SEPADAPTER: A SEPARATION ADAPTER FOR STREAMING AND OFFLINE MULTI-TALKER ASR BACKBONES

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-talker speech recognition must transcribe a target speaker's speech while limiting interference from overlapping voices. We propose SEPADAPTER, an activity-conditioned residual module placed after convolutional subsampling and before the automatic speech recognition (ASR) encoder. It uses a three-channel, per-frame target-relative activity signal indicating whether the target, a competing speaker, and remaining competitors are active, together with temporal context, to correct acoustic features through a residual connection. We implement the core design in FastConformer, Zipformer, and Whisper, with separately trained weights and backbone-specific context handling for streaming and offline recognition. In a parameter-matched FastConformer comparison, the 3.78M-parameter dual-path module reduces concatenated minimum-permutation word error rate (cpWER) by 0.356 percentage points (14.52% to 14.17%, 2.45% relative) over framewise conditioning. This reduction appears in all twelve seed-by-lookahead cells across three paired training runs. Diagnostics on one paired training run show gains concentrated in overlapped words and increased errors when adapter history is blocked. Insertion after encoder block 8 raises cpWER by about 7 points under the shared recipe, even though the advantage over framewise conditioning grows. The separately optimized FastConformer system achieves lower cpWER than published streaming self-speaker adaptation (SSA) results in all twelve LibriSpeechMix conditions, by 0.22–5.47 points. The mean gap grows from 0.66 points at 1120 ms ASR lookahead to 4.16 points at 80 ms, although training and diarization recipes differ. Our Whisper system, which also uses self-enrollment and encoder fine-tuning, improves over locally re-evaluated SE-DiCoW on three LibriMix conditions under oracle activity. With DiariZen activity, it reduces LibriSpeechMix three-speaker time-constrained cpWER (tcpWER) from the published 21.1% to 19.84%; results vary across domains. The controlled results support temporal activity-conditioned correction as a complement to the encoder's temporal modeling.

Keywords: multi-talker ASR, target-speaker ASR, streaming ASR, speaker activity.

1 INTRODUCTION

When two people speak at once, a recognizer must recover the target speaker's words from their shared acoustic signal. A diarizer supplies a timeline of who is active, but during overlap it marks several speakers as active without separating their words. Streaming recognition makes this harder by restricting the future speech available to interpret the current segment. A conditioning module for this setting must therefore (i) identify the target from activity alone, without an enrollment utterance; (ii) use context across frames within the encoder's lookahead; (iii) leave a pretrained recognizer unchanged when training starts; and (iv) remain small relative to the recognizer.

Recent activity-conditioned recognizers use the diarizer output only partially. Streaming self-speaker adaptation (SSA) (Wang et al., 2025) scales features by the target's activity and transforms each frame independently. It ignores when competitors speak, although the diarizer provides this, and cannot relate a frame to earlier frames in which the target spoke alone. SE-DiCoW (Polok

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

et al., 2026a) instead injects framewise activity transforms and enrollment attention throughout the Whisper encoder. Repeated conditioning alters representations that later layers were pretrained to expect; we do not test per-layer conditioning directly, but moving our adapter deeper into the encoder degrades recognition.

We propose SEPADAPTER, a single module placed after the convolutional frontend and before the ASR encoder. It receives the activity of the target, its main competitor, and all remaining speakers, and processes acoustic features across several frames. Its output is a *correction*: a learned feature vector at each frame, added to the original vector through a gate. The correction starts at zero, so the pretrained recognizer is unchanged at initialization, and every encoder layer then receives features already adjusted for the target.

Two design choices require evidence: whether the module benefits from its own temporal processing, given that the encoder already models time, and whether it should act before or inside the encoder. For the first, the temporal *dual-path* (DP) adapter is compared with a *framewise* (FW) control: a module with the same activity inputs and almost the same number of parameters that updates each frame independently, as SSA does. For the second, each module is placed before the encoder or after encoder block 8 under the same training recipe.

On FastConformer, DP lowers cpWER by 0.356 percentage points relative to FW; diagnostics on one trained pair locate larger gains in overlapping speech. Moving DP after block 8 increases cpWER by about 7 points. Across recognizers, our streaming system improves on published SSA at all lookaheads, and our Whisper system, which also uses self-enrollment, is competitive with SE-DiCoW.

2 RELATED WORK

Speaker-conditioned recognition. Token-level serialized output training (Kanda et al., 2022) lets one decoder emit all speakers in one stream, but the decoder must also resolve speaker changes. Target-speaker transducers select one speaker through an enrollment utterance (Moriya et al., 2022), which must be recorded in advance. Wang et al. (2025) introduce self-speaker adaptation (SSA), which avoids enrollment: it multiplies FastConformer features by the target's Sortformer activity and applies two feed-forward layers with a residual connection before the encoder. This compact injection supports streaming and keeps the encoder unchanged. However, it is framewise and uses target activity alone, discarding competitor activity and earlier context that the diarizer output makes available. SEPADAPTER supplies both. We compare with the SSA results published by Wang et al. (2025), which we did not author, not with the larger Nemotron-based SSA of Park et al. (2026).

Diarization-conditioned Whisper. DiCoW (Polok et al., 2026b) converts activity into silence, target, non-target, and overlap (STNO) probabilities. It compares attention query–key biasing with frame-level diarization-dependent transformations (FDDT), which blend category-specific affine transforms before each encoder layer without speaker embeddings. This yields strong offline recognition, but each FDDT is framewise, and nearly identical STNO histories can leave fully overlapping targets ambiguous. SE-DiCoW (Polok et al., 2026a) resolves such cases by selecting a target-dominant segment from the whole recording and cross-attending to its features in the encoder layers. Whole-recording enrollment suits offline use but not bounded streaming, and conditioning is repeated throughout the encoder. SEPADAPTER instead applies activity once, before the encoder; our Whisper system retains enrollment and encoder fine-tuning.

Feature separation and insertion depth. Sidecar (Meng et al., 2023) inserts a dilated temporal-convolution separator into a frozen ASR encoder to produce speaker-specific feature masks. Its depth study favors a shallow insertion, showing that a recognizer can be reused without adapting every layer, but it uses no diarization cue. Whisper-Sidecar (Meng et al., 2024) adds an enrollment-based target identifier and decoder prompts for joint multi- and target-talker recognition. Our paired FastConformer experiments examine how placement changes both absolute accuracy and the temporal-versus-framewise contrast.

Adapter components and training. We build on bottleneck adapters (Houlsby et al., 2019), feature-wise linear modulation (FiLM), which scales and shifts each feature channel according to

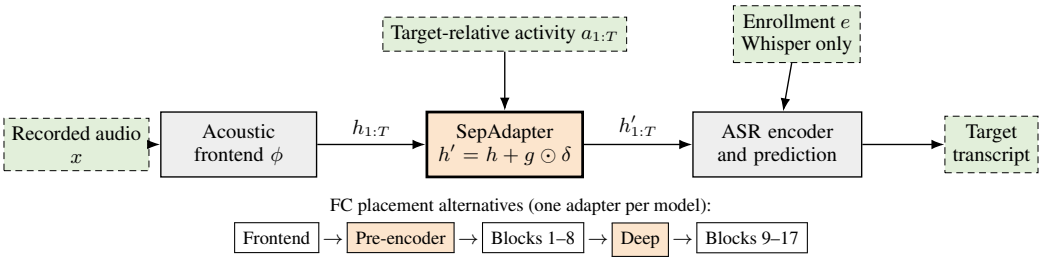

Figure 1: SEPADAPTER in the recognition pipeline. The frontend maps recorded audio x to T feature frames $h_{1:T}$. Conditioned on activity $a_{1:T}$, the adapter returns corrected features $h'_{1:T}$ of the same shape. Hosts use separately trained weights: FastConformer (D=512, cached streaming, RNNT), Zipformer (D=384, causal chunk masks, RNNT), and Whisper (D=1280, bidirectional window, decoder). The bottom strip marks the two FastConformer insertion points; Fig. 2 expands the correction branch.

a conditioning input (Perez et al., 2018), dual-path processing of within- and across-chunk context (Luo et al., 2020; Chen et al., 2020), and clean-target distillation (Moriya et al., 2023).

3 SEPADAPTER

Let x be the input mixture: a single-channel audio waveform containing all speakers' voices. The recognizer's frontend converts it into a sequence $h_{1:T} = \phi(x)$ of T learned *acoustic feature* vectors, one per 80-ms frame for FastConformer; $h_t \in \mathbb{R}^D$ describes the mixed speech of all speakers around time t, not any single speaker. FastConformer and Zipformer provide these vectors after convolutional subsampling or embedding, and Whisper after its log-mel/convolutional frontend. We call the requested speaker the *target* and other active speakers *interferers*. For each target stream, SEPADAPTER uses the acoustic features and target-relative activity $a_{1:T}$ to compute the correction $g \odot \delta_t$, a learned vector of the same width that is added to h_t so that the encoder receives features adjusted towards the target. Here, *separation* means feature modification for target recognition, not isolated-waveform reconstruction.

3.1 SELECTING A TARGET WITH ACTIVITY

For frame t, i.e., the time step of h_t, $p_s(t) \in [0, 1]$ is the activity probability or binary indicator for speaker s. The activity vector a_t tells the adapter whether the target, its main competitor, and anyone else is speaking at that frame. It has three channels: target τ, first interferer s_1, and all remaining interferers combined:

$$a_t = \left[p_\tau(t),\ p_{s_1}(t),\ \max_{s \notin \{\tau, s_1\}} p_s(t) \right] \in [0, 1]^3. \tag{1}$$

The channels are *target-relative*: the first always refers to the requested speaker, not a fixed global speaker ID. A channel is *absent* when its speaker does not exist, e.g., the third channel of a two-speaker recording; absent channels are zero; for binary activity, the last channel is a logical OR.

Controlled FastConformer training chooses s_1 by longest active duration in the full window. Inference instead chooses the first non-target stream to become active, breaks simultaneous arrivals by stream index, and retains that assignment. Before any competitor arrives, both interferer channels are zero. All remaining non-target diarizer streams enter the pooled channel without filtering. Thus only inference routing uses activity observed so far (Appendix A). Pooling fixes the input at three channels, so one trained adapter serves recordings with any number of speakers; the price is that speakers beyond s_1 are merged and cannot be told apart.

3.2 A CORRECTION THAT STARTS AT ZERO

Adding a new module must not damage a pretrained recognizer when fine-tuning starts. We therefore make the correction exactly zero at initialization. The *correction branch* is the path that projects h_t

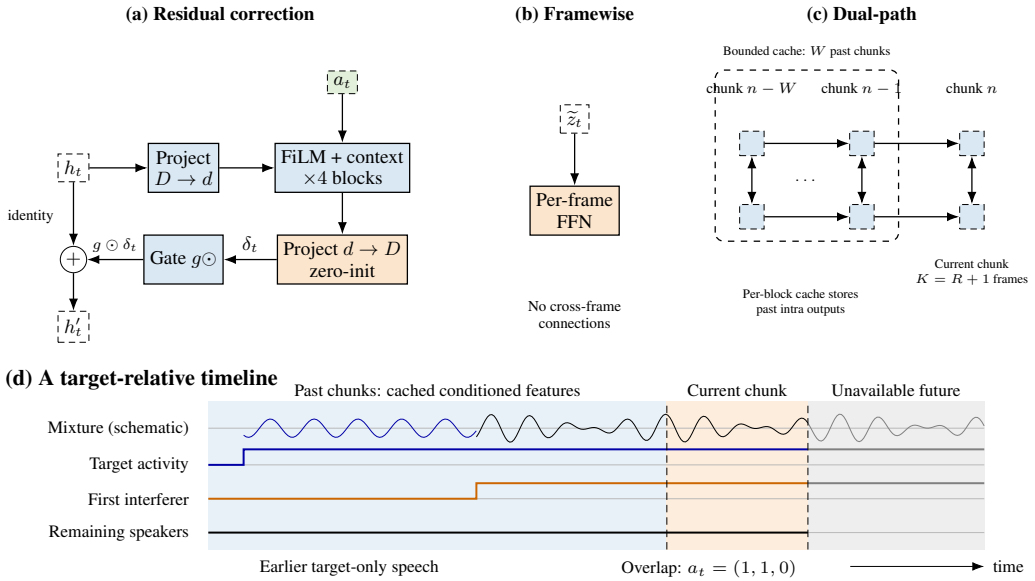

Figure 2: Residual correction and context. (a) The branch adds $g \odot \delta_t$; zero output projection preserves the input at initialization. (b) FW updates each frame independently. (c) DP: vertical arrows denote intra-chunk attention; horizontal arrows denote inter-chunk attention over the current chunk and each block's bounded cache. Two frame positions are shown schematically. (d) Illustrative timeline: current activity marks both voices as active; cached features provide earlier acoustic context.

to a bottleneck of width $d < D$ (Houlsby et al., 2019), processes the sequence with activity, and projects it back before gating and addition:

$$z_t^{(0)} = W_{\text{in}} h_t + b_{\text{in}}, \qquad z = F_\theta(z^{(0)}, a),$$
$$\delta_t = W_{\text{out}} z_t + b_{\text{out}}, \qquad h'_t = h_t + g \odot \delta_t. \tag{2}$$

Here $W_{\text{in}} \in \mathbb{R}^{d \times D}$ and $W_{\text{out}} \in \mathbb{R}^{D \times d}$ are projection weights, with biases $b_{\text{in}} \in \mathbb{R}^d$ and $b_{\text{out}} \in \mathbb{R}^D$. The stack F_θ maps the bottleneck sequence $z^{(0)}$ and activity a to z, preserving sequence length and width d. Its projected output is $\delta_t \in \mathbb{R}^D$; the actual addend is $g \odot \delta_t$, with learned channel-wise gate $g \in \mathbb{R}^D$ and elementwise multiplication $\odot$. Controlled FastConformer uses $D = 512$, $d = 192$ and four blocks. Initializing $W_{\text{out}} = b_{\text{out}} = 0$ gives $h'_t = h_t$, while $g = 0.1$ permits output-projection gradients. This identity holds at initialization; fine-tuning can change the recognizer (Appendix D).

3.3 Activity-conditioned context processing

Each of the four blocks in F_θ receives a bottleneck sequence $z_{1:T}$ and returns one of the same shape. First, feature-wise linear modulation (FiLM) (Perez et al., 2018) uses activity to produce $\widetilde{z}_t$, the bottleneck feature rescaled and shifted according to who is speaking:

$$[\gamma_t; \beta_t] = U a_t + c, \qquad \widetilde{z}_t = (1 + \gamma_t) \odot z_t + \beta_t, \tag{3}$$

The learned weight $U \in \mathbb{R}^{2d \times 3}$ and bias $c \in \mathbb{R}^{2d}$ map three activity channels to a scale adjustment $\gamma_t \in \mathbb{R}^d$ and shift $\beta_t \in \mathbb{R}^d$. Both parameters start at zero, initially leaving the block input unchanged. Next, dual-path (DP) processing (Luo et al., 2020; Chen et al., 2020) reshapes $\widetilde{z}_{1:T}$ into N chunks of K frames. Intra attention takes this $N \times K \times d$ tensor and mixes frames along K. Inter attention takes its same-shaped output and mixes matching frame positions along N, without pooling chunks. The result is restored to a frame sequence, the block output, which is the next block's input. Each path uses four heads, layer normalization and a width-768 feed-forward sublayer. Masks prevent artificial padding frames from contributing to attention.

Dual-path attention controls within-chunk and past-chunk context separately, matching the streaming interface. In Fig. 2(d), both voices are active now, while earlier frames where the target spoke

alone provide context; DP exposes this context inside the branch, whereas FW leaves it to the encoder (tested in Sec. 5.2).

The framewise control. We replace both attention paths with per-frame feed-forward layers, retaining FiLM, projections, gate, and initialization. The trainable adapter parameter counts are 3,775,168 for dual-path (DP) and 3,773,672 for framewise (FW), a difference of 1,496 or 0.04%. Perturbation tests verify that changing one frame cannot change another frame's adapter output. Both recognizers retain the same temporal encoder. This is a parameter-matched design comparison: DP also uses positional parameters and training dropout, whereas FW uses wider feed-forward layers without either (Appendix A.1).

3.4 Streaming context and insertion depth

In streaming FastConformer (Rekesh et al., 2023), the encoder consumes audio in chunks, and the adapter processes each chunk as it arrives, keeping a fixed-size memory (cache) of its own past outputs inside each DP block (Fig. 2c); Zipformer (Yao et al., 2024) uses causal chunk masking, and Whisper (Radford et al., 2023) bidirectional processing within its window. Width, positional encoding, weights, and update regime are backbone-specific (Appendix C).

For cached streaming, adapter chunks align exactly with encoder attention chunks: with right context R frames, $K = R+1$. Thus $K = 14/7/2/1$ frames of 80 ms at the four evaluated ASR lookaheads, 1120/560/160/80 ms. Intra attention stays within the current chunk. Each DP block stores its own past intra-attention outputs in a bounded cache; inter attention reads them alongside the current chunk (Fig. 2c). The cache holds $W = \lceil 70/K \rceil$ past chunks, exactly 70 frames (5.6 s) at these settings. For L blocks, stored state is $O(LWKd)$ per target, independent of recording length. This bounds each block's stored history, not the full stack's effective receptive field. Training uses causal inter attention over the available segment.

The *pre-encoder* position is between convolutional subsampling and encoder block 1. The controlled *deep* position is after block 8 of the 17-layer FastConformer. It receives states of the same width, using the same adapter initialization and streaming policy. Data, objective, schedule, and decoding are shared within each pair.

4 Experimental Setup

Two kinds of models answer different questions. *Controlled models* answer design questions: the compared adapters, such as DP and FW, are trained in pairs from the same initialization with the same data, schedule, and 100k-step endpoint, so score differences can be attributed to the design. *Optimized systems* measure performance against published systems: each recognizer has its own training recipe and checkpoint selection, so their scores are compared with SSA and SE-DiCoW but are not used to infer design effects.

Task, data, and metrics. LibriSpeechMix is a synthetic multi-talker benchmark built from LibriSpeech (Kanda et al., 2020); LSM-k denotes its k-speaker mixtures. AMI is reported for headset (IHM) and distant-microphone (SDM) recordings, while NOTSOFAR-SDM contains real meetings (McCowan et al., 2005; Abramovski et al., 2025). The Whisper tables additionally use clean and noisy Libri2Mix/Libri3Mix mixtures (Cosentino et al., 2020), a distinct benchmark. We assign emitted streams to reference speakers by the best one-to-one permutation and then report *cpWER*, the concatenated minimum-permutation word error rate. For long-form Whisper we additionally report *tcpWER*, which restricts matching with word times. Lower is better. *Oracle activity*, derived from reference speaker timestamps, shows recognition when activity is correct. *Estimated activity*, from a diarizer, measures the realistic pipeline, in which diarization errors propagate to recognition. We label these conditions separately.

Training. Controlled models start from NeMo's public streaming hybrid FastConformer large model (Rekesh et al., 2023), which has 17 encoder blocks and was pretrained on single-speaker speech. We insert the adapter and fine-tune it jointly with the recognizer on 790.8 h of 30-s multi-speaker windows: real meetings from NOTSOFAR-1 and AMI (distant microphone) and ICSI

(mixed headset channels), plus LibriMix-style mixtures that we simulate from single-speaker LibriSpeech utterances, including a harder high-overlap set (Appendix A). The loss combines the recognizer's transducer (RNNT) and connectionist temporal classification (CTC) objectives with clean-target knowledge distillation (KD): when the target's clean speech is available, encoder features computed from the mixture are pulled towards those the unadapted recognizer computes from the clean target. Each window is trained at one of the four ASR lookaheads, so one model serves all four. Within each seed, DP and FW share initialization, data, schedule, and endpoint. The separately optimized FastConformer is warm-started and selected at step 90k on development loss.

Activity and decoding. Estimated activity for FastConformer comes from the public Streaming Sortformer v2.1 checkpoint (Medennikov et al., 2025; NVIDIA, 2026), without task-specific fine-tuning. Streaming Sortformer processes audio in blocks and keeps a speaker cache; its *buffering preset*, the block length, right context, and cache sizes, sets how much audio it accumulates before emitting activity. We selected the preset using evaluation-set activity references and held it fixed across all controlled arms and lookaheads, so the controlled comparisons are conditional on this shared operating point (Appendix C). Diarization error rate (DER), the disagreement between predicted and reference activity, documents this operating point. Controlled training uses simulated/reference activity; at evaluation a frame is active when the diarizer probability exceeds 0.6. On LibriSpeechMix we decode every diarizer stream active for at least 0.64 s. The system is not told the true number of speakers, so a missed or spurious stream remains in the score as deletions or insertions. Controlled models use cached-streaming greedy decoding, in which a *cache* carries state from earlier chunks; the optimized system uses cache-aware chunked encoding. ASR lookahead excludes diarizer buffering (30.4 s on LibriSpeechMix and 1.04 s on CH109) and is not an end-to-end latency. Every DP–FW pair receives identical activity. We score with MeetEval (von Neumann et al., 2023).

Endpoint and comparisons. The primary error measure E is cpWER on LibriSpeechMix three-speaker mixtures with diarized activity, averaged over the four lookaheads at the fixed 100k step. Subscripts p and d mark pre-encoder and deep placement. Three planned contrasts answer the design questions, with negative values favoring the first term: H1, the *temporal effect*, asks whether DP beats FW before the encoder; H2, the *placement effect*, whether pre-encoder DP beats deep DP; and H3, the *interaction*, whether DP's advantage over FW differs between the two positions:

$$\begin{aligned} H1 &= E(\mathrm{DP_p}) - E(\mathrm{FW_p}), \\ H2 &= E(\mathrm{DP_p}) - E(\mathrm{DP_d}), \\ H3 &= \big[E(\mathrm{DP_p}) - E(\mathrm{FW_p})\big] - \big[E(\mathrm{DP_d}) - E(\mathrm{FW_d})\big]. \end{aligned} \tag{4}$$

Each contrast uses its complete paired seed blocks: 42/43/44 for H1 and 42/43 for H2/H3, including both placement-specific terms in H3. H1/H2 have a -0.2-point practical target; H3 was predicted to be negative. These contrasts were recorded internally before the deep-placement runs, following earlier exploration; the record and amendments appear in Appendix A.

Uncertainty. Intervals use 20,000 paired mixture-bootstrap draws that resample the same IDs across models and lookaheads, describing evaluation-sample uncertainty given trained weights. The three planned contrasts also receive Bonferroni-adjusted 98.3% intervals; KD, operator, and diagnostic comparisons are secondary, with pointwise intervals.

5 Controlled Study: Temporal Conditioning and Placement

Table 1 summarizes the planned comparisons; Appendix B gives all cells.

5.1 Temporal versus framewise conditioning

This comparison asks whether the adapter should combine information across frames (*temporal*, DP) or process each frame on its own (*framewise*, FW; Sec. 3.3). Temporal processing helps even though the encoder that follows already models time. Averaged over four lookaheads and three paired training runs, cpWER is 14.17% with DP and 14.52% with FW: DP's error is 0.356 percentage points, or 2.45% relative, lower than FW's. DP improves at all twelve seed$\times$lookahead cells, by

Table 1: Primary contrasts in cpWER percentage points. Intervals use a shared paired mixture bootstrap (20,000 draws). The 98.3% intervals adjust for three primary contrasts. Paired runs count seed–host groups, not adapter layers; each H3 group contains DP/FW at both positions. H3 is the pre-encoder DP-minus-FW contrast minus the deep contrast.

Contrast	Paired runs	Mean	95% CI	98.3% CI
Temporal: DP − FW	3	-0.356	[-0.427, -0.288]	[-0.444, -0.273]
Location: pre − deep	2	-7.223	[-7.482, -6.968]	[-7.541, -6.916]
Operator × location	2	+2.544	[+2.378, +2.710]	[+2.344, +2.747]

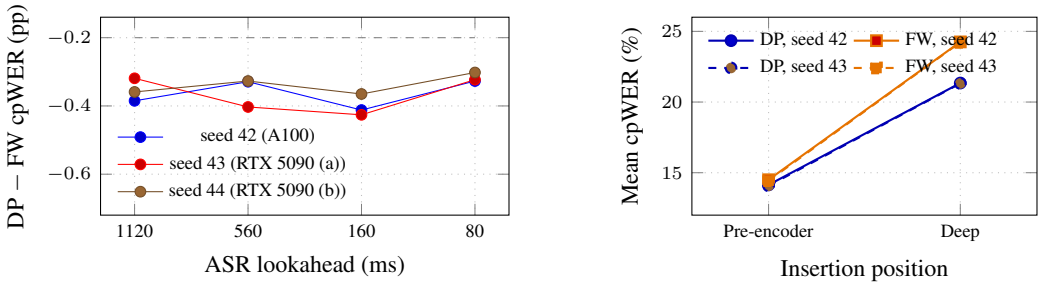

Figure 3: Left: dual-path minus framewise cpWER at the pre-encoder position, KD on; negative values favor DP, as in H1. Each paired training run has four lookaheads; the dashed line marks the planned -0.2 pp target. Right: completed two-block depth comparison, with deep insertion after block 8. Solid lines are seed 42 and dashed lines seed 43. Both adapters have higher cpWER there, with a larger increase for FW.

0.30–0.43 points (Fig. 3, left). H1's adjusted interval lies beyond the planned -0.2-point margin (Table 1).

At 80 ms, a chunk contains one frame, so intra attention cannot combine positions. The DP advantage remains 0.30–0.33 points. The benefit therefore does not require multiple frames in the current chunk.

5.2 Where do the gains occur?

If temporal context helps to follow the target through competing speech, DP's advantage should appear where speakers overlap. We separately decode the fixed seed-43 pair and align reference words to clean component signals. A word is overlapped when its aligned span intersects another speaker's word span, labeling 66,321 of 157,728 words (42.0%). Every error is retained, including unassigned insertions (Appendix D).

DP's advantage comes from overlapped speech (Table 2). On overlapped words, DP's error rate is 0.70–0.96 points below FW's at every lookahead; on non-overlapped words the two do not differ, as all intervals include zero. The overlap-minus-non-overlap intervals exclude zero, confirming that the benefit is larger in overlap.

5.3 Does the adapter use its past features?

If DP's advantage comes from temporal context, removing the adapter's memory of past chunks should hurt DP and leave FW unchanged. We prevent the trained seed-43 DP adapter from reading past-chunk cache entries, starting at the first chunk at or after 5.6 s. Current-chunk inputs and the encoder/decoder cache policy are retained.

Blocking history raises whole-mixture cpWER by 0.241 pp $[0.167, 0.319]$ at 1120 ms and 0.416 pp $[0.322, 0.511]$ at 80 ms. FW changes by exactly zero in both conditions. Sham interventions and pre-intervention prefix checks also pass. The trained DP adapter therefore relies on its stored history, although this test does not identify what information the cache carries.

Table 2: Where DP helps (seed-43 pair). Each entry is DP's error rate minus FW's on overlapped or non-overlapped reference words (pp); negative means DP makes fewer errors. Interaction is the overlap difference minus the non-overlap difference; an interval below zero means DP's benefit is larger in overlap. Pointwise paired 95% intervals; word boundaries come from clean-source forced alignment, and unassigned insertions remain in the total score (Appendix D).

ms	Overlap	Non-overlap	Interaction CI
1120	-0.801 [-1.068, -0.527]	+0.010 [-0.042, +0.060]	[-1.076, -0.540]
560	-0.960 [-1.219, -0.707]	-0.022 [-0.078, +0.035]	[-1.195, -0.686]
160	-0.948 [-1.224, -0.676]	-0.045 [-0.107, +0.017]	[-1.182, -0.632]
80	-0.701 [-0.958, -0.446]	-0.045 [-0.110, +0.019]	[-0.918, -0.398]

5.4 How does insertion depth change the temporal advantage?

Insertion after block 8 of the 17-block encoder raises DP mean cpWER by 7.204/7.243 points in runs 42/43 relative to pre-encoder insertion (H2, Table 1).

DP's advantage over FW is 0.36/0.37 points before the encoder and 2.91/2.91 after block 8 (Fig. 3, right). H3 is $+2.544$ pp, opposite to the predicted direction. A larger temporal advantage at the deep location coexists with worse absolute recognition for both variants.

A larger DP–FW advantage therefore need not indicate a better insertion point; placement must also be judged by absolute accuracy. Two positions do not separate information loss from optimization effects or locate an optimal layer.

5.5 Does the advantage remain without distillation?

The main comparison uses clean-target knowledge distillation (KD). Repeating both pre-encoder architectures without KD gives DP reductions of 0.358 and 0.161 points in runs 43/44, respectively, or 0.259 pp $[0.189, 0.331]$ on average. Removing KD reduces the DP advantage by 0.094 pp $[0.042, 0.145]$. Thus, the temporal benefit remains, but its magnitude depends on KD. Full contrasts appear in Appendix B.

Temporal operator and cost. A parameter-matched temporal convolutional network (TCN) adapter, trained once with seed 42, averages 14.04% cpWER versus DP's 14.14% with a different effective history (Appendix B), so other temporal operators are also viable. DP's advantage over FW costs 4–10% more ASR wall time and 210 KiB of adapter cache per target, whereas FW is stateless (Table 8, Appendix D).

6 System-Level Results

This section compares our separately optimized systems with published systems. They differ from the baselines in data and recipe as well as architecture, so their gains are not attributable to temporal processing alone. To check that the recognizers actually use the adapter, we re-evaluate each trained checkpoint with the correction switched off ($h'_t = h_t$), keeping all other weights. On single-speaker LibriSpeechMix (LSM-1), FastConformer cpWER changes by only 0.13–0.15 points, but on two- and three-speaker mixtures it rises to 86.65–87.19% and 146.54–148.73%; Zipformer behaves alike (Table 6). The recognizers therefore depend on the adapter to handle mixtures.

6.1 Streaming FastConformer against published SSA

We compare our streaming FastConformer system with SSA (Wang et al., 2025) on LibriSpeechMix with one to three speakers at four ASR lookaheads, and find lower cpWER in all twelve conditions, by 0.22–5.47 points (Table 3). The lookahead is how much future audio the recognizer may wait for before emitting words; a shorter lookahead gives lower latency but harder recognition. The mean gap grows from 0.66 points at the longest lookahead (1120 ms) to 4.16 points at the shortest (80 ms), so the advantage is largest when the least future context is available. SSA's offline scores (2.2/2.8/5.0) remain lower than every streaming row. The advantage also carries over from simulated mixtures

Table 3: LibriSpeechMix cpWER (%, lower is better) at four ASR lookaheads. SSA is the published max-3-speaker streaming system (Wang et al., 2025); training and diarization recipes differ.

ASR lookahead	1 speaker		2 speakers		3 speakers	
(ms)	SSA	Ours	SSA	Ours	SSA	Ours
1120	4.00	**2.50**	5.40	**5.18**	9.80	**9.53**
560	4.30	**2.60**	5.90	**5.36**	11.00	**9.75**
160	6.20	**2.75**	7.80	**5.72**	15.70	**10.23**
80	7.10	**2.81**	9.20	**6.01**	15.60	**10.61**

to real conversations: on CH109 (109 two-speaker CALLHOME English telephone conversations), our 1120-ms, 30-s windowed greedy system scores 23.12 versus SSA's published 26.21, although SSA was additionally fine-tuned for 50k steps on Fisher telephone speech. SSA also fine-tunes its diarizer; we keep public Sortformer at similar nominal DER (3.43/4.19/5.92% versus 3.3/4.1/5.6% on 2-mix/3-mix/CH109), under different scoring settings (Appendix C).

6.2 Offline Whisper against SE-DiCoW

SepAdapter-W combines a 4.061M residual core and 209.8M enrollment blocks with Whisper large-v3-turbo (Radford et al., 2023), fine-tuning the encoder with a frozen decoder. Unlike SE-DiCoW, it applies no framewise activity transforms inside the encoder: activity enters once, through the adapter, while enrollment cross-attention is kept in eight encoder layers. The model has 1105.5M parameters versus SE-DiCoW's 1101.8M (+0.34%), so gains are not explained by added capacity. The 36k checkpoint, selected on AMI dev with greedy decoding, is fixed for oracle and DiariZen evaluation; Table 7 adds paired oracle comparisons to a locally re-evaluated baseline.

With DiariZen, six of our scores are lower than published SE-DiCoW, two are equal once ours are rounded to the one decimal place SE-DiCoW reports (LSM-1, LSM-2), and two are higher (Table 4). Reductions reach 1.26 points on LSM-3 ($21.1 \rightarrow 19.84$), 0.53 on clean Libri2Mix, and 0.32 on AMI-IHM. These comparisons are descriptive, not significance tests.

Table 4: Offline tcpWER (%, 5-s collar). SE: published SE-DiCoW (Polok et al., 2026a); Ours: fixed 36k checkpoint. Bold: the lower score after rounding ours to one decimal place, the precision SE-DiCoW reports. NSF: NOTSOFAR-SDM; SDM/IHM: AMI. LSM: LibriSpeechMix; L2/L3: Libri2Mix/3Mix; B/C: both (noisy)/clean.

	NSF	SDM	IHM	LSM-1	LSM-2	LSM-3	L2-B	L2-C	L3-B	L3-C
Oracle activity										
SE	15.8	14.3	**11.0**	1.7	**2.1**	**2.9**	7.7	2.8	19.9	9.7
Ours	**15.61**	**14.13**	11.04	1.70	2.23	3.75	**7.09**	**2.70**	**18.34**	**8.06**
DiariZen activity										
SE	**26.1**	18.5	15.3	1.8	3.0	21.1	8.7	3.4	**35.6**	29.3
Ours	26.24	**18.46**	**14.98**	1.80	3.00	**19.84**	**8.44**	**2.87**	36.84	**28.87**

Conclusion. SepAdapter applies target-relative activity, including competitors, with temporal context in one correction before the encoder. Paired comparisons, overlap localization, and cache intervention show that this beats framewise conditioning in the tested recipe, while deeper insertion degrades recognition. It improves on published streaming SSA and is competitive with SE-DiCoW, with gains that depend on domain and activity source.

AI USE STATEMENT

We used ChatGPT/Codex and Claude to assist with research planning and hypothesis refinement, experiment design, implementation and debugging, data-processing and evaluation scripts, interpretation of results, literature search, manuscript drafting and editing, and translation. Numerical results reported here come from model runs and scoring scripts, not from language-model estimates. Code checks include negative controls, checkpoint/protocol receipts, and paired-output validation, as described in the paper and artifact. The authors remain responsible for checking AI-assisted code, citations, mathematical statements, interpretations, and the final manuscript.

ETHICS STATEMENT

This work uses existing speech corpora rather than newly recruited participants. Data access and redistribution remain subject to the original corpus licenses and restrictions. Speaker-attributed transcription can expose sensitive conversations and enable surveillance; improved recognition does not remove those risks. Deployment requires appropriate consent and privacy controls. Diarization assigns anonymous speaker streams within a recording; it does not by itself establish a person's identity.

REPRODUCIBILITY STATEMENT

Section 4 specifies the endpoint and paired statistics; the appendices give training recipes, completed results, and diagnostic definitions. The anonymous supplementary archive is `SepAdapter_ICLR2027_ANONYMOUS_SUPPLEMENT_20260926_FINAL.zip`. Its `sepadapter_supplement/README.md` gives the commands and file map; `MANIFEST.json` and `SHA256SUMS` bind the distributed files. This archive is the review access point and requires no author account or private repository.

Two CPU tasks are reproducible from the supplied files. First, `repro_h1/reproduce.py` recomputes H1 from all 24 cells' per-mixture error and reference-word counts, using the paired bootstrap and three-contrast interval adjustment; `repro_h1/MANIFEST.json` records checkpoint, code, receipt, and count-file hashes. Second, `scorer_audit/rescore.py` independently scores the supplied one-mixture transcripts with MeetEval, recovering 21 errors in 51 reference words. The separate local audio-to-text smoke test produced this sample; the portable scorer audit starts from its transcripts. Hashes establish file identity, not inference correctness.

The `code/` directory supplies the adapter implementation, host train/evaluation source snapshots, and CPU checks of parameter counts, routing, and nonzero synthetic output/cache parity against the retained DP implementation. `code/ANONYMIZATION.json` records source transformations and original/exported hashes. These checks do not rerun trained ASR.

`provenance/CHECKPOINT_PROVENANCE.json` separates the controlled 100k endpoints, optimized FastConformer 90k export, and Whisper 36k checkpoint selected on AMI dev. The FastConformer export matches its retained resume weights, but its exact executed trainer revision remains unresolved. The AMI-dev selection sweep is retained on the training host and is not included. Full training and full-dataset audio-to-text reproduction require the backbone environments, model weights, activity inputs, data-generation manifests, and audio under the original access terms; they are outside this package. Strict adapter reconstruction and the local inference check provide additional recorded evidence, not a complete reproduction from scratch.

REFERENCES

Igor Abramovski, Alon Vinnikov, Shalev Shaer, Naoyuki Kanda, Xiaofei Wang, Amir Ivry, and Eyal Krupka. Summary of the notsofar-1 challenge: Highlights and learnings, 2025.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html`.

Jingjing Chen, Qirong Mao, and Dong Liu. Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation. In *Proc. Interspeech*, pp. 2642–2646, 2020.

Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent. Librimix: An open-source dataset for generalizable speech separation, 2020.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proc. ICML*, pp. 2790–2799, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *Proc. ICLR*, 2022.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. The ICSI meeting corpus. In *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2003. doi: 10.1109/ICASSP.2003.1198793. URL https://www.ee.columbia.edu/~dpwe/pubs/icassp03-janin.pdf.

Naoyuki Kanda, Yashesh Gaur, Xiaofei Wang, Zhong Meng, and Takuya Yoshioka. Serialized output training for end-to-end overlapped speech recognition. In *Proc. Interspeech*, pp. 2797–2801, 2020.

Naoyuki Kanda, Jian Wu, Yu Wu, Xiong Xiao, Zhong Meng, Xiaofei Wang, Yashesh Gaur, Zhuo Chen, Jinyu Li, and Takuya Yoshioka. Streaming multi-talker asr with token-level serialized output training. In *Proc. Interspeech*, pp. 3774–3778, 2022. doi: 10.21437/Interspeech.2022-7.

Yi Luo, Zhuo Chen, and Takuya Yoshioka. Dual-path rnn: Efficient long sequence modeling for time-domain single-channel speech separation. In *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 46–50, 2020. doi: 10.1109/ICASSP40776.2020.9054266.

Iain McCowan, Jean Carletta, Wessel Kraaij, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska, Wilfried Post, Dennis Reidsma, and Pierre Wellner. The ami meeting corpus. In *Proc. 5th Int. Conf. Methods and Techniques in Behavioral Research*, pp. 137–140, 2005.

Ivan Medennikov, Taejin Park, Weiqing Wang, He Huang, Kunal Dhawan, Jinhan Wang, Jagadeesh Balam, and Boris Ginsburg. Streaming sortformer: Speaker cache-based online speaker diarization with arrival-time ordering, 2025.

Lingwei Meng, Jiawen Kang, Mingyu Cui, Yuejiao Wang, Xixin Wu, and Helen Meng. A sidecar separator can convert a single-talker speech recognition system to a multi-talker one. In *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10095295.

Lingwei Meng, Jiawen Kang, Yuejiao Wang, Zengrui Jin, Xixin Wu, Xunying Liu, and Helen Meng. Empowering whisper as a joint multi-talker and target-talker speech recognition system. In *Proc. Interspeech*, pp. 4653–4657, 2024. doi: 10.21437/Interspeech.2024-971.

Takafumi Moriya, Hiroshi Sato, Tsubasa Ochiai, Marc Delcroix, and Takahiro Shinozaki. Streaming target-speaker asr with neural transducer. In *Proc. Interspeech*, pp. 2673–2677, 2022. doi: 10.21437/Interspeech.2022-11425.

Takafumi Moriya, Hiroshi Sato, Tsubasa Ochiai, Marc Delcroix, Takanori Ashihara, Kohei Matsuura, Tomohiro Tanaka, Ryo Masumura, Atsunori Ogawa, and Taichi Asami. Knowledge distillation for neural transducer-based target-speaker asr: Exploiting parallel mixture/single-talker speech data. In *Proc. Interspeech*, pp. 899–903, 2023. doi: 10.21437/Interspeech.2023-2280.

NVIDIA. Streaming Sortformer 4-speaker v2.1: Model card. Hugging Face, 2026. URL https://huggingface.co/nvidia/diar_streaming_sortformer_4spk-v2.1. Accessed September 24, 2026.

Taejin Park, Ivan Medennikov, Kunal Dhawan, Weiqing Wang, Jagadeesh Balam, and Boris Ginsburg. Pushing the boundaries of streaming multi-speaker ASR: A systematic study of architectural trade-offs, 2026. URL https://arxiv.org/abs/2609.10265.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proc. AAAI*, pp. 3942–3951, 2018.

Alexander Polok, Dominik Klement, Samuele Cornell, Matthew Wiesner, Jan Cernocky, Sanjeev Khudanpur, and Lukas Burget. Se-dicow: Self-enrolled diarization-conditioned whisper, 2026a.

Alexander Polok, Dominik Klement, Martin Kocour, Jiangyu Han, Federico Landini, Bolaji Yusuf, Matthew Wiesner, Sanjeev Khudanpur, Jan Cernocky, and Lukas Burget. Dicow: Diarization-conditioned whisper for target speaker automatic speech recognition. *Computer Speech & Language*, pp. 101841, 2026b. doi: 10.1016/j.csl.2025.101841.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proc. 40th Int. Conf. Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, 2023.

Dima Rekesh, Nithin Rao Koluguri, Samuel Kriman, Somshubra Majumdar, Vahid Noroozi, He Huang, Oleksii Hrinchuk, Krishna Puvvada, Ankur Kumar, Jagadeesh Balam, and Boris Ginsburg. Fast conformer with linearly scalable attention for efficient speech recognition, 2023.

Thilo von Neumann, Christoph Boeddeker, Marc Delcroix, and Reinhold Haeb-Umbach. MeetEval: A toolkit for computation of word error rates for meeting transcription systems. In *Proc. CHiME Workshop*, 2023.

Weiqing Wang, Taejin Park, Ivan Medennikov, Jinhan Wang, Kunal Dhawan, He Huang, Nithin Rao Koluguri, Jagadeesh Balam, and Boris Ginsburg. Speaker targeting via self-speaker adaptation for multi-talker asr. In *Proc. Interspeech*, pp. 5498–5502, 2025. doi: 10.21437/Interspeech.2025-2142.

Zengwei Yao, Liyong Guo, Xiaoyu Yang, Wei Kang, Fangjun Kuang, Yifan Yang, Zengrui Jin, Long Lin, and Daniel Povey. Zipformer: A faster and better encoder for automatic speech recognition. In *Proc. Int. Conf. Learning Representations (ICLR)*, 2024.

A Training and evaluation details

Controlled recipe. The cohort uses 790.8 h of 30-s windows from NOTSOFAR-1 SDM, AMI SDM, ICSI headset mixtures (Janin et al., 2003), and generated/hard simulated mixtures, with sampling weights 4:4:4:6:3. Training minimizes $0.7\mathcal{L}_{\mathrm{RNNT}} + 0.3\mathcal{L}_{\mathrm{CTC}}$ plus clean-target KD where clean references exist. The teacher is the frozen adapter-free host; L1 and cosine distances between teacher and student encoder features have coefficient 0.5. Overlapped frames receive weight 5, other frames weight 1. Batch size is 32, cosine learning rate starts at 10^{-5}, and adapter/CTC-head multipliers are 5/100. SpecAugment probability is 0.3. The four ASR contexts are sampled with probabilities 0.55/0.15/0.15/0.15, and all controlled arms end at 100k steps. Seeds 42/43/44 use an A100 and two RTX 5090 hosts; only the latter enable activation checkpointing. Host policy is shared within a seed, so across-seed variability includes hardware effects.

Protocol and amendments. The internal experiment record preceded the deep-mount runs but followed earlier exploration; it is not an external preregistration. H1–H3, their directions, and the fixed endpoint are defined in Sec. 4. The primary execution path was amended to cached-streaming greedy decoding on September 16. After the seed-43 deep results, the third deep pair was replaced by KD-off arms; H3 therefore uses two complete pairs, not three. We report Bonferroni-adjusted intervals for the three primary contrasts and pointwise intervals for secondary analyses. The technical supplement retains amendments, per-cell checkpoint identifiers, and diagnostic protocols for these studies.

Activity routing. Training assigns the first interferer slot to the competitor with longest active duration in the window; inference assigns it by first activity and retains the assignment. Both architectures share this mismatch. A 1,500-trial prefix check verifies causal inference routing on fixed diarizer outputs, not the diarizer itself. Pooling merges identities beyond the first competitor. The 0.64-s emitted-stream threshold does not impose oracle speaker count: count mismatches in 186/2,620 mixtures remain in every controlled cell.

A.1 EXACT ADAPTER CONFIGURATIONS

All variants use biased $512 \to 192 \to 512$ projections, four width-192 blocks, zero-initialized output weights/bias, and a learned 512-channel gate initialized to 0.1. Each block begins with zero-initialized FiLM. Layer normalization is affine with $\epsilon = 10^{-5}$; activations are GELU. No speaker-query memory is enabled.

DP has 3,775,168 parameters. Each block adds a zero-initialized 14×192 within-chunk position table and applies pre-normalized intra/inter Transformer layers, each with four heads, a $192 \to 768 \to 192$ feed-forward sublayer, and dropout 0.1 during training. A final layer normalization follows inter processing. There is no inter-position embedding. Unfilled cache and padded positions are masked; each block retains $W \times K \times 192$ past intra outputs with $W = 70/K$.

FW has 3,773,672 parameters. Each block uses two pre-normalized residual feed-forward sublayers of widths $192 \to 1157 \to 192$ followed by layer normalization, without temporal mixing, position embeddings, dropout, or cache. TCN has 3,764,792 parameters: chunk-local depthwise kernel-3 convolution plus chunk mean, causal depthwise kernel-6 convolution with dilations $(1, 2, 4, 7)$ across blocks, and a $192 \to 2014 \to 192$ feed-forward sublayer. Padding is masked before convolution. Its causal path spans 70 past frames across the stack, whereas DP retains 70 per block; this is not a matched-effective-history comparison.

B COMPLETED CONTROLLED RESULTS

Table 5: Controlled cohort on LibriSpeechMix 3-mix: cpWER (%), diarized activity, cached-streaming greedy decoding, fixed 100k steps. Each cell scores the same 2,620 mixtures. Mean averages four ASR lookaheads; no checkpoint selection is used.

Adapter	Mount	Seed	KD	1120	560	160	80	Mean
Dual-path	Pre-encoder	42	on	13.20	13.81	14.56	14.98	14.14
Dual-path	Pre-encoder	43	on	13.13	13.70	14.54	14.91	14.07
Dual-path	Pre-encoder	43	off	13.19	13.77	14.66	15.03	14.16
Dual-path	Pre-encoder	44	on	13.30	13.98	14.74	15.11	14.28
Dual-path	Pre-encoder	44	off	13.48	14.04	14.81	15.22	14.39
Dual-path	Deep	42	on	20.22	20.74	21.98	22.44	21.34
Dual-path	Deep	43	on	20.26	20.76	21.84	22.40	21.32
Framewise	Pre-encoder	42	on	13.59	14.14	14.98	15.31	14.50
Framewise	Pre-encoder	43	on	13.45	14.10	14.97	15.24	14.44
Framewise	Pre-encoder	43	off	13.58	14.21	15.04	15.25	14.52
Framewise	Pre-encoder	44	on	13.66	14.31	15.11	15.41	14.62
Framewise	Pre-encoder	44	off	13.64	14.19	15.04	15.32	14.55
Framewise	Deep	42	on	23.38	23.87	24.67	25.09	24.25
Framewise	Deep	43	on	23.35	23.86	24.64	25.06	24.23
TCN	Pre-encoder	42	on	13.07	13.63	14.49	14.96	14.04

Table 5 retains every complete four-lookahead endpoint, including TCN and KD-off. All cells use 2,620 mixtures and 157,728 reference words. The seed-42 TCN-minus-DP difference is -0.102 pp $[-0.179, -0.024]$, conditional on these checkpoints. The per-cell ledger is in the technical supplement; the main text reports all primary contrasts and their uncertainty.

Distillation contrasts. Averaged over seeds 43/44, KD-off minus KD-on is $+0.097$ pp $[+0.058, +0.135]$ for DP and $+0.003$ $[-0.035, +0.041]$ for FW. DP minus FW without KD is -0.259 $[-0.331, -0.189]$; the architecture-by-KD interaction is $+0.094$ $[+0.042, +0.145]$. These mixture-bootstrap intervals do not estimate training-seed uncertainty.

C HOST RECIPES AND SYSTEM EVIDENCE

FastConformer and Zipformer. The optimized FastConformer is a separate warm-started system selected at step 90k by mean RNNT development loss over AMI dev and LibriSpeechMix dev 2/3-mix at all four lookaheads. The no-FDDT export matches all 777 retained tensors of the resume

checkpoint, with 34 frame-transformation tensors removed and that branch disabled. Both files have SHA-256 identifiers in the anonymous artifact; the exact executed trainer revision remains unresolved. Zipformer is pretrained on LibriSpeech and GigaSpeech and remains frozen except for attention LoRA (Hu et al., 2022) ($r = 8$, approximately 0.165M parameters). Its 3.72M adapter first trains for 12 epochs on 300 h of dense mixtures with clean-target KD (LR 3×10^{-4}), then jointly with LoRA for four epochs on 300 h of one- to three-speaker mixtures plus AMI headset-mixed training audio under RNNT loss (LR 2×10^{-4}). Evaluation uses 320-ms chunk masks, activity threshold 0.5 and a 0.32-s emitted-stream threshold; bounded-cache deployment is not established for this host.

Table 6: Zipformer cpWER (%) at 320-ms chunk masking with the adapter+LoRA checkpoint. "Estimated" uses Sortformer activity; "Oracle" uses reference activity; OFF bypasses the adapter. AMI-IHM is the 16-meeting test set.

Activity	1-mix	2-mix	3-mix	CH109	AMI-IHM
Estimated	3.71	6.53	10.63	25.99	–
Oracle ON	3.06	5.80	9.18	25.46	26.09
Oracle OFF	3.02	85.22	149.10	–	–

Whisper checkpoint 36k. The new system retains the residual/enrollment architecture but uses a new training run. An earlier trained adapter, enrollment blocks, and CTC head are combined with the public CTC-pretrained encoder. The retained S1 checkpoint follows 8k updates with encoder and decoder frozen (LR 2×10^{-6}, warm-up 500, cosine horizon 12k, adapter/head multipliers 0.25/100). S2 fine-tunes the encoder with a frozen decoder, batch 24×4, peak LR 4×10^{-6}, 10k warm-up, and a 40k cosine horizon. We evaluate its 36k update, including activity-corruption training from step 30k, without weight averaging. Corruption comprises competing-speaker activity swaps, segment changes, and Gaussian perturbations. The source weights are 6:6:1:1:1:1 for NOTSOFAR train plus all dev1, AMI-SDM train, two noisy Libri2Mix training sets, noisy Libri3Mix, and regenerated three-distinct-speaker LibriSpeech mixtures. LibriSpeech training enrollment uses only training partitions. The objective is cross-entropy plus 0.3 CTC loss; MUSAN probability is 0.3.

Whisper checkpoint selection. We selected the 36k checkpoint by the lowest tcpWER under greedy decoding on AMI-SDM dev, which is held out from training. Test-set scores were not used to choose this checkpoint. The selected weights are fixed across all ten datasets and both activity conditions in Table 4. Development-set selection uses greedy decoding; the reported test matrix uses the beam-search configuration below. The development sweep is retained on the training host; the anonymous artifact records this selection protocol and the evaluated checkpoint hash, but does not include the sweep log or certify its numerical ranking.

Whisper evaluation. Both activity conditions use beam 5, CTC weight 0.2, length penalty 0.1, enrollment seed 42, and the NOTSOFAR/CHiME-8-compatible text normalizer (`whisper_nsf`). MeetEval tcpWER uses a 5-s collar. Each condition contains 160 NOTSOFAR device recordings (80 meetings), 16 recordings per AMI setting, 2,620 mixtures per LSM setting, or 3,000 per LibriMix setting. Table 4 uses the published SE-DiCoW row, preserving its one-decimal precision. Table 7 instead pairs our oracle outputs with public SE-DiCoW revision `470fce9` evaluated in our harness, with identical reference text and timing verified per recording. Its intervals describe evaluation-sample uncertainty conditional on the fixed evaluated models; they do not estimate variability across training runs. DiariZen scores have no paired baseline outputs in this comparison. The published restricted-NOTSOFAR footnote marks the original DiCoW baseline, not the SE-DiCoW row used here.

Earlier Whisper diagnostic. The technical supplement retains the earlier three-checkpoint average and its oracle overlap analysis. That system used NOTSOFAR dev1 for selection rather than training. Its regional results locate larger Libri3Mix gains in estimated overlap; they are not measurements on the new 36k checkpoint.

Table 7: Checkpoint 36k versus locally re-evaluated SE-DiCoW, oracle activity. Paired 95% intervals use 20,000 ratio-of-sums bootstrap draws; NOTSOFAR resamples 80 physical meetings with both devices, other rows resample recordings/mixtures. Intervals condition on the selected checkpoints. Bold differences exclude zero before rounding.

Dataset	Ours	SE (local)	Δ	95% CI
NOTSOFAR-SDM	15.61	15.60	$+0.01$	$[-0.15, +0.18]$
AMI-SDM	14.13	14.23	-0.10	$[-0.28, +0.12]$
AMI-IHM	11.04	10.97	$+0.07$	$[-0.12, +0.30]$
LSM-1	1.70	1.72	-0.02	$[-0.10, +0.05]$
LSM-2	2.23	2.07	$\mathbf{+0.16}$	$[+0.06, +0.27]$
LSM-3	3.75	2.95	$\mathbf{+0.80}$	$[+0.55, +1.06]$
L2M-B	7.09	7.66	$\mathbf{-0.58}$	$[-0.79, -0.36]$
L2M-C	2.70	2.81	-0.11	$[-0.25, +0.03]$
L3M-B	18.34	19.89	$\mathbf{-1.54}$	$[-1.83, -1.25]$
L3M-C	8.06	9.73	$\mathbf{-1.68}$	$[-1.92, -1.44]$

Diarization operating point. We selected buffering presets of the public, non-task-fine-tuned Sortformer by nominal DER: the target was a value near but above SSA's published DER, rather than minimizing recognition error separately for each ASR model. This selection used evaluation-set activity references; the resulting caches are fixed across all controlled arms and lookaheads. Our fixed-cache rescore gives 3.43/4.19/5.92% on 2-mix/3-mix/CH109 at threshold 0.5 and NeMo collar 0.25, including overlap; the ASR masks use threshold 0.6. NeMo's collar corresponds to twice that width in Pyannote. SSA uses zero collar and removes boundary silence for LibriSpeechMix. Under those options, our VAD-based references give 8.42/9.33% at threshold 0.5 and 9.13/10.21% at 0.6. Thus nominal DER proximity motivated the preset choice but does not establish matched input quality. The retained LibriSpeechMix/CH109 configurations buffer 30.4/1.04 s respectively; ASR lookaheads exclude this buffering. The supplement gives the complete rescore. Our adaptation corpora are public under their access terms; we do not assume that SSA's different recipes use private recordings.

D DIAGNOSTIC DEFINITIONS

Overlap and history. For FC, a wav2vec 2.0 CTC aligner (Baevski et al., 2020) estimates word spans from each clean source and known transcript; mixture offsets put them on one timeline. A word is overlapped when its span intersects another speaker's word span. Midpoint assignment and 50/100-ms expansions test boundary sensitivity. Whole-mixture speaker assignment precedes regional scoring; substitutions/deletions inherit the reference region, and insertions use the adjacent aligned reference or an unassigned category. Regional errors sum to the primary error count. This replaces exploratory pseudo-timing, which marked 53.6% rather than 42.0% of words as overlapped. History intervention masks only past adapter-cache entries from the first chunk at or after 5.6 s, preserving the current chunk and host-cache policy. Sham, unchanged-prefix and stateless-FW checks pass; subsequent host states may diverge. The test measures cache reliance, not recovery of speaker identity.

Residual properties. For $G = \mathrm{diag}(g)$, corrections lie in $Gb_{\mathrm{out}} + \mathrm{range}(GW_{\mathrm{out}})$; mean-centered corrections have rank at most d. Zero output projection gives exact identity at initialization, not guaranteed retention after training. The nonzero gate permits output-projection gradients; deeper branch gradients begin after that projection changes. Identical acoustic and activity histories produce identical deterministic corrections, so temporal processing alone cannot distinguish targets with fully identical cues.

Additional cost measurement. The encoder-only probe uses one idle RTX 5090, FP32, batch 1, 30-s input and CUDA synchronization. DP adds 10.0–10.6% encoder time, FW 5.9–6.0%, and TCN 7.6–8.2%. This differs from the cached-streaming ASR RTF in Table 8; neither includes diarization. The supplement retains individual timings and the full diagnostic specification.

Table 8: Cached-streaming ASR real-time factor (RTF, lower is faster) per target on one RTX 5090, FP32, batch 1, 40 utterances (647.9 s); excludes diarization. DP/FW use seed 43; TCN uses seed 42. Cache is KiB per target. Effective histories differ.

Adapter	1120 ms	560 ms	160 ms	80 ms	Cache
Dual-path	0.0079	0.0151	0.0515	0.0996	210
Framewise	0.0076	0.0139	0.0469	0.0915	0
TCN	0.0075	0.0144	0.0485	0.0954	52.5