QK biasing vs. FDDT

Table 4 presents the performance of the methods proposed in Section 4. It can be seen that the out-of-the-box Whisper model does not perform well, as it lacks a mechanism to prevent transcribing all present speech, irrespective of who is considered the target speaker. By masking the non-target speaker audio (Input masking), we improve on all datasets substantially.

Furthermore, we can see that QK biasing after initialization (and before fine-tuning) does not perform well, reaching WER metrics above 100%, suggesting strong levels of hallucination. After fine-tuning, we can see that not shifting positional embeddings results in better performance with regard to both tcpWER and tcORC WER. We further analyze the difference between these two modifications in Table 5.

Lastly, the third section of Table 4 presents FDDT. First, it can be observed that the method performs comparably to input masking right after suppressive initialization, suggesting that the initialization does not break the original Whisper model compared to QK biasing. Furthermore, after single-domain (SD) fine-tuning (i.e., fine-tuning the model only on the corresponding training set), we can observe a massive improvement on all the datasets.

Our model faces a challenge with fully overlapped speech, such as in Libri2Mix, where two speakers talk concurrently most of the time. This limitation arises from the model design, where each target speaker is decoded by an independent instance of the TS-ASR model. With fully overlapped speech, it might be difficult for an independent TS-ASR instance to determine which speaker's speech is responsible for decoding. As a consequence, multiple TS-ASR instances can decide to decode speech from the same speaker. To mitigate this, we incorporate a Co-Attention mechanism that allows the model to compare information across target speaker channels (i.e., TS-ASR instances). This mechanism allows the channels to collaboratively decide which instance is responsible for decoding each speaker in the input utterance. By resolving this ambiguity, Co-Attention reduces errors on Libri2Mix by 2%–3% absolute. However, the improvement is not as significant on AMI or NOTSOFAR-1, as these datasets have a significantly lower percentage of overlapped speech, and also, more speakers are present, which makes the scenario more challenging.

FDDT MD refers to a multi-domain model, which utilizes training data from AMI-sdm, NOTSOFAR-1, and Libri2Mix weighted with a ratio of 4:4:1. We selected the best-performing checkpoint based on the NOTSOFAR-1 development set. It can be seen that the model outperforms the other approaches on both real datasets. However, it performs worse on Libri2Mix, suggesting the domain mismatch between real-world and synthetic mixtures. It also demonstrates that the increased amount of data is not solving the full overlap issue and that a speaker interaction module is indeed necessary.

The addition of Co-Attention to the MD FDDT model slightly improves performance on Libri2Mix compared to MD FDDT alone, indicating its potential to better handle fully overlapped speech in synthetic mixtures. However, it remains inferior to the performance demonstrated by SD + Co-Attention on Libri2Mix, where the smaller, more targeted domain training likely aligns better with the dataset's specific characteristics. On the other hand, MD FDDT + Co-Attention does not offer improvements on AMI or NOTSOFAR-1, which may be attributed to the variability in the number of speakers and the dynamic interaction patterns in these real-world datasets. This variability affects the normalization of attention scores within the Co-Attention module. This suggests that while Co-Attention provides some benefits in specific scenarios, its integration with MD FDDT may require further design changes to handle diverse speaker configurations more effectively.

Furthermore, Fig. 3 shows the evolution of tcpWER of QK biasing and FDDT. It can be seen that FDDT converges much quicker than QK biasing, which is mainly caused by the non-disturbing initialization. After 1000 steps, the FDDT approach reaches tcpWER below 30%, suggesting that FDDT is an effective module that quickly turns Whisper into a target-speaker model.

Table 4
Comparison of different diarization-conditioning methods. The table is divided into three sections: the first section presents the performance of vanilla Whisper and Input Masking methods, which do not require additional training; the second and third sections show the results for QK biasing and FDDT with different configurations. All methods are evaluated using ground-truth diarization to isolate the effect of diarization conditioning from diarization errors. An asterisk (*) indicates values that could not be computed due to scalability issues, while a dash (–) denotes single-domain setups (LibriCSS) where training data is unavailable.

	AMI-sdm		NOTSOFAR-1		Libri2Mix				LibriCSS	
	Test		Eval-small		Test-clean		Test-both		Test	
	tcp	tcORC	tcp	tcORC	tcp	tcORC	tcp	tcORC	tcp	tcORC
Whisper	220.0	212.0	260.1	*	66.1	66.1	69.4	69.4	588.2	*
Input masking	52.8	47.9	61.6	54.0	42.2	42.1	47.9	47.9	56.2	55.0
QKb init.	276.3	*	260.0	*	323.6	*	339.4	*	990.9	*
QKb w shift	55.8	54.1	65.3	*	9.9	9.9	16.5	16.5	–	–
QKb w/o shift	47.8	46.6	28.2	*	7.9	7.9	16.4	16.4	–	–
FDDT init.	78.3	68.7	89.7	77.5	100.6	96.1	105.9	100.4	102.0	101.9
FDDT SD	17.8	17.5	20.9	20.3	6.3	6.3	13.8	13.8	–	–
+ CoAttention	**17.5**	17.2	20.8	20.3	**4.4**	**4.4**	**11.0**	**11.0**	–	–
FDDT MD	17.6	**16.7**	**19.7**	**19.1**	6.9	6.9	15.9	15.9	**8.8**	**8.8**
+ CoAttention	18.1	17.7	20.0	19.4	5.8	5.8	14.4	14.4	11.0	11.0

Table 5
Comparison of cpWER between QK biasing with and without shifted positional embeddings with ground-truth diarization.

	AMI-sdm		NOTSOFAR-1		Libri2Mix			
	Test		Eval-small		Test-clean		Test-both	
	cp	tcp	cp	tcp	cp	tcp	cp	tcp
QKb w shift	21.3	55.8	25.2	65.4	9.9	9.9	16.5	16.5
QKb w/o shift	45.9	47.8	27.3	28.5	7.9	7.9	16.4	16.4

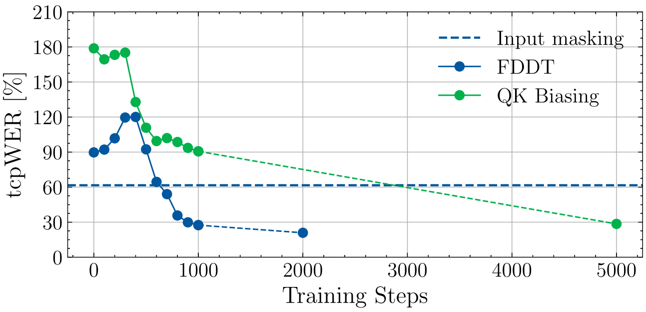


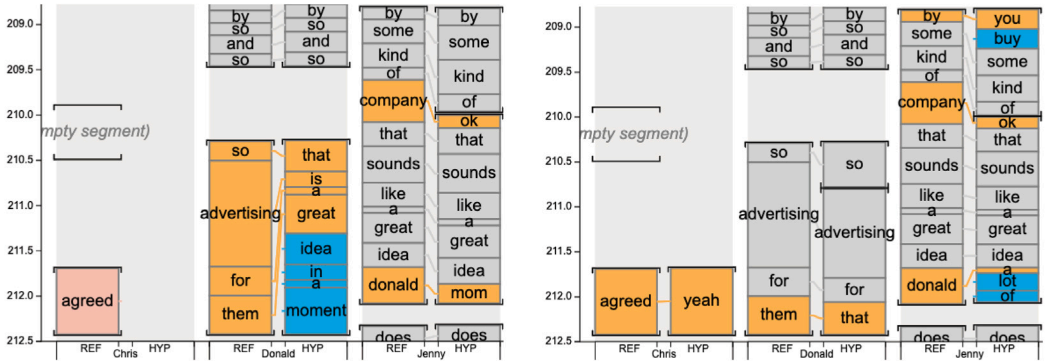
Fig. 3. Test TCP-WER as a function of training steps for the NOTSOFAR-1 model evaluated on the eval-small dataset.

6.2.1. QK biasing shift

Table 5 compares cpWER and tcpWER for QK biasing with and without shifted positional embeddings across three datasets: AMI-sdm, NOTSOFAR, and Libri2Mix. Notably, while the absolute values of the metrics change between the two settings, the relative difference between cpWER and tcpWER is effectively zero for Libri2Mix. This is primarily because Libri2Mix contains short audio segments (under 30 s), which are transcribed in a single pass. As a result, most words fall within the tcpWER collar boundaries, and incorrectly predicted timestamps do not influence the transcription of subsequent segments. Moreover, while QK biasing with shifted positional embeddings yields lower cpWER on real datasets, its performance degrades on Libri2Mix—likely because all speakers start speaking at the very beginning of the recording, creating no position "holes" during in decoder cross-attention. This observation underscores the potential pitfalls of drawing conclusions solely based on synthetic evaluation data. Consequently, our further analyses and hypotheses are based on results obtained from the real-world datasets AMI-sdm and NOTSOFAR-1.

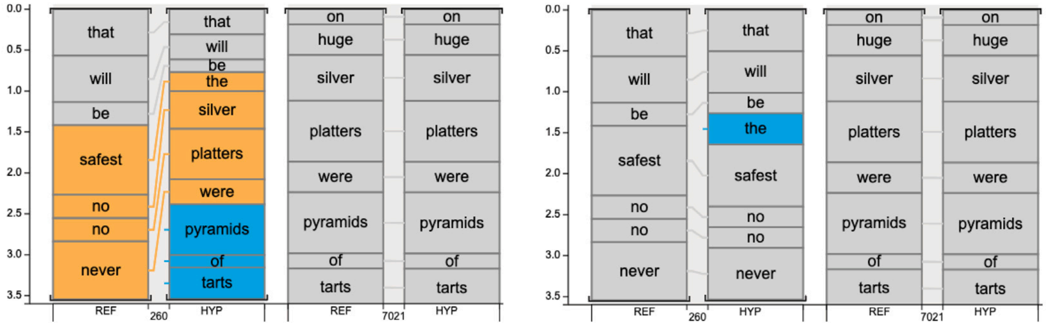
Notably, shifting positional embeddings leads to improved transcription quality on both AMI-sdm (a 24.6% absolute reduction in cpWER) and NOTSOFAR-1 (a 2.1% absolute reduction in cpWER). We hypothesize that encoder attention masking may introduce discontinuities (i.e. "holes") in the positional information within the encoder output. Since the decoder cross-attends to this representation, such discontinuities may hinder its ability to accurately track input positions, potentially resulting in hallucinations.

Conversely, the observed gap between cpWER and tcpWER indicates that the positional information becomes less reliable after shifting positional embeddings. This suggests that Whisper's timestamp prediction mechanism is highly dependent on the original positional encodings applied to the encoder input. Interestingly, when positional embeddings are not shifted, the cpWER–tcpWER gap narrows to just a few percentage points on both AMI-sdm and NOTSOFAR-1, further supporting this interpretation.



(a) Baseline decoding (b) Decoding with Co-Attention

Fig. 4. Recognition examples on the NOTSOFAR-1 dataset. In 4(a), the baseline model leaks words from speaker "Jenny" into the utterance of speaker "Donald" during an overlapping segment. In contrast, 4(b) shows that the Co-Attention mechanism nearly resolves this specific overlap correctly, but introduces errors in other overlapping regions.



(a) Baseline decoding (b) Decoding with Co-Attention

Fig. 5. Recognition examples on the Libri2Mix dataset. In 5(a), the baseline model leaks content from the other speaker and omits parts of the original utterance. In contrast, 5(b) shows that Co-Attention resolves the overlap nearly perfectly.

6.2.2. Examples of the effects of Co-Attention mechanism

We compared the decoding results of the FDDT-MD system, with and without the Co-Attention module (cf. Table 4), on examples from the NOTSOFAR-1 and Libri2Mix datasets. Figs. 4 and 5 illustrate that, while the Co-Attention mechanism effectively resolves overlapping speech, it may also introduce recognition errors, such as insertions or substitutions, in regions where the baseline model was already correct. This suggests that Co-Attention is particularly useful in resolving severe overlap but may inadvertently disturb simpler contexts.

Fig. 4 shows an example from the NOTSOFAR-1 dataset. In 4(a), the baseline model leaks words from speaker "Jenny" into speaker "Donald" during their overlapped segment. In contrast, 4(b) shows that Co-Attention nearly correctly resolves this specific overlap, although it introduces additional errors in other overlapped regions.

Fig. 5 presents a similar analysis on Libri2Mix. In 5(a), the baseline model leaks content from the interfering speaker and omits parts of the original utterance. In 5(b), the Co-Attention mechanism handles the overlap nearly perfectly, eliminating most recognition errors.

6.3. Performance with real diarization

We have demonstrated that our proposed DiCoW model achieves strong performance when ground-truth speaker and segmentation information is provided. We also evaluated our proposed system with real diarization (cf. Section 5.4). In this section, we focus on a more detailed analysis of the impact of diarization errors on DiCoW. We use the FDDT model fine-tuned on all datasets (AMI-sdm, NOTSOFAR-1, Libri2Mix).

As shown in Table 6, the performance deteriorates significantly with respect to the system with ground-truth diarization. This decline is primarily attributed to errors introduced by the diarization process (cf. Table 7). On the NOTSOFAR-1 eval-small dataset, we achieved a tcpWER of 33.5% (22.6% tcORC-WER), with 14.5% (8.3%) deletions, 10% (3.6%) insertions, and 9% (10.7%) substitutions, indicating that the system struggles with omissions in this scenario. This is expected, given that this dataset features recordings with a higher number of speakers on average and significantly more overlapping speech, both of which increase the

Table 6
Comparison of overall performance with ground-truth and a real diarization system. Both ground-truth and real diarization system results were obtained with a model trained on multiple datasets (cf. FDDT MD in Table 4).

Diarization	AMI-sdm		NOTSOFAR-1		Libri2Mix				LibriCSS	
	Test		Eval-small		Test-clean		Test-both		Test	
	tcp	tcORC	tcp	tcORC	tcp	tcORC	tcp	tcORC	tcp	tcORC
Ground-truth	17.6	16.7	19.7	19.1	6.9	6.9	15.9	15.9	8.8	8.8
Real system	25.0	18.2	33.5	22.6	8.4	8.3	20.6	20.5	11.0	8.9

Table 7
Diarization performance on the test set of each corpus.

Dataset	DER	Miss	FA	Conf.
AMI-sdm	17.3	8.7	3.8	4.8
NOTSOFAR-1	23.9	11.0	4.3	8.6
Libri2Mix (mix clean)	4.8	0.3	4.1	0.4
Libri2Mix (mix both)	10.0	1.2	8.4	0.4
LibriCSS	5.5	3.6	0.4	1.5

Table 8
Comparison of the proposed DiCoW model with various single-channel diarization systems on the NOTSOFAR-1 evaluation set. The single-channel diarization results for USTC-sys1 were extracted from USTC-NERCSLIP system submission (Niu et al., 2024; Abramovski et al., 2025).

Diarization Sys.	DER	Miss	FA	Conf.	tcpWER
DiariZen (Han et al., 2025)	35.3	11.6	7.8	15.9	50.2
Pyannote3.1 (Bredin, 2023)	33.5	19.4	1.7	12.4	47.0
USTC-sys1 (Niu et al., 2024)	21.2	8.2	7.0	6.1	28.9
Ours	23.9	11.0	4.3	8.6	33.5
Ground-truth					19.7

probability of confusion errors during diarization. In contrast, LibriCSS has fewer overlapping segments and fewer speakers per recording. However, most of the diarization errors in LibriCSS (cf. Table 7) are caused by missed speech, resulting in 4% deletions, 4% substitutions and 3% insertions. In addition, we have also compared the performance of the proposed DiCoW model using different diarization systems. The results are given in Table 8.

The errors are partly due to the model being trained with ground-truth diarization, which provides accurate speaker boundaries and correctly labeled silence. In contrast, real diarization can miss portions of a speaker's speech, particularly in scenarios with more than two overlapping speakers. The system struggles to recover from missed speech segments because it has not encountered such cases during training. This can also be verified in Table 8, where the USTC-NERCSLIP diarization system produces fewer omissions and confusions of speakers, having a direct impact on tcpWER. Note that speaker confusion errors count twice in tcpWER, as both deletions and insertions. Moreover, we initialize the FDDT parameters in such a way that the model ignores frames marked as silence from the very beginning (cf. Section 4.4). This further limits the system's ability to recover from diarization errors.

The proposed DiCoW model is designed to handle soft diarization decisions in the form of frame-by-frame speaker activity probabilities $d(s,t)$. However, in this work, we use only hard diarization decisions for both training and inference. As the result, the probabilities $d(s,t)$ and, consequently, the STNO class probabilities (4)–(7) are restricted to binary values 0 or 1.

For training, we rely solely on hard ground-truth diarization decisions. Using real diarization outputs introduces the additional challenge of aligning the speakers identified by the diarization system with the ground-truth speakers labeled in the ASR annotations. This alignment becomes especially problematic when the diarization system predicts an incorrect number of speakers.

In our system submitted to the CHiME-8 challenge (Polok et al., 2024) (closely following the method described in this paper), we demonstrated that soft diarization decisions can improve ASR decoding performance. However, subsequent analysis revealed that these improvements were largely due to the overly aggressive post-processing used to derive hard decisions in our earlier diarization system. Therefore, in this work, we always use hard diarization decisions – whether from the ground-truth or the real diarization outputs – during decoding, as this approach matches the setup used during training. Note that in such a case, Eq. (12) defining the FDDT transform reduces to a straightforward selection of one of the affine transformations corresponding to the respective STNO classes.

Incorporating soft activations into the current framework remains an open problem for future work.

6.4. Extending whisper with joint CTC/attention training and decoding

Table 9 shows the effect of the CTC head on speaker-attributed ASR performance. First, we observe a significant improvement in tcpWER simply by adding joint training with the CTC head, even without using it during decoding (see $\lambda = 0$). This highlights the importance of guiding the training by enforcing a monotonic alignment between input frames and the output token sequence, as

Table 9
Performance of the method with and without the CTC head on the NOTSFOAR-1 eval set, evaluated using tcpWER with a 5 s collar and ground-truth diarization. The table compares the effects of varying λ values in (3). When $\lambda = 1.0$, decoding is still primarily guided by the autoregressive model, and only the top 1000 tokens are rescored by CTC. The 'CTC only' column shows performance when the model uses only CTC for decoding.

	w/o CTC head	$\lambda = 0$	$\lambda = 0.2$	$\lambda = 1.0$	CTC only
Greedy	22.9	22.1	21.7	27.9	46.5
Beam 5	22.2	21.7	20.9	32.0	52.8

Table 10
WER comparison of the proposed method and original Whisper on single-speaker speech with both greedy (beam size 1) and beam-search (beam size 5) decoding. $\lambda = 0.0$ indicates no use of auxiliary CTC logits during decoding despite training with auxiliary CTC loss.

	Beam size	λ	LibriSpeech		TED-LIUM	VoxPopuli
			Test-clean	Test-other	Test	Test
Whisper	1	–	2.5	4.5	**4.3**	10.9
CTC head	1	–	3.9	7.3	8.4	16.1
Proposed	1	0.0	2.1	4.3	5.3	11.2
Proposed	1	0.2	**1.9**	4.1	8.5	11.7
Whisper	5	–	2.2	4.3	**4.3**	**10.0**
Proposed	5	0.0	2.1	4.2	5.0	11.0
Proposed	5	0.2	**1.9**	**4.0**	7.8	11.2

Table 11
Comparison of ORC-WER of fine-tuned Branchformer inference methods on AMI-sdm using ground-truth diarization.

Inference style	Baseline	With FDDT
Utterance-level	34.5	34.5
Segment-group	141.2	26.8

the Whisper decoder (or AED ASR) does not assume such alignment by default. Furthermore, using the CTC head for joint decoding further improves the performance (see $\lambda = 0.2$), although we note that using only the CTC head for decoding is significantly worse than the Whisper decoder. Overall, we surmise that the CTC head mitigates some of Whisper's hallucinatory tendencies despite being considerably worse as a standalone decoder.

6.5. Does target-speaker training affect single-speaker performance?

Next, we examine the performance of the proposed TS-ASR system on single-speaker datasets to assess its impact on Whisper's original capabilities in non-overlapping speech scenarios. This experiment was conducted using the LibriSpeech (Panayotov et al., 2015), TED-LIUM (Rousseau et al., 2012), and VoxPopuli (Wang et al., 2021) evaluation sets, with results shown in Table 10.

For this evaluation, we used the FDDT-MD model without Co-Attention. To simulate a single-speaker scenario, we construct the diarization labels by setting the entire input as belonging to the target speaker, effectively using a static STNO mask with a target speaker probability of 1 across all frames.

We observe that the model slightly outperforms the baseline on LibriSpeech — likely due to overlap with training data — but underperforms on TED-LIUM and VoxPopuli, especially when using the CTC head. This degradation is expected, as the CTC head is trained on a more limited dataset. Overall, the results suggest that our training method preserves Whisper's single-speaker ASR capability while enabling effective multi-speaker handling. This follows the hypothesis that target speaker speech may not need augmentation by FDDTs.

To further analyze this behavior, we measured the mean square change in the FDDT parameters relative to their suppressive initialization across different speaker classes. The mean square changes for *silence* and *non-target* parameters were 1.1×10^{-4}, while those for *target* and *overlap* parameters were 6×10^{-5}. This indicates that the FDDT layers make minimal adjustments to target speech, supporting the idea that target speech does not need significant transformation. As a result, the original flow of information in the model is preserved, maintaining the performance of the original model on single-speaker speech.

6.6. Non-whisper models

To verify that our method is not Whisper-specific, we used the ESPNet (Watanabe et al., 2018) LibriSpeech recipe[7] and trained attention-based encoder decoder (AED) model, namely Branchformer CTC-AED (Peng et al., 2022) on LibriSpeech 960h (Panayotov

[7] https://github.com/espnet/espnet/tree/master/egs2/librispeech/asr1.

et al., 2015). Then, we added additional FDDT parameters to the pre-trained model and6 fine-tuned it on the AMI-sdm dataset segmented the same way as for Whisper fine-tuning. To evaluate whether FDDT improves TS-ASR performance, we performed inference in two regimes commonly used in the literature, following the evaluation protocol from Kanda et al. (2021):

- **Utterance-level:** single-talker segments (usually a single sentence) were extracted according to the provided ground-truth segmentation, and the model was run separately on each segment.
- **Segment-group:** multi-talker chunks consisting of consecutive utterances not separated by silence were extracted, and the model was run separately on each group.

Table 11 presents the ORC-WER results for the two inference styles. As a baseline, we used a fine-tuned model with FDDT parameters but modified the STNO masks to always indicate the target speaker as active, effectively simulating a model not conditioned on diarization. The second column shows results for a model conditioned on diarization (in the case of utterance-style, only target and overlap information is provided, as there is no silence or other active speaker who is not overlapped with the target).

We observe that in the utterance-level inference setting, diarization information does not yield improvements over the baseline, likely because these segments are mostly single-speaker with occasional overlaps and may not provide sufficient context for accurate transcription of overlaps. Additionally, some segments contain nearly complete overlaps, where the model struggles to correctly transcribe the target speech.

In contrast, the segment-group inference setting shows a significant performance gap between the two models (an absolute improvement of 114.4%). This is primarily because the baseline model transcribes all speakers present in the segment group, resulting in many insertion errors. This result demonstrates that incorporating FDDT parameters and fine-tuning them effectively converts a single-speaker model into a TS-ASR model.

Moreover, the 7.7% absolute difference between utterance-level and segment-group performance for the FDDT model suggests that TS-ASR systems may benefit from longer contextual input.

7. Discussion

Despite its promising results, the proposed approach has several limitations:

Dependency on accurate diarization: DiCoW's performance is strongly influenced by the quality of the diarization system. Poor diarization, such as in high-overlap or noisy environments, can degrade the overall effectiveness of the model.

Performance on synthetic versus real data: While the approach shows strong results on both synthetic and real-world datasets, it faces challenges adapting from synthetic benchmarks like Libri2Mix to real-world scenarios due to domain mismatches.

Scalability with increasing speaker count: DiCoW's computational complexity and performance are affected in scenarios with a large number of speakers, as the model processes each speaker independently. However, to address this, we ensure that all computations are fully batched across speakers. This allows us to share computation and maximize GPU efficiency. In practice, we can parallelly transcribe up to 12 speakers on a single 24 GB GPU with a Whisper-large-v3-turbo backbone. We believe this level of parallelism effectively addresses concerns about the method's practicality in multi-speaker scenarios.

Handling of overlapping speech: Although the Co-Attention mechanism addresses overlapping speech to some extent, fully overlapped segments involving multiple dominant speakers remain challenging.

Limited validation on unseen conditions: While we validated DiCoW across several datasets, further testing under diverse acoustic environments, speaker characteristics, and languages is necessary to better understand its generalization capabilities.

Addressing these limitations in future work will help refine the approach and broaden its applicability to a wider range of real-world scenarios.

8. Conclusion

In this study, we presented DiCoW, an approach to extend a single-speaker ASR system to perform target/multi-speaker ASR using a diarization conditioning scheme. The main contributions are:

1. Integration of diarization for target-speaker ASR: DiCoW directly conditions Whisper's ASR capabilities on speaker diarization outputs, bypassing traditional speaker embeddings. This simplifies the workflow, reduces dependency on synthetic data, and improves generalization to unseen speakers.
2. Extension of long-form ASR to multi-speaker scenarios: By building on Whisper's long-form transcription abilities, DiCoW effectively handles overlapping speech and real-world multi-speaker recordings, enabling more reliable transcription for conversations, meetings, and other challenging audio environments.
3. Efficient fine-tuning of pre-trained ASR models: DiCoW leverages large-scale pre-trained models like Whisper, fine-tuning them with diarization conditioning to deliver strong performance across diverse datasets. This approach minimizes training costs while achieving notable accuracy gains on real-world benchmarks such as AMI and NOTSOFAR-1.

Additionally, we extended Whisper with a "CTC head" to mitigate hallucinations and compared different conditioning approaches in terms of ASR performance and speed of convergence, highlighting the advantages of Frame-Level Diarization-Dependent Transformations (FDDT). We demonstrated that adapting Whisper for multi-speaker ASR does not substantially degrade its performance on single-speaker recordings, although some loss of generalization capabilities was observed.

Moreover, while most analyses were based on Whisper, we showed that the proposed method can also be successful with other models, such as AED-based ASR, demonstrating the general effectiveness of the approach.

We have successfully demonstrated that our DiCoW model achieves strong performance across various datasets when provided with ground-truth diarization. In future work, we aim to extend the framework incorporating real speaker diarization information into training in order to reduce the performance drop that comes from training exclusively on ground-truth diarization and only using real diarization during inference.

To facilitate future comparisons and analysis, we release our code and recipes at https://github.com/BUTSpeechFIT/TS-ASR-Whisper.

CRediT authorship contribution statement

Alexander Polok: Writing – original draft, Software, Investigation. **Dominik Klement:** Writing – original draft, Software, Investigation. **Martin Kocour:** Writing – original draft, Software, Investigation. **Jiangyu Han:** Writing – original draft, Software, Investigation. **Federico Landini:** Writing – review & editing, Supervision. **Bolaji Yusuf:** Writing – review & editing, Supervision, Investigation. **Matthew Wiesner:** Writing – review & editing, Supervision. **Sanjeev Khudanpur:** Writing – review & editing, Supervision. **Jan Černocký:** Writing – review & editing, Supervision. **Lukáš Burget:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Alexander Polok reports financial support was provided by Ministry of Education Youth and Sports of the Czech Republic. Sanjeev Khudanpur reports financial support was provided by National Science Foundation. Alexander Polok reports financial support was provided by Brno - JCMM. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The work was supported by the Czech Ministry of Education (MoE) through OP JAK project No. ID:CZ.02.01.01/00/23_020/0008518, Brno Ph.D. Talent Scholarship Programme, and U.S. National Science Foundation CCRI Grant No 2120435. Computing and data repository/services were supported by MoE through e-INFRA CZ (ID:90254) and LINDAT/CLARIAH-CZ LRI (ID:90262) respectively.

Data availability

Our code and recipes can be accessed at https://github.com/BUTSpeechFIT/TS-ASR-Whisper.

References

Abramovski, I., Vinnikov, A., Shaer, S., Kanda, N., Wang, X., Ivry, A., Krupka, E., 2025. Summary of the NOTSOFAR-1 Challenge: Highlights and learnings. Computer Speech & Language (ISSN: 0885-2308) 93, 101796. http://dx.doi.org/10.1016/j.csl.2025.101796.

Bhandari, N., Chen, D., del Río Fernández, M.Á., Delworth, N., Fox, J.D., Jetté, M., McNamara, Q., Miller, C., Novotný, O., Profant, J., Qin, N., Ratajczak, M., Robichaud, J.-P., 2024. Reverb: Open-source ASR and diarization from Rev. arXiv preprint arXiv:2410.03930.

Bredin, H., 2023. Pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe. In: Interspeech 2023. pp. 1983–1987. http://dx.doi.org/10.21437/Interspeech.2023-105.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., et al., 2022. WavLM: Large-scale self-supervised pre-training for full stack speech processing. IEEE J. Sel. Top. Signal Process. 16 (6), 1505–1518.

Chen, Z., Yoshioka, T., Lu, L., Zhou, T., Meng, Z., Luo, Y., Wu, J., Xiao, X., Li, J., 2020. Continuous speech separation: Dataset and analysis. In: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 7284–7288.

Cornell, S., Jung, J.-w., Watanabe, S., Squartini, S., 2024a. One model to rule them all? Towards end-to-end joint speaker diarization and speech recognition. In: 2024 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 11856–11860.

Cornell, S., Park, T.J., Huang, H., Boeddeker, C., Chang, X., Maciejewski, M., Wiesner, M.S., Garcia, P., Watanabe, S., 2024b. The CHiME-8 DASR challenge for generalizable and array agnostic distant automatic speech recognition and diarization. In: 8th International Workshop on Speech Processing in Everyday Environments (CHiME 2024). pp. 1–6. http://dx.doi.org/10.21437/CHiME.2024-1.

Cornell, S., Wiesner, M.S., Watanabe, S., Raj, D., Chang, X., Garcia, P., Masuyam, Y., Wang, Z.-Q., Squartini, S., Khudanpur, S., 2023. The CHiME-7 DASR challenge: Distant meeting transcription with multiple devices in diverse scenarios. In: 7th International Workshop on Speech Processing in Everyday Environments (CHiME 2023). pp. 1–6. http://dx.doi.org/10.21437/CHiME.2023-1.

Cosentino, J., Pariente, M., Cornell, S., Deleforge, A., Vincent, E., 2020. LibriMix: An open-source dataset for generalizable speech separation. ArXiv: Audio and Speech Processing. URL https://api.semanticscholar.org/CorpusID:218862876.

Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P., Ouellet, P., 2010. Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. 19 (4), 788–798.

Fazel-Zarandi, M., Hsu, W.-N., 2023. Cocktail HuBERT: Generalized self-supervised pre-training for mixture and single-source speech. In: 2023 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 1–5. http://dx.doi.org/10.1109/ICASSP49357.2023.10096630.

Gandhi, S., von Platen, P., Rush, A.M., 2023. Distil-Whisper: Robust knowledge distillation via large-scale pseudo labelling. arXiv preprint arXiv:2311.00430.

Graves, A., Fernández, S., Gomez, F., Schmidhuber, J., 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd International Conference on Machine Learning. ICML '06, Association for Computing Machinery, New York, NY, USA, pp. 369–376. http://dx.doi.org/10.1145/1143844.1143891.

Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., Pang, R., 2020. Conformer: Convolution-augmented transformer for speech recognition. In: Interspeech 2020. pp. 5036–5040. http://dx.doi.org/10.21437/Interspeech.2020-3015.

Guo, P., Chang, X., Lv, H., Watanabe, S., Xie, L., 2024. SQ-Whisper: Speaker-querying based Whisper model for target-speaker ASR. arXiv preprint arXiv:2412.05589.

Han, J., Landini, F., Rohdin, J., Silnova, A., Diez, M., Burget, L., 2025. Leveraging self-supervised learning for speaker diarization. In: 2025 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 1–5. http://dx.doi.org/10.1109/ICASSP49660.2025.10889475.

Hori, T., Watanabe, S., Hershey, J., 2017. Joint CTC/attention decoding for end-to-end speech recognition. In: Barzilay, R., Kan, M.-Y. (Eds.), Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Vancouver, Canada, pp. 518–529. http://dx.doi.org/10.18653/v1/P17-1048, URL https://aclanthology.org/P17-1048.

Horiguchi, S., Takashima, Y., García, P., Watanabe, S., Kawaguchi, Y., 2022. Multi-channel end-to-end neural diarization with distributed microphones. In: 2022 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 7332–7336. http://dx.doi.org/10.1109/ICASSP43922.2022.9746749.

Huang, Z., Raj, D., García, P., Khudanpur, S., 2023. Adapting self-supervised models to multi-talker speech recognition using speaker embeddings. In: 2023 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 1–5. http://dx.doi.org/10.1109/ICASSP49357.2023.10097139.

Kanda, N., Horiguchi, S., Takashima, R., Fujita, Y., Nagamatsu, K., Watanabe, S., 2019. Auxiliary interference speaker loss for target-speaker speech recognition. In: Interspeech 2019. pp. 236–240. http://dx.doi.org/10.21437/Interspeech.2019-1126.

Kanda, N., Xiao, X., Gaur, Y., Wang, X., Meng, Z., Chen, Z., Yoshioka, T., 2022. Transcribe-to-diarize: Neural speaker diarization for unlimited number of speakers using end-to-end speaker-attributed ASR. In: 2022 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 8082–8086.

Kanda, N., Ye, G., Wu, Y., Gaur, Y., Wang, X., Meng, Z., Chen, Z., Yoshioka, T., 2021. Large-scale pre-training of end-to-end multi-talker ASR for meeting transcription with single distant microphone. In: Interspeech 2021. pp. 3430–3434. http://dx.doi.org/10.21437/Interspeech.2021-102.

Karafiát, M., Burget, L., Matějka, P., Glembek, O., Černockỳ, J., 2011. iVector-based discriminative adaptation for automatic speech recognition. In: 2011 IEEE Automatic Speech Recognition and Understanding Workshop. ASRU, IEEE, pp. 152–157.

Kinoshita, K., Delcroix, M., Tawara, N., 2021a. Advances in integration of end-to-end neural and clustering-based diarization for real conversational speech. In: Interspeech 2021. pp. 3565–3569. http://dx.doi.org/10.21437/Interspeech.2021-1004.

Kinoshita, K., Delcroix, M., Tawara, N., 2021b. Integrating end-to-end neural and clustering-based diarization: Getting the best of both worlds. In: 2021 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 7198–7202. http://dx.doi.org/10.1109/ICASSP39728.2021.9414333.

Leviathan, Y., Kalman, M., Matias, Y., 2023. Fast inference from transformers via speculative decoding. In: Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J. (Eds.), Proceedings of the 40th International Conference on Machine Learning. In: Proceedings of Machine Learning Research, 202, PMLR, pp. 19274–19286, URL https://proceedings.mlr.press/v202/leviathan23a.html.

Li, J., et al., 2022. Recent advances in end-to-end automatic speech recognition. APSIPA Trans. Signal Inf. Process. 11 (1).

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization. In: International Conference on Learning Representations. pp. 1–18, URL https://openreview.net/forum?id=Bkg6RiCqY7.

Ma, H., Peng, Z., Shao, M., Li, J., Liu, J., 2024. Extending Whisper with prompt tuning to target-speaker ASR. In: 2024 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 12516–12520.

Mccowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska Masson, A., Post, W., Reidsma, D., Wellner, P., 2005. The AMI meeting corpus. In: Int'l. Conf. on Methods and Techniques in Behavioral Research.

Meng, L., Kang, J., Wang, Y., Jin, Z., Wu, X., Liu, X., Meng, H., 2024. Empowering Whisper as a joint multi-talker and target-talker speech recognition system. In: Interspeech 2024. pp. 4653–4657. http://dx.doi.org/10.21437/Interspeech.2024-971.

Mošner, L., Serizel, R., Burget, L., Plchot, O., Vincent, E., Peng, J., Černocký, J., 2024. Multi-channel extension of pre-trained models for speaker verification. In: Interspeech 2024. pp. 2135–2139. http://dx.doi.org/10.21437/Interspeech.2024-1260.

Neumann, T.v., Boeddeker, C.B., Delcroix, M., Haeb-Umbach, R., 2023. MeetEval: A toolkit for computation of word error rates for meeting transcription systems. In: Proceedings of the 7th International Workshop on Speech Processing in Everyday Environments. CHiME 2023, pp. 27–32. http://dx.doi.org/10.21437/CHiME.2023-6.

Niu, S., Wang, R., Du, J., Yang, G., Tu, Y., Wu, S., Qian, S., Wu, H., Xu, H., Zhang, X., Zhong, G., Yu, X., Chen, J., Wang, M., Cai, D., Gao, T., Wan, G., Ma, F., Pan, J., Gao, J., 2024. The USTC-NERCSLIP systems for the CHiME-8 NOTSOFAR-1 challenge. In: 8th International Workshop on Speech Processing in Everyday Environments (CHiME 2024). pp. 31–36. http://dx.doi.org/10.21437/CHiME.2024-7.

Panayotov, V., Chen, G., Povey, D., Khudanpur, S., 2015. Librispeech: An ASR corpus based on public domain audio books. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 5206–5210. http://dx.doi.org/10.1109/ICASSP.2015.7178964.

Park, T.J., Han, K.J., Kumar, M., Narayanan, S., 2019. Auto-tuning spectral clustering for speaker diarization using normalized maximum eigengap. IEEE Signal Process. Lett. 27, 381–385.

Peng, Y., Dalmia, S., Lane, I., Watanabe, S., 2022. Branchformer: Parallel MLP-attention architectures to capture local and global context for speech recognition and understanding. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (Eds.), Proceedings of the 39th International Conference on Machine Learning. In: Proceedings of Machine Learning Research, Vol. 162, PMLR, pp. 17627–17643, URL https://proceedings.mlr.press/v162/peng22a.html.

Plaquet, A., Bredin, H., 2023. Powerset multi-class cross entropy loss for neural speaker diarization. In: Interspeech 2023. pp. 3222–3226. http://dx.doi.org/10.21437/Interspeech.2023-205.

Polok, A., Klement, D., Han, J., Sedláček, S., Yusuf, B., Maciejewski, M., Wiesner, M., Burget, L., 2024. BUT/JHU system description for CHiME-8 NOTSOFAR-1 challenge. In: 8th International Workshop on Speech Processing in Everyday Environments (CHiME 2024). pp. 18–22. http://dx.doi.org/10.21437/CHiME.2024-4.

Polok, A., Klement, D., Wiesner, M., Khudanpur, S., Černocký, J., Burget, L., 2025. Target speaker ASR with Whisper. In: 2025 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 1–5. http://dx.doi.org/10.1109/ICASSP49660.2025.10887683.

Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., Sutskever, I., 2023. Robust speech recognition via large-scale weak supervision. In: International Conference on Machine Learning. PMLR, pp. 28492–28518.

Raj, D., Denisov, P., Chen, Z., Erdogan, H., Huang, Z., He, M., Watanabe, S., Du, J., Yoshioka, T., Luo, Y., Kanda, N., Li, J., Wisdom, S., Hershey, J.R., 2021. Integration of speech separation, diarization, and recognition for multi-speaker meetings: System description, comparison, and analysis. In: 2021 IEEE Spoken Language Technology Workshop. SLT, pp. 897–904. http://dx.doi.org/10.1109/SLT48900.2021.9383556.

Raj, D., Povey, D., Khudanpur, S., 2023. SURT 2.0: Advances in transducer-based multi-talker speech recognition. IEEE/ ACM Trans. Audio Speech Lang. Process. 31, 3800–3813. http://dx.doi.org/10.1109/TASLP.2023.3318398.

Rousseau, A., Deléglise, P., Estève, Y., 2012. TED-LIUM: An automatic speech recognition dedicated corpus. In: Calzolari, N., Choukri, K., Declerck, T., Do˘ gan, M.U.g., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S. (Eds.), Proceedings of the Eighth International Conference on Language Resources and Evaluation. LREC'12, European Language Resources Association (ELRA), Istanbul, Turkey, pp. 125–129, URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/698_Paper.pdf.

Shafey, L.E., Soltau, H., Shafran, I., 2019. Joint speech recognition and speaker diarization via sequence transduction. In: Interspeech 2019. pp. 396–400. http://dx.doi.org/10.21437/Interspeech.2019-1943.

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S., 2018. X-vectors: Robust DNN embeddings for speaker recognition. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 5329–5333.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I., 2017. Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems. Vol. 30, Curran Associates, Inc., URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vinnikov, A., Ivry, A., Hurvitz, A., Abramovski, I., Koubi, S., Gurvich, I., Peer, S., Xiao, X., Elizalde, B.M., Kanda, N., Wang, X., Shaer, S., Yagev, S., Asher, Y., Sivasankaran, S., Gong, Y., Tang, M., Wang, H., Krupka, E., 2024. NOTSOFAR-1 challenge: New datasets, baseline, and tasks for distant meeting transcription. In: Interspeech 2024. pp. 5003–5007. http://dx.doi.org/10.21437/Interspeech.2024-1788.

Wang, H., Liang, C., Wang, S., Chen, Z., Zhang, B., Xiang, X., Deng, Y., Qian, Y., 2023. Wespeaker: A research and production oriented speaker embedding learning toolkit. In: 2023 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 1–5.

Wang, C., Riviere, M., Lee, A., Wu, A., Talnikar, C., Haziza, D., Williamson, M., Pino, J., Dupoux, E., 2021. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In: Zong, C., Xia, F., Li, W., Navigli, R. (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Online, pp. 993–1003. http://dx.doi.org/10.18653/v1/2021.acl-long.80, URL https://aclanthology.org/2021.acl-long.80.

Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Enrique Yalta Soplin, N., Heymann, J., Wiesner, M., Chen, N., Renduchintala, A., Ochiai, T., 2018. ESPnet: End-to-End speech processing toolkit. In: Interspeech 2018. pp. 2207–2211. http://dx.doi.org/10.21437/Interspeech.2018-1456.

Watanabe, S., Hori, T., Kim, S., Hershey, J.R., Hayashi, T., 2017. Hybrid CTC/attention architecture for end-to-end speech recognition. IEEE J. Sel. Top. Signal Process. 11 (8), 1240–1253.

Watanabe, S., Mandel, M., Barker, J., Vincent, E., Arora, A., Chang, X., Khudanpur, S., Manohar, V., Povey, D., Raj, D., Snyder, D., Subramanian, A.S., Trmal, J., Yair, B.B., Boeddeker, C., Ni, Z., Fujita, Y., Horiguchi, S., Kanda, N., Yoshioka, T., Ryant, N., 2020. CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings. In: 6th International Workshop on Speech Processing in Everyday Environments (CHiME 2020). pp. 1–7. http://dx.doi.org/10.21437/CHiME.2020-1.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M., 2020. Transformers: State-of-the-Art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, Online, pp. 38–45, URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Yoshioka, T., Abramovski, I., Aksoylar, C., Chen, Z., David, M., Dimitriadis, D., Gong, Y., Gurvich, I., Huang, X., Huang, Y., Hurvitz, A., Jiang, L., Koubi, S., Krupka, E., Leichter, I., Liu, C., Parthasarathy, P., Vinnikov, A., Wu, L., Xiao, X., Xiong, W., Wang, H., Wang, Z., Zhang, J., Zhao, Y., Zhou, T., 2019. Advances in online audio-visual meeting transcription. In: 2019 IEEE Automatic Speech Recognition and Understanding Workshop. ASRU, pp. 276–283. http://dx.doi.org/10.1109/ASRU46091.2019.9003827.

Yu, F., Zhang, S., Fu, Y., Xie, L., Zheng, S., Du, Z., Huang, W., Guo, P., Yan, Z., Ma, B., Xu, X., Bu, H., 2022. M2MeT: The ICASSP 2022 multi-channel multi-party meeting transcription challenge. In: 2022 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 6167–6171. http://dx.doi.org/10.1109/ICASSP43922.2022.9746465.

Zhang, W., Qian, Y., 2023. Weakly-supervised speech pre-training: A case study on target speech recognition. In: Interspeech 2023. pp. 3517–3521. http://dx.doi.org/10.21437/Interspeech.2023-1280.