LONGCAP: HOUR-SCALE TIMESTAMPED AUDIO CAPTIONING VIA CHUNK-WISE PROCESSING

Anonymous authors
Paper under double-blind review

ABSTRACT

Describing hours of audio requires accurate event timing and sustained descriptive detail throughout the recording. We present LongCap, a framework for timestamped audio captioning (TAC), with the human-annotated LongCapBench and three diarization-inspired metrics: Timeline Error Rate (TER), Granularity Error Rate (GER), and Caption Similarity (tcpCapSim). LongCap interleaves audio chunks with structured event captions, using relative timestamps and retained context to support length extrapolation within a 1M-token inference context window. Event-level reannotation restores caption detail, while systematic studies examine model scaling, dependency supervision, and self-distillation. LongCap-Diar achieves strong benchmark results and extrapolates to 16 hours of concatenated audio despite training on recordings no longer than 40 minutes. Our TAC models are competitive with Qwen3.5-Omni-Plus on short clips, outperform evaluated baselines in caption similarity and timeline accuracy in segment-level comparisons on LongCapBench-TAC-Long, and caption full recordings approaching two hours. On LongCapBench-TAC, 9B and 35B-A3B offer favorable size–performance tradeoffs, while 122B-A10B substantially improves event-dependency recovery. These findings support long-horizon general audio understanding and provide a foundation for always-on voice agents. Homepage

1 INTRODUCTION

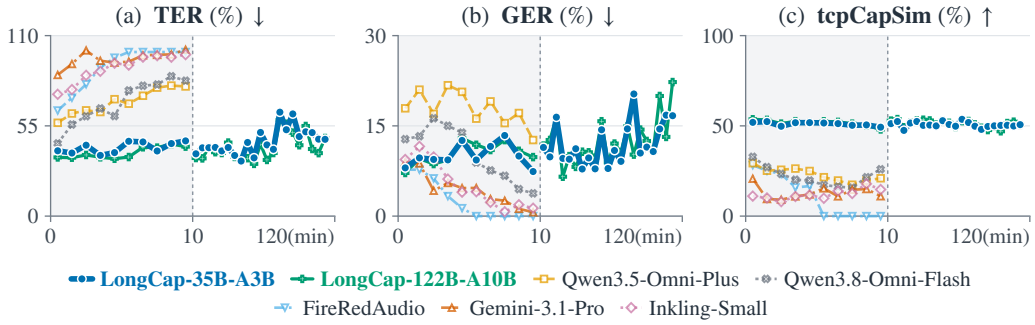


Figure 1: Timestamped audio captioning (TAC) results on the LongCapBench-TAC-Long benchmark. LongCap produces stable and comprehensive audio captions up to 2 hours, while competitors show severe degradation within 10 minutes. (a) TER, (b) GER, and (c) tcpCapSim measure timeline mismatch, granularity mismatch, and caption semantic similarity, respectively.

Audio understanding is expanding from individual tasks toward comprehensive descriptions of recordings. Speech recognition recovers what was said (Radford et al., 2023; Kanda et al., 2020), diarization identifies who spoke when (Anguera et al., 2012; Park et al., 2022), and audio tagging detects sound classes (Gemmeke et al., 2017). Audio captioning brings these aspects together in a free-text description of a short clip (Kim et al., 2019; Drossos et al., 2020). Recent captioners produce more comprehensive descriptions of speech, music, and environmental sound, using either free text or a structured format (Qwen Team, 2025b; Ma et al., 2026; Tian et al., 2026). A growing line of work goes further by describing individual audio events together with their timestamps (Kumar

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.

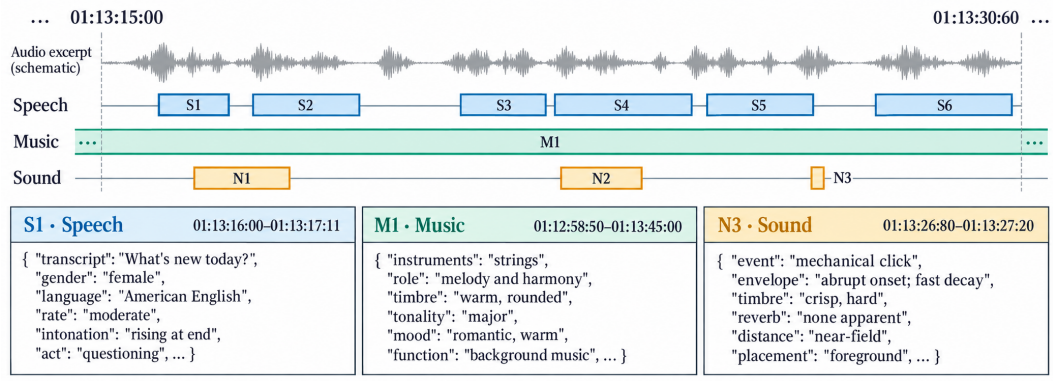


Figure 2: TAC format. A schematic excerpt of long-form audio illustrates overlapping speech, music, and sound events with start/end timestamps and structured attribute dictionaries.

et al., 2026; Rong et al., 2026; Shao et al., 2026). In this work, we refer to this form as *timestamped audio captioning* (TAC): the output is a timeline of audio events, each with start and end timestamps and a caption (Fig. 2). Joint speech recognition and diarization is a special case in which each event caption contains only the transcript and the speaker identity. To make the descriptions easier to parse, we further require every caption to be structured as key–value pairs over a closed, expert-designed key set, so that each aspect of interest can be checked directly.

To extend this representation to hour-long recordings, we develop LongCap together with an evaluation framework and a training recipe. Inspired by diarization error rate (DER) and tcpWER (Fiscus et al., 2007; von Neumann et al., 2023) in speech recognition and diarization, we introduce Timeline Error Rate (TER) to measure timeline mismatch, Granularity Error Rate (GER) to measure event granularity mismatch, and Caption Similarity (tcpCapSim) to measure semantic agreement between captions. We construct LongCapBench, a human-annotated benchmark covering speech recognition, diarization, and TAC, with recordings approaching two hours. Our modeling approach addresses two challenges: *holistic processing* (Qwen Team, 2025b; Ghosh et al., 2026) increases the token distance between an event's audio and its description, and models trained on short recordings can degrade on longer inputs (Narayanan et al., 2019). LongCap uses *chunk-wise processing* with chunk-relative timestamps, interleaving audio chunks and event captions to keep emission local while retaining prior context. We first develop and validate this design for speech recognition and diarization, where supervision is abundant, then extend the recipe to TAC. We examine model scaling from 0.8B to 122B parameters across pre-training and post-training, assessing short-form accuracy and long-form robustness. We further study *density degradation*, explicit supervision of long-range event dependencies, and self-distillation.

LongCap-Diar achieves the lowest macro-averaged recognition and diarization errors among the evaluated systems (Fig. 5) and extrapolates to 16 hours of concatenated audio within a 1M-token context window, despite training on recordings no longer than 40 minutes (App. D.1). For TAC, LongCap is competitive with Qwen3.5-Omni-Plus on short clips and captions full recordings approaching two hours, surpassing evaluated competitors in caption similarity and timeline accuracy in segment-level comparisons on LongCapBench-TAC-Long (Fig. 1). Our scaling studies identify favorable size–performance tradeoffs at 9B and 35B-A3B on LongCapBench-TAC (Fig. 8). The evaluated 122B-A10B model is substantially stronger at recovering event dependencies, achieving 14.8% long-form connectivity F_1 against silver references, compared with 4.2% for 35B-A3B (Fig. 9). These results suggest different scaling needs for event descriptions and dependencies, supporting long-horizon general audio understanding and always-on voice agents.

2 Task Formulation: Timestamped Audio Captioning (TAC)

The formulation of *timestamped audio captioning* (TAC) is inspired by the well-studied task of joint speech recognition and diarization (Fiscus et al., 2007; Watanabe et al., 2020; Cornell et al., 2023), and extends its scope from speech transcription to general audio captioning. Specifically, TAC localizes and describes every audio event in the recording. Given an audio recording X, the

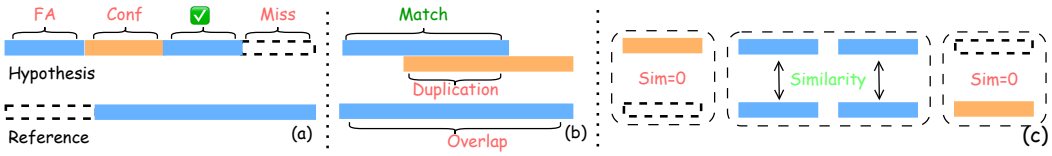


Figure 3: TAC scoring metrics. (a) TER measures timeline mismatch, (b) GER measures event granularity mismatch, and (c) tcpCapSim measures time-constrained caption similarity.

model predicts a list of audio events $E = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]$, where each event $\boldsymbol{e}_i = (t_i^s, t_i^e, c_i)$ is a tuple of start timestamp, end timestamp, and a caption describing the sounds within that span (Fig. 2).

2.1 Formatting

Structured Caption: Unlike prior work that writes the caption as free-running natural language (Qwen Team, 2025b; Ma et al., 2026), we require every event caption to be *structured*: each caption is explicitly a set of key–value pairs, in which the keys come from a pre-defined closed set and each value is a short piece of precise natural language. The structure makes the description explicit and easy to parse: each key of interest can be accessed and analyzed directly.

Attribute Key Set: To define the attribute key set, we collaborated with audio professionals from the speech, music, and film industries. The experts collaborated to make the keys cover common aspects of audio information while avoiding being duplicated, ambiguous or over-professional, converging to 62 keys. A key appears only when it is applicable and is otherwise absent. All attributes, together with the event timestamps, are assembled into a nested JSON object and serialized as plain text, so the TAC task remains an audio-to-text conversion task: it models the conditional distribution $P(E \mid X)$. The detailed design of these keys is given in App. A.1.

2.2 Scoring

Inspired by the Diarization Error Rate (DER) (Fiscus et al., 2007) and tcpWER (von Neumann et al., 2023) in speech recognition and diarization, we derive the *Timeline Error Rate* (TER) and *tcpCapSim* for TAC, and additionally introduce *Granularity Error Rate* (GER) for event granularity mismatch. Figure 3 illustrates TER, GER, and tcpCapSim in panels (a)–(c), respectively; more detailed definitions are given in App. A.2. All three compare predictions $\hat{E} = [\hat{\boldsymbol{e}}_1, \ldots, \hat{\boldsymbol{e}}_{\hat{n}}]$ with reference events $E = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n]$.

Event Grouping: DER and tcpWER optimize over speaker permutations, with utterances already grouped by speaker labels. General audio events lack equivalent identity labels, so their correspondence must be inferred from semantic content. To approximate enumeration of possible groupings, a text LLM proposes $M{=}3$ candidates $\mathcal{G}^{(m)}$. Each event belongs to exactly one group; groups may contain multiple events from either side or have one side empty. Each metric is evaluated under all three groupings and independently retains its best score.

TER (↓): TER measures timeline mismatches for general audio events; its name avoids confusion with speaker diarization. We accumulate miss duration T_{miss} for too few active predicted groups, false alarm duration T_{fa} for too many, and confusion duration T_{conf} for remaining simultaneous activity in different groups. We normalize their sum by T_{ref}, the reference activity duration summed across groups after merging overlapping intervals within each group.

GER (↓): GER measures splitting, merging, and repetition of events through one-to-one matching: each event can match only one counterpart. A split or merge can therefore preserve time coverage while reducing matched overlap. Summed over groups, T_{overlap} is the overlap of reference and prediction activity unions; T_{match} is the maximum one-to-one matched overlap; and T_{dup} is summed prediction event duration minus its union duration, accounting for overlapping predictions.

$$\text{TER} = \min_{\mathcal{G}^{(m)}} \frac{T_{\text{miss}}^{(m)} + T_{\text{fa}}^{(m)} + T_{\text{conf}}^{(m)}}{T_{\text{ref}}^{(m)}}, \qquad \text{GER} = \min_{\mathcal{G}^{(m)}} \frac{T_{\text{overlap}}^{(m)} - T_{\text{match}}^{(m)} + T_{\text{dup}}^{(m)}}{\sum_{i=1}^{n} \left(t_i^e - t_i^s\right)}. \tag{1}$$

Table 1: Statistics of LongCapBench. Recordings are evaluation clips.

Subset	Recordings	Volume (h)	Mean length (min)	Maximum length (min)	Utterances/events
Diar.-short	442	53.01	7.20	11.10	44,860
Diar.-long	15	15.02	60.06	78.16	4,068
TAC-short	1,500	18.50	0.74	1.00	9,380
TAC-long	4	7.05	105.70	117.05	6,520

tcpCapSim (↑): A group receives similarity credit only if a reference event and a predicted event overlap or are at most five seconds apart. We pool values by scored attribute key on each side and ask a text LLM to compare each shared key's values for semantic agreement, scoring them in $[0, 1]$. A missing counterpart or failed time constraint gives zero credit. Precision $P^{(m)}$ and recall $R^{(m)}$ divide the summed attribute scores by predicted and reference attribute counts, respectively, across all groups, counting each key once per group on each side. We take their arithmetic mean, yielding $\text{tcpCapSim} = \max_{\mathcal{G}^{(m)}}(P^{(m)} + R^{(m)})/2$.

2.3 LongCapBench

We curate *LongCapBench* to evaluate speech recognition and diarization alongside TAC, with short- and long-form subsets for both settings. Table 1 summarizes the benchmark statistics. All labels are finalized by human annotators. Benchmark composition is described in App. A.3.

TAC annotation follows a two-stage workflow. Starting from Qwen3.5-Omni-Plus (Qwen Team, 2026b) predictions, annotators first correct event timestamps, then verify each event's attributes. This process presents two main challenges. First, unaided listeners often miss less salient events; model-generated proposals guide event discovery and make verification more reliable and efficient. Second, the breadth of audio attributes exceeds the expertise a generalist can acquire through brief training. We therefore route data batches through specialist groups, each reviewing a small set of related attributes. Review requiring specialist expertise, particularly in music, becomes a throughput bottleneck. Comparing drafts with returned annotations, approximately 65% and 53% of matched start/end boundaries differ on TAC-short and TAC-long, respectively; 58% and 91% of attribute entries are added, removed, or modified before final benchmark curation. The benchmark therefore uses human-reviewed and corrected references, and we retain Qwen3.5-Omni-Plus as an evaluation baseline despite its role in generating the initial drafts. App. A.3 describes the annotation workflow.

3 LongCap

Architecture: LongCap instantiates a TAC model with standard components: Qwen3.5 backbone language models (0.8B, 2B, 9B, 35B-A3B, and 122B-A10B), the Qwen3-Omni-Captioner audio encoder (Qwen Team, 2025b) feeding the backbone through a linear adaptor, and a cross-entropy loss on the text positions. All parameters are unfrozen during training. Qwen3.5 uses hybrid linear and full attention and natively supports a 256k-token context (Qwen Team, 2026a). For contexts beyond 256k tokens, we use YaRN (Peng et al., 2024) to extend the context window to 1M tokens.

3.1 Chunk-wise Processing

Since the output E is serialized as plain text (§2.1), modeling $P(E \mid X)$ reduces to audio-conditioned language modeling. The prevailing recipe (Qwen Team, 2025b; Ghosh et al., 2026) is *holistic processing*: the whole recording X is encoded into the context first, and only then is E decoded autoregressively, conditioned on the entire recording. LongCap instead adopts *chunk-wise processing*, which interleaves perception and emission (Fig. 4). The recording is split into consecutive, non-overlapping chunks $X = [X_1, \ldots, X_K]$. Each chunk X_k starts at τ_k^s and ends at τ_k^e; its label Y_k contains all audio events that end within the chunk.[1] The model then reads and

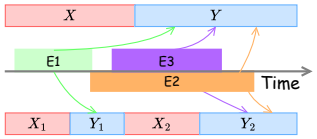


Figure 4: Holistic (top) versus chunk-wise (bottom) processing of a long recording.

[1] Events spanning multiple chunks are assigned only to the chunk in which they end.

writes the interleaved sequence $[X_1, Y_1, X_2, Y_2, \ldots, X_K, Y_K]$, retaining earlier audio chunks and generated labels in context:[2]

$$P(E \mid X) = \prod_{k=1}^{K} P\big(Y_k \mid X_{\leq k},\, Y_{<k}\big). \tag{2}$$

Comparison to Holistic Processing: Compared with holistic processing, chunk-wise processing offers two advantages on hour-long recordings. *Local caption emission*: holistic processing emits an event's description only after the entire recording and all earlier output tokens, turning every prediction into long-context retrieval, where accuracy is known to degrade (Liu et al., 2024; Hsieh et al., 2024). Chunk-wise processing emits each description after the chunk containing the event's end, following chunk-local speech decoding (Jia et al., 2025). This keeps emission near the event endpoint while retaining earlier chunks for long-range evidence. *Reduced length bias*: generation can degrade beyond training lengths in language modeling (Press et al., 2022) and speech recognition (Narayanan et al., 2019). Chunk-wise processing repeats the same local emission step on longer recordings, supporting length extrapolation (Radford et al., 2023; Jia et al., 2025). This ability relies on the base text model's long-context capacity, which we preserve during training (§3.2).

3.2 Long-Form Speech Recognition and Diarization

Speech recognition and diarization is a reduced version of TAC: each event caption keeps only two keys, the speaker identity and the transcript (§2.1). Its supervision, however, is far more abundant than that of the general TAC task, which makes it an ideal test bed for ablating chunk-wise processing in the long-form setup. Our ablations on this task, presented in §4.2, justify three design choices:

Relative timestamps: Absolute timestamps become unreliable beyond the training duration range in our experiments (§4.2). Following Whisper's window-relative timing (Radford et al., 2023), we express event timestamps t_i^s, t_i^e relative to the start of their chunk.[3]

Chunk size: We compare a fixed chunk size of 32 s with random sampling from a continuous uniform distribution $\mathcal{U}(8, 32)$ s or the discrete set $\{8, 16, 24, 32\}$ s. The chunking strategy has only a marginal impact on performance. We adopt the discrete set $\{8, 16, 24, 32\}$ s, randomly sampling each chunk's duration during both training and inference.

Long-form text: To retain the base model's long-context and text capabilities, we include long-form text in the training mixture, following text-mixing strategies in long-context and speech-LLM training (Gao et al., 2025; Peng et al., 2025).

Our diarization model is trained at most 20 minutes, but can extrapolate to 16 hours (concatenated LongCapBench-Diar-Long examples) with stable WER, although the accumulated speaker count is far beyond the training distribution.

3.3 Migration to Timestamped Audio Captioning

We initialize each LongCap model directly from a pretrained Qwen3.5 text backbone and the Qwen3-Omni-Captioner audio encoder. Before the optional extensions below, all training recordings are at most 20 minutes long.

3.3.1 Pre-Training

TAC requires two complementary capabilities: timestamp prediction and audio attribute prediction. Because comprehensive TAC supervision is scarce, we develop these capabilities through multi-task pre-training. Speech recognition and diarization on recordings of up to 20 minutes provide timestamp supervision, while natural-language captions paired with audio chunks of up to 30 seconds teach audio attribute prediction. These annotated chunks are spliced into training recordings of up to 10 minutes in total. Following §3.2, we also include long-form text sequences of up to 256k

[2] Conditioning each Y_k on $X_{\leq k}$ rather than the full X is the causal assumption of chunk-wise processing.

[3] Recording-level event timestamps are recovered by adding the chunk start to the relative predictions. An event crossing the chunk's left boundary carries a negative start.

tokens to preserve long-context capabilities. We mix diarization, captioning, and text data in a 1:1:1 ratio by token count. The resulting model is named `LongCap-Base`. Data curation is described in App. B.1.

3.3.2 Supervised Fine-Tuning (SFT)

We refine the model and its training data in two stages, followed by two optional extensions. We conduct the refinement with the 35B-A3B model and use the resulting data to train models at all backbone sizes. During SFT, captioning, diarization, and text data are mixed in a $3:1:1$ ratio by token count. All SFT data volumes and mean durations below refer to audio retained after filtering.

Seed supervision: We curate 15k hours of audio, with recordings averaging 49 seconds and lasting at most 10 minutes. We split each recording into clips of at most one minute and query Qwen3.5-Omni-Plus to generate structured captions in the format of §2.1. The clips are annotated independently, and their annotations are reassembled into recording-level labels. Fine-tuning `LongCap-Base` on these labels yields `LongCap-SFT-1`, establishing basic TAC capabilities.

Alleviating density degradation: Both Qwen3.5-Omni-Plus and `LongCap-SFT-1` exhibit *density degradation*: later events within a chunk contain fewer filled attributes (Fig. 8c). To restore event-level detail, we curate another 10k hours of audio, with recordings averaging 49 seconds and lasting at most 10 minutes, and annotate them with `LongCap-SFT-1`. Using the predicted timestamps, we extract each event's audio span and query Qwen3.5-Omni-Plus to annotate it independently. Fine-tuning `LongCap-SFT-1` on these refined annotations yields `LongCap-SFT-2`, which maintains more consistent attribute density across event positions.

Optional 1: Length Expansion: The initial training recipe uses recordings no longer than 20 minutes. To test whether longer training examples improve extrapolation to two-hour recordings, we curate an additional 20k hours of audio covering diverse acoustic content. Approximately 10k hours come from recordings longer than 20 minutes, with a maximum duration of 40 minutes. We use `LongCap-SFT-2` to annotate this corpus; the longest resulting audio–caption sequences approach the 256k-token training context limit.

Optional 2: Event Dependency Modeling: Although the model retains the full audio and caption history, its labels largely originate from short-window annotation. To make cross-event dependencies explicit, we use GLM-5.3-Flash (Z.ai, 2026) to analyze the event sequence and, where appropriate, assign each event a semantically related earlier event as its *ancestor*. We include both the current event's ID and its labeled ancestor in the JSON string used as the training target. Following these links recursively forms event chains that describe how events evolve over time, providing explicit supervision for long-range dependencies.

4 Experiments

4.1 Experimental Setup

Training and Inference: We train all five backbone sizes following §3.3, with 100k continued pre-training steps and two SFT epochs at a constant learning rate of 1×10^{-5}. Context parallelism supports 256k-token training sequences. Inference uses a customized vLLM implementation (Kwon et al., 2023) with greedy decoding. Hardware, hyperparameters, and decoding safeguards are detailed in App. B.2.

Benchmarks: We evaluate speech recognition and diarization on AliMeeting (Yu et al., 2022), AISHELL-4 (Fu et al., 2021), AMI-IHM and AMI-SDM, four CHiME-6 sessions (Watanabe et al., 2020), and the LongCapBench-Diar subsets (§2.3). TAC is evaluated on the LongCapBench-TAC subsets (§2.3). For general audio understanding, we use the MMAU test-mini subset (Sakshi et al., 2025) and MMAR (Ma et al., 2025). Following Omni-Captioner (Ma et al., 2026), we first caption each recording with the TAC model, then use a text LLM to answer the benchmark questions from the caption alone.

Competitors: For speech recognition and diarization, our baselines are MOSS Transcribe Diarize (MOSI.AI, 2026), VibeVoice-ASR (Peng et al., 2026b), Doubao ASR 2.0 (Volcengine, 2026), and Qwen3.8-Omni-Flash (Alibaba Cloud, 2026). For TAC, we compare against Qwen3.5-Omni-

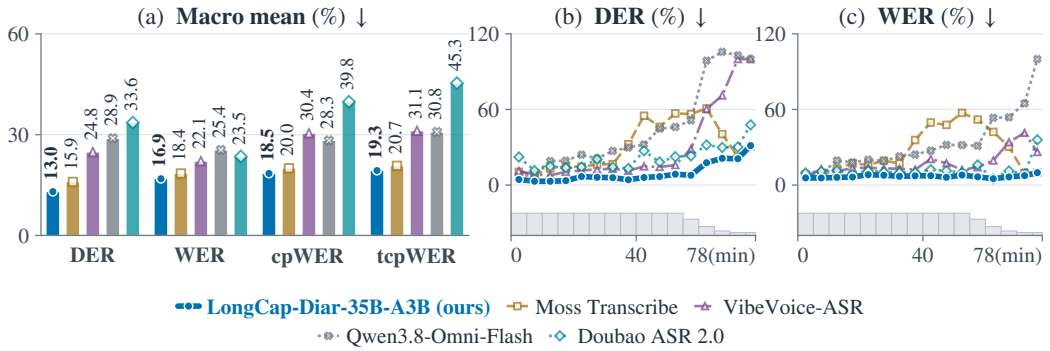


Figure 5: Speech recognition and diarization: (a) DER, WER, cpWER, and tcpWER, macro-averaged over five benchmark conditions; (b–c) LongCapBench-Diar-Long errors in disjoint five-minute windows. Gray bars count contributing recordings. Benchmark-specific checkpoints and AMI overlap are described in App. D.1.

Plus (Qwen Team, 2026b), Qwen3.8-Omni-Flash (Alibaba Cloud, 2026), Gemini-3.1-Pro (Google, 2026a), FireRedAudio (Shen et al., 2026), and Inkling-Small (Thinking Machines Lab, 2026), using commercial APIs or released checkpoints. Differences in output format complicate direct comparisons with LAT (Shao et al., 2026) and AF-Next (Ghosh et al., 2026); we compare their native caption content in App. D.2. App. B describes LongCap data, training, and inference; App. A.2 specifies scoring, and App. D provides additional comparison protocols.

4.2 Diarization Results

Overall performance. LongCap-Diar-35B-A3B achieves the lowest macro-averaged DER, WER, cpWER, and tcpWER in Fig. 5a. We further study temporal robustness on LongCapBench-Diar-Long, whose recordings span 50–78 minutes (Fig. 5b–c). In five-minute windows, WER remains below 10% throughout, while DER stays below 9% through 60 minutes before rising in the sparsely sampled tail. These results show that our chunk-wise model can outperform the evaluated holistic baselines, MOSS Transcribe Diarize and VibeVoice-ASR (MOSI.AI, 2026; Peng et al., 2026b), as well as the commercial Qwen3.8-Omni-Flash (Alibaba Cloud, 2026) and Doubao ASR 2.0 (Volcengine, 2026) APIs. Holistic versus chunk-wise processing is further ablated in Table 6; the absolute/relative timestamp comparison is shown separately in Fig. 6a. Our model is also better at dealing with very long and far-field spontaneous speech (CHiME-6). With a 1M-token context window, LongCap-Diar can further extrapolate to 16 hours of concatenated audio. Details are in §D.1.

Design comparisons. The timestamp and chunk-policy comparisons use 2B or 35B-A3B models trained on reduced data, so their scores are not directly comparable to the main results. In Fig. 6a, relative timestamps slightly worsen WER but avoid absolute timestamps' complete DER failure beyond the 20-minute training span; the absolute-timestamp checkpoint was selected by test WER among 15 checkpoints. Panel (b) shows little sensitivity to chunk size. In (c), the long-text condition has lower WER, but also uses a different training mixture and larger token budget.

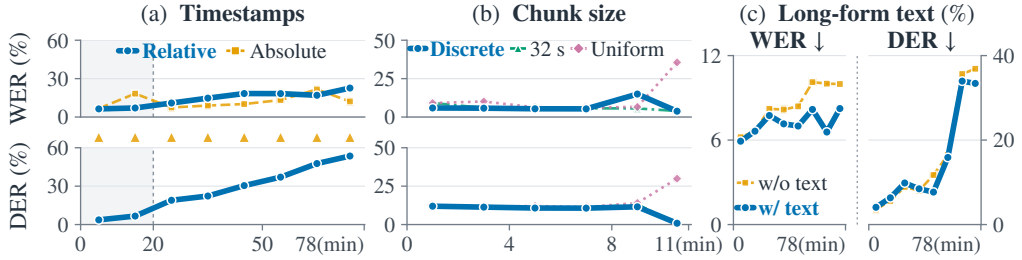


Figure 6: Diarization design comparisons: (a) Relative vs. absolute timestamps. (b) Different chunking strategies. (c) With and without long-form text in the training mixture. Comparison protocols are detailed in App. D.1.

4.3 TIMESTAMPED AUDIO CAPTION RESULTS

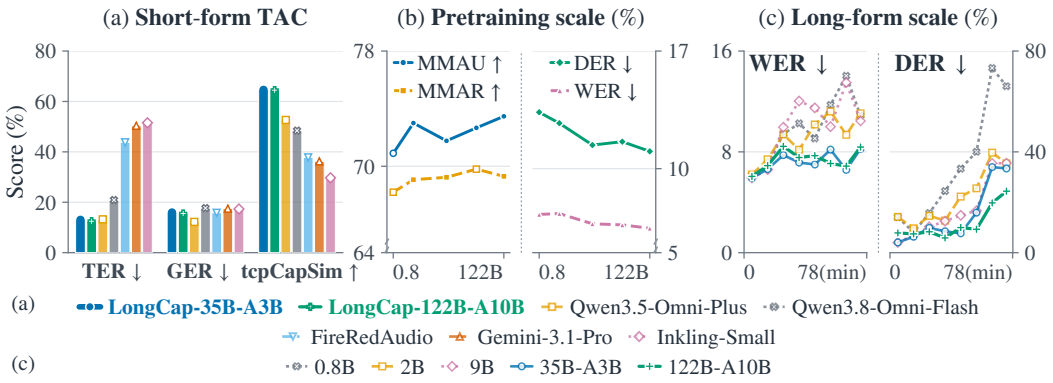


Figure 7: (a) Short-form TAC results on LongCapBench-TAC-Short. (b–c) Continued-pretraining scale comparisons: (b) MMAU/MMAR accuracy and DER/WER averaged equally over English and Chinese evaluation sets; (c) WER and DER over long-form recordings.

Overall Results: LongCap achieves strong TAC performance on both short clips and long recordings. On LongCapBench-TAC-Short (Fig. 7a), both LongCap variants outperform the evaluated baselines in caption similarity and achieve competitive timeline accuracy, while Qwen3.5-Omni-Plus retains lower granularity error. On LongCapBench-TAC-Long (Fig. 1), LongCap maintains higher caption similarity and lower timeline error than the baselines in the segment-level comparison. Full-recording measurements extend to 117.1 minutes, with higher timeline error in some later windows and substantial music granularity errors in the unplotted final partial window. Coverage also decreases from four recordings to one.[4] A supplemental reference-grounded comparison with AF-Next and LAT using native captioning prompts is reported in App. D.2.

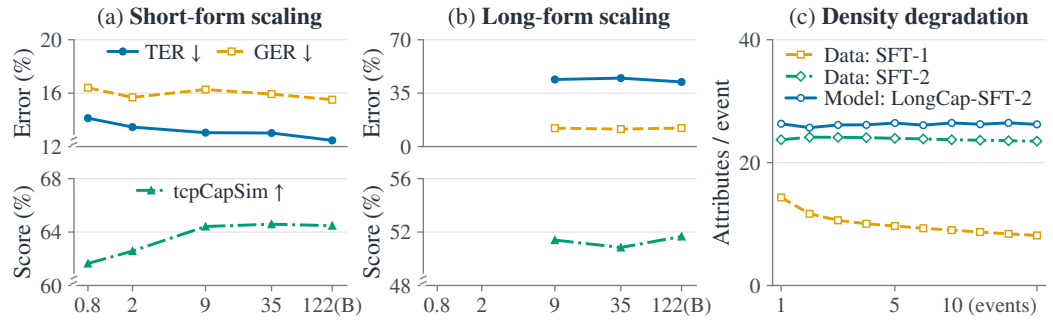


Figure 8: TAC scaling and caption density. (a–b) LongCap-SFT-2 on LongCapBench-TAC-Short and LongCapBench-TAC-Long; 0.8B and 2B lack reliable full-recording outputs. (c) Filled attributes per event by position within the chunk.

Scaling Results: Continued pre-training yields modest gains on short-form audio tasks (Fig. 7b), but larger improvements in long-form recognition and diarization (Fig. 7c). For post-training, Fig. 8a–b compares `LongCap-SFT-2` before data self-distillation. Scaling from 0.8B/2B to 9B/35B-A3B lowers short-form TER and raises tcpCapSim from 61.65/62.58% to 64.42/64.59%, while GER varies little (Fig. 8a). The practical gain is greater on long-form audio: 0.8B and 2B produce frequent invalid or repetitive captions far beyond the training range, precluding reliable full-recording evaluation (Fig. 8b; App. B.2). With limited additional gains at 122B-A10B, 9B and 35B-A3B offer practical model sizes for TAC.

[4]Missing events are already penalized by TER and are therefore not penalized again by GER. On TAC-Long, most semantic groups for some competitors contain no hypothesis events, explaining their very low GER.

Density Degradation: Chunk-wise SFT-1 labels contain fewer filled attributes for later events within each chunk (Fig. 8c). Event-level reannotation in SFT-2 addresses this degradation, producing more consistent attribute density across event positions. `LongCap-SFT-2` learns this improved profile, maintaining event-level detail in its full-recording outputs.

Long-form Expansion Results: The main benefit of long-form expansion is lower full-recording TER (Table 8, App. D.3): 44.84% to 41.09% at 35B-A3B and 42.35% to 39.64% at 122B-A10B, compared with smaller TER gains on short-form audio. This pattern is consistent with extending supervision to longer recordings. Long-form GER also decreases, while tcpCapSim changes are mixed.

Dependency Modeling: We evaluate predicted ancestor links against DeepSeek-V4-Pro and Gemini-3.1-Pro silver annotations (DeepSeek, 2026; Google, 2026a). Link support (LS) measures reference support for individual links; connectivity F_1 measures recovery of direct and transitive reference connections. Both models show substantial gaps on long recordings (Fig. 9). Despite limited captioning gains beyond 35B-A3B, 122B-A10B improves long-form connectivity F_1 from 4.2% to 14.8%. App. D.4 details matching, checkpoints, and GLM labeling controls.

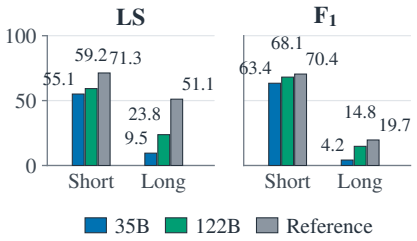


Figure 9: LS and F_1 (%) for 35B-A3B, 122B-A10B, and GLM-5.3-Flash on reference events.

5 Related Work

Long-Form Speech Recognition and Diarization: Earlier work studies long-form generalization (Narayanan et al., 2019) and full-document contexts (Chen et al., 2024). Whisper decodes sequential 30-second windows with window-relative timestamps (Radford et al., 2023). CHiME benchmarks unsegmented recognition and diarization of distant speech (Watanabe et al., 2020; Cornell et al., 2024). Recent speech LLMs jointly emit speaker identities, timestamps, and transcripts (MOSI.AI, 2026; Peng et al., 2026b; Mu et al., 2026); MOSS and VibeVoice-ASR support single-pass inputs of up to 90 and 60 minutes, respectively. Chunk-wise decoding extends from ASR (Jia et al., 2025) to joint transcription and diarization with cross-chunk speaker caches (Shi et al., 2026; Peng et al., 2026a). LongCap extends this approach to structured TAC and validates length extrapolation in recognition and diarization (§4.2).

Audio Captioning: Audio captioning classically pairs a clip of ten to thirty seconds with one free-text sentence (Kim et al., 2019; Drossos et al., 2020). Bagpiper uses rich captions for open-ended audio tasks (Tian et al., 2026). Omni-Captioner studies the tradeoff between descriptive detail and hallucination (Ma et al., 2026); the Qwen3-Omni-Captioner model card advises inputs of at most 30 seconds (Qwen Team, 2025a). Timestamped variants attach start and end times to each described event, echoing dense video captioning (Krishna et al., 2017). The TAC model (Kumar et al., 2026), distinct from our task acronym, emits typed, timestamped descriptions; its TAC-V pipeline processes independent 20-second audio chunks. AudioMap names the task time-aware dense audio captioning (Rong et al., 2026), AudioTime supplies temporally aligned training data (Xie et al., 2025), LAT reaches 30-minute inputs and contributes the human-verified LAT-Bench (Shao et al., 2026), and audio diarization localizes open-set events without describing them (Werning & Haeb-Umbach, 2026). Hour-scale inputs have been reached for question answering and temporal grounding (Kutsakov et al., 2026). LongCap combines retained context across chunks (§3.1), 62-key structured captions (§2.1), and long-form training (§3.3) with human-labeled LongCapBench evaluation approaching two hours (§2.3).

6 Conclusion

LongCap enables hour-scale TAC through chunk-wise generation, relative timestamps, and detailed event supervision. On LongCapBench-TAC, 9B and 35B-A3B offer favorable size–performance tradeoffs, while 122B-A10B better recovers event dependencies. This distinction highlights different scaling needs for describing events and connecting them over time, providing a foundation for long-horizon audio understanding and always-on voice agents.

AI USE STATEMENT

We used generative AI extensively for data construction, annotation, refinement, and quality assessment. We also used LLMs for event grouping and semantic-similarity evaluation, as described in §2.2. AI assistants supported portions of code development and experiment management, and helped polish the manuscript and proofread its text. The core research ideas were conceived entirely by the human researchers.

ETHICS STATEMENT

Our data preparation uses public and private audio resources and model-assisted annotation; human annotators finalize the benchmark labels. Our collaboration with audio professionals and human annotators followed standard ethical practices and respected their rights, welfare, and privacy. Speech and speaker attributes can expose sensitive information, and generated descriptions may contain errors or reflect biases in the source data and annotation models. These limitations matter for applications involving speaker profiling or consequential decisions. The four LongCapBench-TAC-Long recordings have research-use permission; this does not by itself establish permission to redistribute their audio.

REPRODUCIBILITY STATEMENT

Appendix A specifies the attribute schema, scoring rules, and benchmark composition. Appendix B describes data curation, training configurations, and inference safeguards, and Appendix C provides the seed and event-wise annotation prompts. Appendix D documents additional evaluation protocols, checkpoint comparisons, and their limitations.

REFERENCES

Alibaba Cloud. Qwen-Omni: API documentation. Alibaba Cloud Model Studio documentation, 2026. URL https://help.aliyun.com/zh/model-studio/qwen-omni.

Xavier Anguera, Simon Bozonnet, Nicholas Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals. Speaker diarization: A review of recent research. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(2):356–370, 2012. doi: 10.1109/TASL.2011.2125954.

Hervé Bredin. pyannote.metrics: A toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems. In *Interspeech 2017*, pp. 3587–3591. ISCA, 2017. doi: 10.21437/Interspeech.2017-411. URL https://www.isca-archive.org/interspeech_2017/bredin17_interspeech.html.

William Chen, Takatomo Kano, Atsunori Ogawa, Marc Delcroix, and Shinji Watanabe. Train long and test long: Leveraging full document contexts in speech processing. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 13066–13070, 2024. doi: 10.1109/ICASSP48485.2024.10446727.

Samuele Cornell, Matthew Wiesner, Shinji Watanabe, Desh Raj, Xuankai Chang, Paola Garcia, Matthew Maciejewski, Yoshiki Masuyama, Zhong-Qiu Wang, Stefano Squartini, and Sanjeev Khudanpur. 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, 2023. doi: 10.21437/CHiME.2023-1. URL https://www.isca-archive.org/chime_2023/cornell23_chime.html.

Samuele Cornell, Tae Jin Park, He Huang, Christoph Boeddeker, Xuankai Chang, Matthew Maciejewski, Matthew S. Wiesner, Paola Garcia, and Shinji Watanabe. 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, 2024. doi: 10.21437/CHiME.2024-1. URL https://www.isca-archive.org/chime_2024/cornell24_chime.html.

DeepSeek. Models & pricing. DeepSeek API documentation, 2026. URL https://api-docs.deepseek.com/quick_start/pricing/. Accessed September 26, 2026.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. Clotho: An audio captioning dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 736–740, 2020. doi: 10.1109/ICASSP40776.2020.9052990. URL https://doi.org/10.1109/ICASSP40776.2020.9052990.

Jonathan G. Fiscus, Jerome Ajot, and John S. Garofolo. The rich transcription 2007 meeting recognition evaluation. In Rainer Stiefelhagen, Rachel Bowers, and Jonathan G. Fiscus (eds.), *Multimodal Technologies for Perception of Humans, International Evaluation Workshops CLEAR 2007 and RT 2007, Baltimore, MD, USA, May 8–11, 2007, Revised Selected Papers*, volume 4625 of *Lecture Notes in Computer Science*, pp. 373–389. Springer, 2007. doi: 10.1007/978-3-540-68585-2_36.

Yihui Fu, Luyao Cheng, Shubo Lv, Yukai Jv, Yuxiang Kong, Zhuo Chen, Yanxin Hu, Lei Xie, Jian Wu, Hui Bu, Xin Xu, Jun Du, and Jingdong Chen. AISHELL-4: An open source dataset for speech enhancement, separation, recognition and speaker diarization in conference scenario. In *Interspeech 2021*, pp. 3665–3669, 2021. doi: 10.21437/Interspeech.2021-1397. URL https://www.isca-archive.org/interspeech_2021/fu21b_interspeech.html.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. How to train long-context language models (effectively). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7376–7399, 2025. doi: 10.18653/v1/2025.acl-long.366. URL https://aclanthology.org/2025.acl-long.366/.

Zhifu Gao, Shiliang Zhang, Ian McLoughlin, and Zhijie Yan. Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition. In *Interspeech 2022*, pp. 2063–2067, 2022. doi: 10.21437/Interspeech.2022-9996. URL https://www.isca-archive.org/interspeech_2022/gao22b_interspeech.html.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 776–780, 2017. doi: 10.1109/ICASSP.2017.7952261.

Sreyan Ghosh, Arushi Goel, Kaousheik Jayakumar, Lasha Koroshinadze, Nishit Anand, Zhifeng Kong, Siddharth Gururani, Sang-gil Lee, Jaehyeon Kim, Aya Aljafari, Chao-Han Huck Yang, Sungwon Kim, Ramani Duraiswami, Dinesh Manocha, Mohammad Shoeybi, Bryan Catanzaro, Ming-Yu Liu, and Wei Ping. Audio Flamingo Next: Next-generation open audio-language models for speech, sound, and music. *CoRR*, abs/2604.10905, 2026. doi: 10.48550/ARXIV.2604.10905. URL https://doi.org/10.48550/arXiv.2604.10905.

Google. Gemini 3.1 Pro Preview. Google Gemini API documentation, 2026a. URL https://ai.google.dev/gemini-api/docs/models/gemini-3.1-pro-preview.

Google. Gemini 3 Flash Preview. Google Gemini API documentation, 2026b. URL https://ai.google.dev/gemini-api/docs/models/gemini-3-flash-preview.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling (COLM)*, 2024. URL https://arxiv.org/abs/2404.06654. arXiv:2404.06654.

Junteng Jia, Gil Keren, Wei Zhou, Egor Lakomkin, Xiaohui Zhang, Chunyang Wu, Frank Seide, Jay Mahadeokar, and Ozlem Kalinli. Efficient streaming LLM for speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025. doi: 10.1109/ICASSP49660.2025.10890853. URL https://arxiv.org/abs/2410.03752.

Naoyuki Kanda, Yashesh Gaur, Xiaofei Wang, Zhong Meng, and Takuya Yoshioka. Serialized output training for end-to-end overlapped speech recognition. In *Interspeech 2020*, pp. 2797–2801, 2020. doi: 10.21437/Interspeech.2020-999. URL https://www.isca-archive.org/interspeech_2020/kanda20b_interspeech.html.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. AudioCaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 119–132, 2019. doi: 10.18653/v1/N19-1011. URL https://aclanthology.org/N19-1011/.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 706–715, 2017. URL https://openaccess.thecvf.com/content_iccv_2017/html/Krishna_Dense-Captioning_Events_in_ICCV_2017_paper.html.

Sonal Kumar, Prem Seetharaman, Ke Chen, Oriol Nieto, Jiaqi Su, Zhepei Wang, Rithesh Kumar, Dinesh Manocha, Nicholas J. Bryan, Zeyu Jin, and Justin Salamon. TAC: Timestamped audio captioning. *arXiv preprint arXiv:2602.15766*, 2026.

Aleksandr Kutsakov, Mariia Sadovina, Georgii Gospodinov, Alexandr Maximenko, Oleg Kutuzov, Pavel Bogomolov, and Fyodor Minkin. GigaChat audio: Time-aware large audio language model. In *Interspeech 2026*, pp. 5835–5840, 2026. doi: 10.21437/Interspeech.2026-2343. URL https://www.isca-archive.org/interspeech_2026/kutsakov26_interspeech.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, pp. 611–626, 2023. doi: 10.1145/3600006.3613165. URL https://arxiv.org/abs/2309.06180.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Ziyang Ma, Yinghao Ma, Yanqiao Zhu, Chen Yang, Yi-Wen Chao, Ruiyang Xu, Wenxi Chen, Yuanzhe Chen, Zhuo Chen, Jian Cong, Kai Li, Keliang Li, Siyou Li, Xinfeng Li, Xiquan Li, Zheng Lian, Yuzhe Liang, Minghao Liu, Zhikang Niu, Tianrui Wang, Yuping Wang, Yuxuan Wang, Yihao Wu, Guanrou Yang, Jianwei Yu, Ruibin Yuan, Zhisheng Zheng, Ziya Zhou, Haina Zhu, Wei Xue, Emmanouil Benetos, Kai Yu, Eng-Siong Chng, and Xie Chen. MMAR: A challenging benchmark for deep reasoning in speech, audio, music, and their mix. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2505.13032.

Ziyang Ma, Ruiyang Xu, Zhenghao Xing, Yunfei Chu, Yuxuan Wang, Jinzheng He, Jin Xu, Pheng-Ann Heng, Kai Yu, Junyang Lin, Eng Siong Chng, and Xie Chen. Omni-Captioner: Data pipeline, models, and benchmark for omni detailed perception. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.12720.

MOSI.AI. MOSS transcribe diarize technical report. *arXiv preprint arXiv:2601.01554*, 2026.

Bingshen Mu, Xian Shi, Xiong Wang, Zhifang Guo, Ting He, Xize Cheng, Yu Xi, Jin Xu, and Lei Xie. DiaScriber: A speech LLM for joint diarization and transcription in multi-speaker scenarios. *CoRR*, abs/2608.22796, 2026.

Arun Narayanan, Rohit Prabhavalkar, Chung-Cheng Chiu, David Rybach, Tara N. Sainath, and Trevor Strohman. Recognizing long-form speech using streaming end-to-end models. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 920–927, 2019. doi: 10.1109/ASRU46091.2019.9003913. URL https://arxiv.org/abs/1910.11455.

Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J. Han, Shinji Watanabe, and Shrikanth Narayanan. A review of speaker diarization: Recent advances with deep learning. *Computer Speech & Language*, 72:101317, 2022. doi: 10.1016/j.csl.2021.101317. URL https://arxiv.org/abs/2101.09624.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=wHBfxhZu1u.

Jing Peng, Ziyi Chen, Haoyu Li, Yucheng Wang, Duo Ma, Mengtian Li, Yunfan Du, Dezhu Xu, Yu Xi, Kai Yu, and Shuai Wang. G-STAR: End-to-end global speaker-tracking attributed recognition. *CoRR*, abs/2603.10468, 2026a.

Yifan Peng, Krishna C. Puvvada, Zhehuai Chen, Piotr Żelasko, He Huang, Kunal Dhawan, Ke Hu, Shinji Watanabe, Jagadeesh Balam, and Boris Ginsburg. VoiceTextBlender: Augmenting large language models with speech capabilities via single-stage joint speech-text supervised fine-tuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5787–5802, 2025. doi: 10.18653/v1/2025.naacl-long.298. URL https://aclanthology.org/2025.naacl-long.298/.

Zhiliang Peng, Jianwei Yu, Yaoyao Chang, et al. VibeVoice-ASR technical report. *arXiv preprint arXiv:2601.18184*, 2026b.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2108.12409. arXiv:2108.12409.

Qwen Team. Qwen3-Omni-30B-A3B-Captioner. Model card, Hugging Face, 2025a. URL https://huggingface.co/Qwen/Qwen3-Omni-30B-A3B-Captioner.

Qwen Team. Qwen3-Omni technical report. *CoRR*, abs/2509.17765, 2025b. doi: 10.48550/ARXIV.2509.17765. URL https://doi.org/10.48550/arXiv.2509.17765.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.5-Omni technical report. *CoRR*, abs/2604.15804, 2026b. doi: 10.48550/ARXIV.2604.15804. URL https://doi.org/10.48550/arXiv.2604.15804.

Qwen Team. Qwen3-ASR technical report. *CoRR*, abs/2601.21337, 2026c. URL https://arxiv.org/abs/2601.21337.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, 2023. URL https://proceedings.mlr.press/v202/radford23a.html.

Yan Rong, Fengji Ma, Xu Li, Jinting Wang, Chen Zhang, and Li Liu. AudioMap: Cloze-and-choice reinforcement learning for time-aware dense audio captioning. *arXiv preprint arXiv:2608.09559*, 2026.

S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. MMAU: A massive multi-task audio understanding and reasoning benchmark. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://iclr.cc/virtual/2025/poster/29528.

Mingchen Shao, Hang Su, Wenjie Tian, Bingshen Mu, Zhennan Lin, Lichun Fan, Zhenbo Luo, Jian Luan, and Lei Xie. Listening with time: Precise temporal awareness for long-form audio understanding. *arXiv preprint arXiv:2604.22245*, 2026.

Feiyu Shen, Fenglong Xie, Junjie Li, Kun Xie, Lei Xie, Xu Tang, Xuelong Geng, Yan Jia, Yao Hu, Yichen Han, Yichen Wu, Ziqi Dai, Junjie Chen, Kai Huang, Manzhen Wei, and Yixuan Li. FireRedAudio: A general-purpose audio language model with decoupled continuous representations for understanding and generation. *arXiv preprint arXiv:2608.24168*, 2026. URL https://arxiv.org/abs/2608.24168.

Mohan Shi, Xiong Xiao, Ruchao Fan, Shaoshi Ling, and Jinyu Li. Train short, infer long: Speech-LLM enables zero-shot streamable joint ASR and diarization on long audio. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 17442–17446, 2026. doi: 10.1109/ICASSP55912.2026.11464726. URL https://arxiv.org/abs/2511.16046.

Team OLMo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. OLMo 3, 2025. URL https://arxiv.org/abs/2512.13961.

Thinking Machines Lab. Inkling-Small. Model card, Hugging Face, 2026. URL https://huggingface.co/thinkingmachines/Inkling-Small.

Jinchuan Tian, Haoran Wang, Bo-Hao Su, Chien-yu Huang, Qingzheng Wang, Jiatong Shi, William Chen, Xun Gong, Siddhant Arora, Chin-Jou Li, Masao Someki, Takashi Maekaku, Keita Goto, Yusuke Shinohara, Jin Sakuma, Chao-Han Huck Yang, and Shinji Watanabe. Bagpiper: Solving open-ended audio tasks via rich captions. In *Conference on Language Modeling (COLM)*, 2026. URL https://arxiv.org/abs/2602.05220.

Volcengine. Doubao Speech: Recording file recognition API (standard version). API documentation, 2026. URL https://www.volcengine.com/docs/6561/1354868.

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 *Proceedings of the 7th International Workshop on Speech Processing in Everyday Environments (CHiME)*, 2023. URL https://arxiv.org/abs/2307.11394.

Hongji Wang, Chengdong Liang, Shuai Wang, Zhengyang Chen, Binbin Zhang, Xu Xiang, Yanlei Deng, and Yanmin Qian. WeSpeaker: A research and production oriented speaker embedding learning toolkit. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10096626. URL https://arxiv.org/abs/2210.17016.

Shinji Watanabe, Michael Mandel, Jon Barker, Emmanuel Vincent, Ashish Arora, Xuankai Chang, Sanjeev Khudanpur, Vimal Manohar, Daniel Povey, Desh Raj, David Snyder, Aswin Shanmugam Subramanian, Jan Trmal, Bar Ben Yair, Christoph Boeddeker, Zhaoheng Ni, Yusuke Fujita, Shota Horiguchi, Naoyuki Kanda, Takuya Yoshioka, and Neville Ryant. CHiME-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings. *arXiv preprint arXiv:2004.09249*, 2020. URL https://arxiv.org/abs/2004.09249.

Alexander Werning and Reinhold Haeb-Umbach. Audio diarization: A new paradigm for exploring audio recordings with unknown event classes. *arXiv preprint arXiv:2607.12703*, 2026.

Zeyu Xie, Xuenan Xu, Zhizheng Wu, and Mengyue Wu. AudioTime: A temporally-aligned audio-text benchmark dataset. In *ICASSP 2025 – 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025. doi: 10.1109/ICASSP49660.2025.10889879.

Fan Yu, Shiliang Zhang, Yihui Fu, Lei Xie, Siqi Zheng, Zhihao Du, Weilong Huang, Pengcheng Guo, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. M2MeT: The ICASSP 2022 multi-channel multi-party meeting transcription challenge. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6167–6171, 2022. doi: 10.1109/ICASSP43922.2022.9746465. URL https://arxiv.org/abs/2110.07393.

Z.ai. GLM-5.3-Flash/FlashX. Z.ai model documentation, 2026. URL https://docs.z.ai/guides/vlm/glm-5.3-flash. Accessed September 26, 2026.

A SCORING AND BENCHMARK

A.1 ATTRIBUTE KEY SET

The structured caption schema contains 62 unique keys organized into six top-level categories. The tree below presents the complete key set in English. Each leaf is marked by the event types to which it applies: `[S]` for speech, `[M]` for music, and `[E]` for sound events. Speech, music, and sound events draw on 33, 25, and 9 keys, respectively; shared keys appear once with multiple marks. A key is included in an event's caption only when applicable. The numeric prefixes retain identifiers from the broader annotation taxonomy rather than defining a consecutive list. Subcategories 1.3, 3.2, 4.2, 6.3, and 6.4 have no retained keys in this event-level schema and are intentionally absent; the displayed 62-key set is complete.

```
1. Source Identity & Inventory
|-- 1.1 Human Voice Sources
|   |-- speaker identity                                  [S]
|   |-- speaker age                                       [S]
|   |-- speaker gender                                    [S]
|   `-- vocal role type                                   [S]
|-- 1.2 Musical Sources
|   |-- instrument list                                   [M]
|   |-- vocal type                                        [M]
|   `-- instrument role                                   [M]
`-- 1.4 Source Provenance & Authenticity
    |-- source authenticity                               [E]
    `-- edit / splice detection                           [E]

2. Linguistic & Verbal Content
|-- 2.1 Transcription
|   |-- orthographic transcription                        [S]
|   `-- lyrics transcription                              [M]
|-- 2.2 Language & Dialect
|   |-- language identification                           [S]
|   |-- dialect                                           [S]
|   |-- accent                                            [S]
|   `-- code-switching                                    [S]
|-- 2.3 Lexical & Grammatical Properties
|   |-- vocabulary level                                  [S]
|   |-- syntactic complexity                              [S]
|   `-- named entities                                    [S]
`-- 2.4 Disfluency & Non-lexical Vocalizations
    |-- disfluency types                                  [S]
    |-- non-lexical vocalizations                         [S]
    `-- backchannel tokens                                [S]

3. Acoustic & Signal Properties
|-- 3.1 Loudness & Dynamics
|   `-- loudness contour                                  [S,E]
|-- 3.3 Timbre & Texture
|   |-- voice quality                                     [S]
|   |-- vocal acoustic detail                             [S]
|   |-- instrument timbre descriptors                     [M]
|   `-- timbre & texture (subcategory asked as one key)   [E]
|-- 3.4 Speech Prosody
|   |-- speech rate                                       [S]
|   |-- stress & prominence                               [S]
|   |-- pause distribution                                [S]
|   `-- pitch and intonation                              [S]
|-- 3.5 Musical Acoustic Properties
|   |-- tempo / bpm                                       [M]
|   |-- beat strength                                     [M]
|   |-- key / tonality                                    [M]
|   |-- chord progression                                 [M]
|   |-- main melody                                       [M]
|   |-- melody contour                                    [M]
|   |-- key modulation / tonal changes                    [M]
|   `-- musical texture type                              [M]
|-- 3.6 Singing Voice Attributes
|   |-- vocal technique                                   [M]
|   |-- singing timbre                                    [M]
|   `-- dynamics in singing                               [M]
|-- 3.7 Recording & Signal Quality
|   `-- perceived audio quality                           [S]
```

```
|-- 3.8 Production & Sound Design
|   `-- intentional effects                          [M]
`-- 3.9 Instrumental Performance Attributes
    |-- instrumental articulation                    [M]
    `-- playing dynamics                             [M]

4. Temporal Structure
|-- 4.1 Segmentation & Boundaries
|   `-- music structure sections                     [M]
`-- 4.3 Temporal Relationships & Dynamics
    |-- tempo changes / rhythmic evolution           [M]
    `-- turn-taking patterns                         [S]

5. Spatial & Environmental Properties
|-- 5.1 Room & Environmental Acoustics
|   |-- reverberation character                      [M,E]
|   `-- distance estimation                          [E]
|-- 5.2 Noise Environment
|   `-- foreground-background separation             [E]
`-- 5.3 Environmental & Event Sound Sources
    `-- sound event identification                   [E]

6. Semantic & Pragmatic & Cultural Interpretation
|-- 6.1 Emotion & Affect
|   |-- speaker emotion                              [S]
|   |-- emotional intensity / arousal                [S,M]
|   |-- valence trajectory                           [S]
|   `-- music mood                                   [M]
|-- 6.2 Intent & Pragmatics
|   |-- speaker intent                               [S]
|   |-- speech act type                              [S]
|   |-- subtext                                      [S]
|   `-- rhetorical strategy                          [S]
|-- 6.5 Cultural, Domain & Contextual Knowledge
|   `-- cultural context                             [S]
`-- 6.6 Cross-modal Interaction
    `-- sound function                               [S,M,E]
```

A.2 Scoring Rules

Diarization Error Rate (DER): We use `pyannote.metrics` (Bredin, 2017) with optimal one-to-one speaker assignment. DER (Fiscus et al., 2007) divides missed-speech, false-alarm, and speaker-confusion duration by reference speaker duration. Overlapping speech is included, counting simultaneous speakers separately. The 0.25 s boundary collar excludes 0.125 s on either side of each reference boundary.

cpWER and tcpWER: MeetEval (von Neumann et al., 2023) concatenates each speaker's utterances chronologically and finds the speaker permutation minimizing substitutions, deletions, and insertions. cpWER divides this count by reference transcript length; tcpWER restricts word alignment using estimated word timings, extending hypothesis intervals by 5 s on each side. This allowance permits temporal matching without excluding boundary regions. Both transcripts are normalized, scoring English by word and Chinese by character. Speaker-agnostic English WER uses Whisper's `EnglishTextNormalizer` (Radford et al., 2023).[5]

TAC event grouping: DER and tcpWER optimize over speaker permutations, using speaker labels to define utterance groups. General audio events lack equivalent identity labels, so we approximate enumeration of possible groupings with three proposals from DeepSeek-V4-Flash (DeepSeek, 2026) with thinking enabled. Using the event intervals $[t_i^s, t_i^e)$ defined in §2, each proposal groups reference and predicted events describing the same occurrence. Each event belongs to one group, with multiple events or an empty side allowed. Repeated occurrences from the same speaker or sound source remain distinct. For each proposal, grouping advances in 60 s steps with a 15 s look-back, retaining full intervals and merging groups that share events across windows into a recording-level grouping. Each metric independently retains its best result across the three candidates: the lowest TER, the lowest GER, or the highest tcpCapSim (§2.2).

TER: TER applies DER accounting to semantic groups. Merge overlapping intervals within each group separately on each side. Too few active predicted groups count as misses, too many as false alarms, and remaining activity in different groups as confusion. Divide these error durations by the merged reference duration T_{ref} in Eq. (1). Overlap between different groups counts separately. TER and GER use full intervals without a boundary collar.

GER: GER compares how shared activity is divided into individual events. One-to-one matching permits each event only one counterpart, so splitting or merging can reduce matched overlap even when time coverage is unchanged. In Eq. (1), T_{overlap} sums the overlap of merged reference and predicted activity within groups; T_{match} is the maximum total overlap under one-to-one event matching within groups; and T_{dup} is summed prediction duration minus merged prediction duration within groups. Divide $T_{\text{overlap}} - T_{\text{match}} + T_{\text{dup}}$ by the total reference event duration $\sum_i (t_i^e - t_i^s)$, counting overlapping events separately. For example, splitting a reference event $[0, 10)$ into predictions $[0, 5)$ and $[5, 10)$ in one group preserves all 10 seconds of coverage, but only one 5-second piece can match the reference event. Thus TER $= 0$ while GER $= (10 - 5)/10 = 0.5$. Merging these two reference events into one prediction gives the same penalty. Predicting $[0, 10)$ twice for a single reference event instead adds 10 seconds through T_{dup}, giving GER $= 1$ despite unchanged coverage.

tcpCapSim: Pool nonempty caption values by key within each group, counting each key once per side. Scoring includes `type` and excludes free-text `caption` and structural `speaker_id`. A group receives credit only if some reference and prediction intervals overlap or are at most 5 s apart. This matching allowance does not exclude boundary regions. DeepSeek-V4-Flash scores each shared key's semantic agreement in $\{0, 0.25, 0.5, 0.75, 1\}$; missing counterparts and groups failing the time constraint receive zero. Summed credit divided by predicted and reference key counts gives precision and recall. tcpCapSim is their arithmetic mean, as described in §2.2.

Aggregation: All metrics use micro aggregation within each benchmark: sum error counts or durations, similarity credit, and their denominators across recordings before computing rates. Reported cross-benchmark macro averages weight benchmark scores equally.

[5] LongCapBench cpWER/tcpWER use WeTextProcessing's English/Chinese normalizers; CHiME-6 uses CHiME8 normalization.

A.3 LONGCAPBENCH DETAILS

LongCapBench intentionally covers both English and Chinese in its short- and long-form subsets for diarization and TAC. We exclude every LongCapBench source recording in its entirety from our training corpus. Table 1 summarizes the subset sizes and durations.

TAC annotation workflow: Starting from Qwen3.5-Omni-Plus (Qwen Team, 2026b) predictions, annotators first correct event timestamps, then verify each event's attributes individually. Model proposals guide event discovery. We assign related attributes to specialized annotator groups, and data batches pass through these groups, each checking a small set of related attributes. Specialist review, particularly for music, is a throughput bottleneck. Maintaining consistency across events and resolving ambiguous annotation rules introduce further difficulties.

LongCapBench-Diar-Short: The short-form diarization subset covers movie excerpts, meetings, podcasts, and casual conversations involving multiple speakers. These sources provide both scripted and spontaneous speech with varied speaker-turn patterns.

LongCapBench-Diar-Long: The long-form diarization subset contains 15 recordings, primarily meetings lasting approximately 50–78 minutes, with 7–19 speakers per recording and a mean of 9.7 speakers. It evaluates sustained transcription and speaker tracking over long conversations.

LongCapBench-TAC-Short: The short-form TAC subset covers speech, music, and sound effects. Music includes both instrumental and vocal content. Movie clips provide representative mixtures of these sound types within the same scene.

LongCapBench-TAC-Long: The long-form TAC subset contains two movies and two shows, with one movie and one show in each of English and Chinese. The movies provide extended mixtures of speech, music, and sound effects, while the shows provide long stretches of conversational and performed speech. All four recordings have licenses permitting research use. Table 2 lists the recordings and their source documentation.

Table 2: Sources and licensing documentation for LongCapBench-TAC-Long

Recording	Language	Min.	Source
A Star Is Born (1937)	English	110.67	link
Personal Tailor (2013)	Chinese	117.05	link
92nd Oscars (2020)	English	93.60	link
Roast! (S3E05)	Chinese	101.49	link

B IMPLEMENTATION

B.1 DATA CURATION

The initial pre-training and SFT recipe uses audio examples of at most 20 minutes, including examples assembled by splicing. Optional length expansion extends the maximum duration to 40 minutes.

B.1.1 PRE-TRAINING DATA

Diarization data: We curate speech-only recordings from web resources and use a cascaded pipeline for automatic annotation and quality filtering. To reduce speaker-clustering degradation on long recordings, we first split the audio into recordings of at most 20 minutes and identify their languages with Whisper (Radford et al., 2023). Paraformer (Gao et al., 2022) and Whisper large-v3-turbo perform the initial transcription of Chinese and English audio, respectively, producing text and timestamps. WeSpeaker (Wang et al., 2023) extracts speaker embeddings, which are clustered to assign utterance-level speaker labels. We then refine utterance boundaries using temporal alignment and local energy information to reduce the risk of clipping speech. Finally, Qwen3-ASR (Qwen Team, 2026c) performs a second transcription pass, whose output provides the final transcript. We filter low-quality recordings based on agreement between the two transcription passes, the proportion of speech, and the token rate.

Clip-level natural-language captions: We curate audio from private resources and publicly available datasets, covering speech, non-speech sounds, and their mixtures. The audio is semantically segmented into chunks of up to 30 seconds, and Qwen3-Omni-Captioner (Qwen Team, 2025a) generates a natural-language caption for each chunk. We apply lightweight filtering to remove obvious labeling failures, such as repetitive text loops. After chunk-level annotation, we splice the chunks and their captions into training recordings with an overall duration of up to 10 minutes. Speech constitutes the majority of this corpus; non-speech content is harder to obtain and accounts for a smaller share.

Long-form text: We use the Dolma3 Longmino Mix corpus for OLMo-3 (Team OLMo et al., 2025).[6] Its original examples contain at most 64k tokens. We concatenate examples from the same domain to construct training contexts of up to 256k tokens. We sample across short and long sequences to keep the sequence-length distribution approximately uniform.

B.1.2 SUPERVISED FINE-TUNING DATA

The SFT audio follows the same broad source and content distribution as the captioning pre-training data, with additional attention to balancing coverage across speech, music, sound effects, and their mixtures. The two initial stages in §3.3 use 15k and 10k hours of audio retained after filtering for seed supervision and event-wise reannotation, respectively. These recordings average 49 seconds and last at most 10 minutes. Optional length expansion adds 20k hours of audio with broad acoustic coverage, including approximately 10k hours from recordings longer than 20 minutes, with a maximum duration of 40 minutes. All of this SFT audio is held out from the pre-training corpus. The seed and event-wise annotation prompts are provided in App. C.

B.2 TRAINING AND INFERENCE DETAILS

Hardware and data mixture: Our diarization models are pre-trained on 32 H100 GPUs. TAC models use 128 H100 GPUs, except for the 122B-A10B model, which uses 128 GB200 GPUs. Each GPU processes 32k tokens per step, corresponding to 1M tokens per step for diarization and 4M for TAC. Continued pre-training lasts 100k steps. The captioning:diarization:text mixture is $1:1:1$ during pre-training and $3:1:1$ during SFT, measured by token count.

Training setup: Table 3 summarizes the pre-training configuration. Context parallelism of degree 8 distributes each sequence across eight GPUs, supporting contexts of up to 256k tokens. SFT uses the same training configuration with the $3:1:1$ data mixture above and a constant learning rate of 1×10^{-5} for two epochs over the retained SFT data.

[6] Corpus identifier: `allenai/dolma3_longmino_mix-100B-1125`.

Table 3: Pre-training configuration. The batch size shown is for TAC training on 128 GPUs; diarization uses 32 GPUs.

Setting	Value
Optimizer	AdamW
Pre-training steps	100k
Pre-training warmup	2k steps
Pre-training peak learning rate	2×10^{-5}
Pre-training learning-rate decay	Cosine decay to 1×10^{-5}
Tokens per GPU	32k
TAC batch size	32k $\times$ 128 tokens ($\approx$4M tokens/step)
Context parallelism	8
Maximum training context	256k tokens
Maximum pre-training audio duration	20 minutes

Inference failures: LongCap uses greedy decoding with a maximum of 8,192 output tokens per chunk. If a chunk reaches this limit without completing its caption, subsequent chunks are still decoded normally. For the 35B-A3B model, approximately 0.3–0.5% of chunks trigger a retry because of malformed captions, such as unparseable JSON or loops that repeatedly emit events. These failures become more frequent with smaller models and longer contexts, making the 0.8B and 2B variants impractical for reliable long-form use.

Retries and postprocessing: We use two measures to reduce these failures. First, when a malformed caption is detected, we regenerate the same chunk up to three times, using repetition penalties of 1.05, 1.10, and 1.15 for successive retries. Every retry still uses greedy decoding. Second, after decoding, we remove obviously repetitive duplicate events, such as a run of more than 100 events repeating the same transcript.

Caption-based audio understanding: On MMAU and MMAR, we evaluate how well LongCap's captions support question answering with a text LLM. LongCap first captions each recording; `gemini-3-flash-preview` (Google, 2026b) then answers the question using the caption and the supplied answer choices, without access to the audio. The reported accuracy measures this two-stage pipeline, rather than LongCap answering the questions directly. This follows the evaluation practice described in §4.1.

C PROMPTS

C.1 PROMPTS

Prompt 1: Seed structured annotation:

```
Listen to the audio. Output one JSON object (no markdown). Schema:

{
"timeline":[{"time_stamps":["<Xs>","<Ys>"],"type":"speech|music|sound_event",
    "speaker_id":"S1|null","transcript":"<text>|null",
    "attributes":{"<id> <attr>":"<val>",...},"caption":"<Chinese>"}]}

TIMELINE (select relevant per segment):
<attribute keys from Appendix A.1, selected by event type>
speech: [S]; music: [M]; sound_event: [E].

RULES:
1. Concurrent layers (music+speech) -> separate overlapping timeline
entries.
2. Speaker IDs S1,S2,... assigned consistently; null for non-speech.
3. transcript = exact spoken words for speech; null otherwise.
4. All values and captions in Chinese. Arrays for list values.
5. time_stamps as float seconds: "<X.Xs>","<Y.Ys>".
```

Prompt 2: Event-wise reannotation:

```
Listen to this short audio segment only. The segment type has already been
identified as "<event_type>". Produce LOCAL structured annotation for this
single segment only.

Output one JSON object (no markdown) with exactly this schema:
{
  "type": "<event_type>",
  "time_stamps": ["0.0s", "<duration to two decimals>s"],
  "speaker_id": "S1|null",
  "transcript": "<exact spoken words or lyrics>|null",
  "attributes": {"<id> <attr>": "<value>|null", ...},
  "caption": "<Chinese>"
}

Annotate only these attributes for this segment type. Include every key; use
null when the attribute is not present or cannot be inferred from the audio:
<all keys for event_type from Appendix A.1>

Existing segment text for context only. Re-listen to the audio and correct
it when the audio disagrees:
<existing event text, or "(empty)">

Rules:
1. Do not output timeline.
2. Keep the output type as "<event_type>".
3. speaker_id is only for speech; use null for music and sound_event.
4. transcript is exact spoken words for speech, exact lyrics for music if
any, and null for non-vocal music or sound_event.
5. All attribute values and captions must be in Chinese.
6. time_stamps are relative to this sliced segment, from 0.0s to its
duration.
```

D Additional Results

D.1 Additional Diarization Results

Table 4: Speech recognition and diarization results on each benchmark

Test set	MOSS	VibeVoice-ASR	Doubao ASR 2.0	Qwen3.8-Omni-Flash	LongCap-Diar (ours)
AliMeeting	**9.27/18.30** **15.49/15.68**	34.73/35.99 38.28/38.62	28.63/24.65 32.13/34.57	18.60/24.20 21.92/22.26	9.49/20.46 16.30/16.65
AISHELL-4	**5.80**/13.44 **13.15/13.43**	13.57/18.19 22.29/22.70	24.66/**12.59** 34.46/41.48	10.19/14.58 15.40/15.73	8.37/14.26 16.94/17.44
AMI-IHM	25.17/24.61 **20.92/21.58**	27.18/22.50 24.35/25.13	47.15/31.81 49.29/56.40	47.84/40.57 37.01/39.32	**21.98/21.03** 21.70/22.46
AMI-SDM	27.97/29.87 32.95/34.09	31.60/27.58 38.75/39.73	50.37/42.91 57.41/66.56	44.69/35.56 35.83/41.14	**15.46/22.73** **21.86/23.48**
LongCapBench-Diar-Short	11.41/5.98 17.34/18.84	16.96/6.28 28.24/29.48	17.00/**5.52** 25.49/27.51	23.18/12.31 31.10/35.48	**9.94**/5.89 **15.47/16.56**
LongCapBench-Diar-Long	27.11/27.00 31.77/32.63	15.45/14.23 20.35/25.69	18.05/10.74 18.54/20.73	29.65/23.52 28.38/43.72	**6.02/6.85** **13.13/14.06**
Macro mean	17.79/19.87 21.94/22.71	23.25/20.80 28.71/30.23	30.98/21.37 36.22/41.21	29.03/25.12 28.27/32.94	**11.88/15.20** **17.57/18.44**

Table 5: CHiME-6 results on sessions S01, S02, S09, and S21, micro-averaged across sessions. The test set has multiple near-field recordings for each example. We infer them all and keep the best results of each example. The maximum audio length is 2 hours 40 minutes. Most competitors are not practically usable on this application.

Model	DER	WER	cpWER	tcpWER
MOSS	99.81	99.81	99.81	99.97
VibeVoice-ASR	100.00	99.99	99.99	100.00
Doubao ASR 2.0	69.90	53.77	69.48	77.04
Qwen3.8-Omni-Flash	79.07	69.25	71.35	75.59
LongCap-Diar (ours)	**29.04**	**37.08**	**32.29**	**34.14**

Table 6: Comparison of 2B models with holistic processing and chunk-wise processing on LongCapBench-Diar-Long. All metrics are reported in %; lower is better. tcpWER uses a 5-second collar. Best values are bold.

Model	DER	WER	cpWER	tcpWER
holistic processing	109.13	80.59	115.87	147.16
chunk-wise processing	**20.30**	**12.74**	**29.01**	**34.80**

Extrapolation to 16 hours: With a 1M-token context window, LongCap-Diar processes 16 hours of concatenated audio. We average results over three recording-order permutations (Fig. 10). WER remains stable throughout, whereas DER, cpWER, and tcpWER increase because the speaker count in the concatenated audio far exceeds the range seen during training. This experiment is intended only to demonstrate long-duration processing.

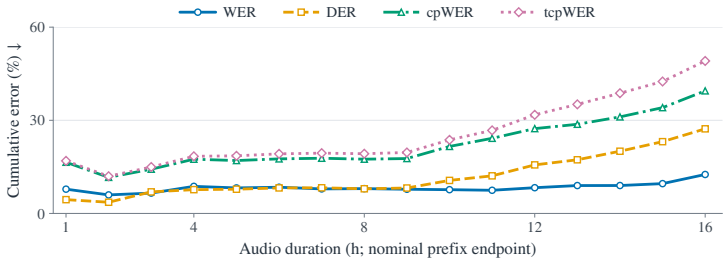


Figure 10: Cumulative error rates up to 16 hours, averaged over three recording orders.

D.2 Caption Content Comparison with AF-Next and LAT

We compare caption content on all 1,500 LongCapBench-TAC-Short clips (569 source recordings). Following the segment-level inference setup in Fig. 1, we also evaluate these models on LongCapBench-TAC-Long using non-overlapping 10-minute chunks (45 chunks from four recordings, including shorter tails). We use the main LongCap checkpoints, AF-Next Captioner (Ghosh et al., 2026), and LAT-Audio-Base (Shao et al., 2026). AF-Next receives the official captioning prompt: *"Generate a detailed caption for the input audio. In the caption, transcribe all spoken content by all speakers precisely."* LAT uses its released dense-captioning prompt: *"Provide a dense audio description for this audio segment."* We decode greedily and set a 16,384-token output limit for both baselines, using BF16 and repetition penalty 1.2 for AF-Next, and dynamic FP8 for LAT.

DeepSeek-V4-Pro jointly receives the reference and four anonymous candidates, returning preferences or ties for all six pairs. Structured references and LongCap outputs are deterministically rendered as field–value lists; native baseline text is unchanged. No LLM paraphrases inputs. The rubric balances salient content coverage against explicit reference contradictions, ignoring schema, numerical timestamps, exact transcript matching, and style. Unannotated details remain unknown. Two separate requests use a deterministic shuffled order and its reversal, without sharing judgments. Inputs are untruncated; the output/reasoning budget is 65,536 tokens.

For each LongCap–baseline pair, a win, tie, or loss receives 1, 0.5, or 0. We average both orders, then inputs; Short's five equally sized subsets and Long's chunks receive equal weight. Table 7 reports preference percentages; 50% denotes equal preference credit.

Short's 95% percentile intervals use 2,000 global source-cluster bootstrap draws, sharing each source's multiplicity across subsets. These intervals describe sensitivity to source composition. Long has only four source recordings. Reference incompleteness, recognizable styles, and single-judge bias remain; preferences do not measure timing or audio-verified accuracy.

Table 7: Tie-adjusted LongCap preference (%), with 95% source-cluster bootstrap intervals for Short.

LongCap model	Baseline	Short (95% interval)	Long (10-min chunks)
LongCap-35B-A3B	AF-Next	88.30 (86.80–89.63)	97.22
	LAT	98.97 (98.48–99.39)	97.78
LongCap-122B-A10B	AF-Next	87.75 (86.21–89.19)	97.22
	LAT	99.20 (98.78–99.54)	98.89

D.3 Length Expansion via Self-Distillation

Table 8: Before and after self-distillation on all 1,500 Short clips and all four full Long recordings. All values are percentages; arrows indicate preferred directions. Bold marks the better value within each model-size pair.

Model	Checkpoint	Short form			Full long form		
		TER ↓	GER ↓	tcpCapSim ↑	TER ↓	GER ↓	tcpCapSim ↑
35B	Before	13.01	**15.93**	**64.59**	44.84	11.46	50.85
	After	**12.42**	16.13	63.75	**41.09**	**10.83**	**51.24**
122B	Before	12.47	**15.51**	**64.48**	42.35	12.15	**51.67**
	After	**12.41**	15.94	63.84	**39.64**	**11.64**	51.19

D.4 Dependency Modeling Results

Setup: We evaluate LongCap-35B-A3B and LongCap-122B-A10B with event-dependency supervision on LongCapBench-TAC-Short and LongCapBench-TAC-Long. DeepSeek and Gemini independently annotate dependencies on the benchmark reference events. These text-derived silver graphs are not audio-verified ground truth. A separate DeepSeek matcher aligns local captions and timestamps with positive temporal overlap, hiding IDs, ancestors, and relations; the same event correspondence serves both references. Native events and arcs are retained without rewriting; an arc must resolve to a unique earlier declaration of its typed ID. Excluding predictions with negative timestamps leaves 1,492/1,495 short-form clips for 35B/122B. Reference-event controls cover all 1,500 clips; every condition includes all four long-form recordings in full. Empty predictions remain in scoring.

Metrics: *Link support* (LS) averages the reference-supported fraction of each emitted arc's endpoint coverage over all three relation types (`same_entity`, `continues`, and `recurs`). Support may come from another earlier member of the same typed reference group; invalid or ambiguous addresses and unknown relations receive zero. *Connectivity* F_1 here measures direct and transitive `same_entity` agreement using interval-pair weights $w_{ij} = d_i d_j$, where d_i, d_j are interval durations. Pairs within the same original component on both sides are excluded. With predicted and reference connected masses W_H, W_R and supported intersection M,

$$F_1 = \frac{2M}{W_H + W_R}. \tag{3}$$

Only resolved reference paths supply positive credit; unresolved paths remain in the denominators, and missing matching judgments remain uncertainty. Scores macro-average defined per-record values: LS is undefined without emitted arcs; F_1 is undefined without positives on either side and zero with positives on only one side. Eligible cohorts therefore differ. Table 9 separates the references; Fig. 9 averages their two macro scores equally.

Results and GLM controls: Averaging the two references, the models' original long-form F_1 increases from 4.2% (35B) to 14.8% (122B), while both remain well below their short-form scores (63.4/68.1%). For controls, GLM-5.3-Flash labels benchmark reference events and each model's predicted events using the same labeling procedure. For predictions, we replace the original links while preserving captions, timestamps, duplicate events, and the evaluated clips. Chronological sorting and unique working IDs also remove ambiguity in event identifiers. Relabeling raises long-form F_1 to 17.4/22.3% (35B/122B), combining new links with address normalization; these gains do not isolate learned dependency semantics or a model-size effect. GLM on reference events reaches 70.4% on short-form and 19.7% on long-form audio, so low long-form agreement persists even with reference events. This is an agreement baseline, not a performance ceiling. The four-recording sample, incomplete matching, and ambiguous native IDs limit interpretation as long-range memory accuracy.

Table 9: Dependency agreement (%, higher is better). Native denotes the model's original links; +GLM denotes GLM-5.3-Flash links on the indicated events. LS covers all relations; F_1 covers `same_entity`.

Benchmark	Condition	LS		F_1	
		DeepSeek	Gemini	DeepSeek	Gemini
Short form	35B native	54.40	55.71	62.34	64.39
	122B native	58.54	59.95	67.20	68.98
	Reference + GLM	70.27	72.29	69.60	71.26
	35B + GLM	57.38	58.83	66.16	68.06
	122B + GLM	58.16	59.56	66.43	68.12
Long form	35B native	9.30	9.73	4.99	3.39
	122B native	22.86	24.66	16.47	13.06
	Reference + GLM	49.56	52.62	22.87	16.44
	35B + GLM	32.13	35.00	18.35	16.43
	122B + GLM	35.44	38.84	24.09	20.46