AUDIOTETRIS: TOKEN ELIMINATION VIA TRANSITIONAL IMPORTANCE SIGNALS FOR EFFICIENT AUDIO LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Audio Language Models (LALMs) achieve strong performance on speech-text tasks, but standard audio encoders produce hundreds of tokens for even short audio clips. Because self-attention scales quadratically in sequence length, this inflates inference-time compute substantially. Current training-free token compression methods prune or merge tokens based solely on coarse attention or feature similarity. However, treating audio tokens as generic, isotropic features smooths over spectral transitions, destroying fine-grained phonetic transitions essential for decoding speech. In contrast, speech signals exhibit highly structured temporal patterns where information density peaks at acoustic transitions. We propose **AudioTETRIS**, a training-free, two-stage framework that uses the audio encoder's representation geometry as an acoustic prior for token importance before applying attention-based compression. Instead of discarding encoder structure after the encoding stage, AudioTETRIS propagates this prior directly into the LLM decoder to compute a per-token fused score. This score drives progressive top-k token selection across decoder layers without requiring additional training or auxiliary models. Evaluated on Qwen2-Audio-7B, Phi-4-Multimodal, and Audio Flamingo 3, AudioTETRIS retains near-uncompressed performance on several speech tasks using only 25% of audio tokens, matches or surpasses full-token accuracy at moderate compression rates, and consistently outperforms existing training-free baselines.

1 INTRODUCTION

Large Audio Language Models (LALMs) have achieved state-of-the-art results across several speech-text tasks like speech recognition, audio QA, and general audio understanding (Chu et al., 2024; Radford et al., 2023; Chu et al., 2023; Wang et al., 2025a; Yang et al., 2024). Typically, these models use a pretrained audio encoder to map the input waveform into a sequence of continuous frame-level embeddings (audio tokens), which are projected into the Large Language Model (LLM) embedding space and prepended to text prompts. However, this approach creates a severe computational bottleneck: standard speech encoders generate hundreds of tokens for just a few seconds of audio (Lin et al., 2025; Chu et al., 2024; Ghosh et al., 2025a). Because decoder self-attention scales quadratically with sequence length, this large token footprint drives up inference costs (Tay et al., 2022).

Furthermore, uniform audio tokenization is inherently redundant (Radford et al., 2023; Chu et al., 2024; Ghosh et al., 2025a; Wang et al., 2025b; Zhang et al., 2026). It clashes with the nature of speech, where information is concentrated during rapid phonetic transitions rather than steady-state regions (Furui, 1986a; Stevens, 2002; Aylett & Turk, 2004; Zhang et al., 2025a). Reducing token count without losing these acoustically rich transition regions is vital for efficient LALM deployment. While recent methods apply token pruning or merging (Lin et al., 2025; Bhati et al., 2025; Gong et al., 2026; Behera et al., 2024), they share a substantial flaw: they rely solely on isolated decoder attention or local feature similarity, ignoring the structural phoneme transitions that

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.

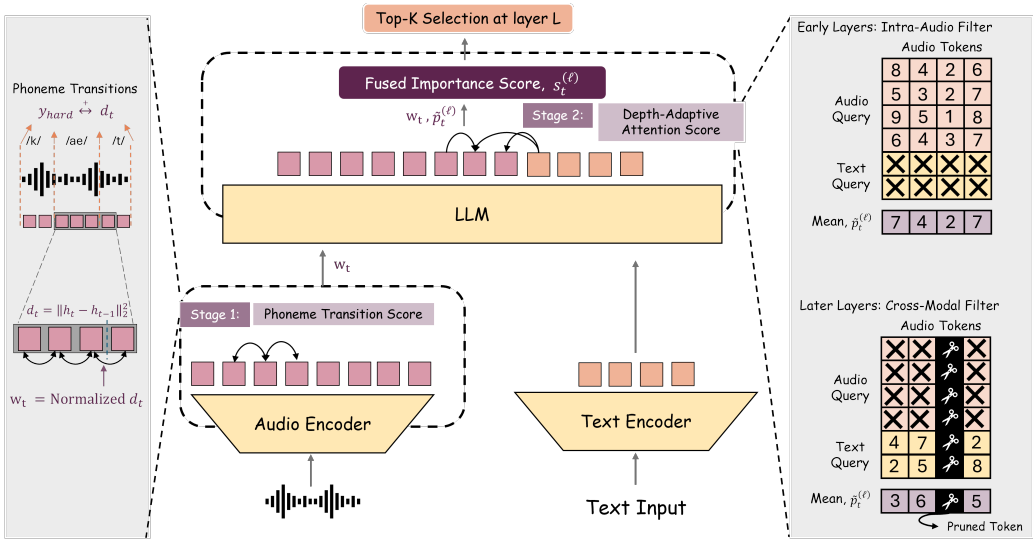


Figure 1: **Overview of AudioTETRIS.** A phoneme-transition-aware acoustic prior w_t is computed from the audio encoder's hidden states via the transitional distance $d_t = \|h_t - h_{t-1}\|_2^2$, preserving phoneme transitions y_{hard} (§4; left). Token pruning occurs in two stages: **Stage 1** injects w_t as a static acoustic prior from the encoder's hidden states; **Stage 2** computes a depth-adaptive LLM attention score, fusing intra-audio attention in early layers with cross-modal (text-to-audio) attention in later layers (right). The two scores are fused into a single importance score, and top-k selection retains the most salient tokens at each pruning layer ℓ.

could be already captured by the encoder (Behera et al., 2024; Lin et al., 2025; Gong et al., 2026; Jung et al., 2026; Bhati et al., 2025).

We hypothesize that speech token compression should be guided by the underlying acoustic structure of the signal, rather than post-hoc decoder attention alone. Our approach is based on a simple insight: regions of rapid spectral transition in the encoder's hidden states coincide with phonetic transitions, where essential acoustic details reside (Furui, 1986a; Stevens, 2002). We empirically validate this hypothesis in Section 4, showing a strong correlation between representation variance in encoder space and actual phonetic segment transitions. Compression methods that ignore this structure could risk discarding high-frequency acoustic details and degrading downstream performance (Zhang et al., 2026; Libera et al., 2026).

To solve this, we introduce **AudioTETRIS** (Figure 1), a training-free framework that compresses audio tokens by jointly exploiting speech-specific acoustic structures and decoder-side attention. First, AudioTETRIS derives a phoneme-transition-aware prior from the temporal derivative of hidden states, isolating regions of high spectral change to protect information-dense transition segments. Second, this prior is propagated into the LLM decoder and fused with a depth-adaptive attention score derived from intra-audio attention in early layers and cross-modal text-to-audio attention in middle layers. By coupling acoustic salience with task-conditioned attention, token survival is driven by both structural and semantic relevance.

We evaluate AudioTETRIS across backbones like Qwen2-Audio-7B, Phi-4-Multimodal, and Audio Flamingo 3 (Chu et al., 2024; Microsoft et al., 2025; Ghosh et al., 2025a) on tasks including Automatic Speech Recognition (ASR), Speech Emotion Recognition, Audio Question Answering, and Speech Translation. AudioTETRIS retains near-lossless accuracy while keeping as few as 25% of tokens, outperforming other training-free baselines, and even surpassing full-sequence models at moderate compression rates.

Our main contributions are:

- **Acoustic transition prior:** We show that temporal variance in encoder hidden states serves as a reliable proxy for phonetic transitions, capturing high-information regions without fine-tuning or external alignments.

- **AudioTETRIS framework:** We propose a training-free compression framework that propagates this acoustic prior into the LLM decoder, dynamically guiding self-attention to preserve both acoustic and semantic details.
- **State-of-the-art efficiency:** Extensive experiments demonstrate near-lossless performance at aggressive compression levels (25% token retention), outperforming strong training-free baselines across diverse benchmarks.

2 Related Work

Large Audio Language Models. LALMs typically couple a pretrained speech encoder, such as Whisper (Radford et al., 2023), with an LLM backbone (Chu et al., 2024; Ghosh et al., 2025b;a; Microsoft et al., 2025; Tang et al., 2024; Deshmukh et al., 2023), projecting audio features into the text embedding space. However, this paradigm yields audio token sequences that are orders of magnitude longer than the accompanying text, creating severe computational overhead during prefill and generation. Reducing this token count without sacrificing the dense lexical content of the underlying signal remains a key challenge (Shao et al., 2026; He et al., 2025).

Training-Free Token Compression. Training-free compression methods eliminate redundant audio tokens without modifying model weights. One direction seeks to prune or merge sequence tokens prior to or during LLM processing (Bolya et al., 2023; Behera et al., 2024; Chen et al., 2024; Jung et al., 2026). Across vision-language models (VLMs), methods like SparseVLM (Zhang et al., 2025b) leverage self-attention logits to select text tokens as 'raters' that adaptively prune visual tokens across layers (see also Shang et al., 2025; Yang et al., 2025; Xing et al., 2025). In the speech domain, SpeechPrune (Lin et al., 2025) compresses long-form speech ($\geq$30s) tokens using approximated attention scores from the LLM's initial transformer layer alongside speech-text feature similarity. However, relying on early-layer LLM dynamics forces the model to evaluate query relevance before building a comprehensive contextual representation of the signal. Alternatively, encoder-level methods like Local Temporal Bipartite Merging (LTBM) (Luo et al., 2026) merge nearby tokens under a temporal window constraint (Li et al., 2023), relying purely on local feature similarity rather than task-specific or phonetic cues.

Training-Based & Multimodal Token Compression. Rather than relying on training-free pruning, alternative strategies preserve performance by introducing extra trainable parameters or secondary modalities (Kavaki & Mandel, 2025). For instance, encoder-level pooling methods often require low-rank adapters (LoRA) (Hu et al., 2022) to fine-tune the backbone and recover downstream accuracy (Bhati et al., 2025). Although effective, this training dependency eliminates the module's utility as a plug-and-play solution for unseen backbones or domain shifts. Other frameworks, such as EchoingPixels (Gong et al., 2026), achieve substantial token reduction by exploiting cross-modal alignment across visual and audio streams, but do not extend naturally to audio-language tasks. Across both directions, training-free methods bypass acoustic structure while structure-aware methods require fine-tuning or secondary modalities - a trade-off AudioTETRIS resolves with a training-free, acoustically grounded importance signal (§3).

3 Methodology

We present **AudioTETRIS**, a training-free framework for compressing audio tokens in LALMs. AudioTETRIS operates in two stages: Stage 1 (§3.2) extracts a phoneme-transition-aware acoustic prior directly from the audio encoder's hidden states, and Stage 2 (§3.3) propagates this prior into the LLM decoder, fusing it with a depth-adaptive attention score to progressively prune audio tokens across decoder layers. Figure 1 illustrates the pipeline.

3.1 Preliminaries

Given an input waveform, the audio encoder outputs a sequence of hidden states $\{h_1, \ldots, h_T\} \in \mathbb{R}^{T \times d}$ representing T audio tokens. These tokens are projected into the LLM embedding space and concatenated with the query embeddings to yield the initial decoder sequence $X = [x_1^a, \ldots, x_T^a, x_1^q, \ldots, x_M^q]$, with a and q referring to audio and query tokens. We index decoder layers as $\ell = 0, \ldots, L_{\text{dec}} - 1$, where L_{dec} is the number of decoder layers, and denote the layer-ℓ

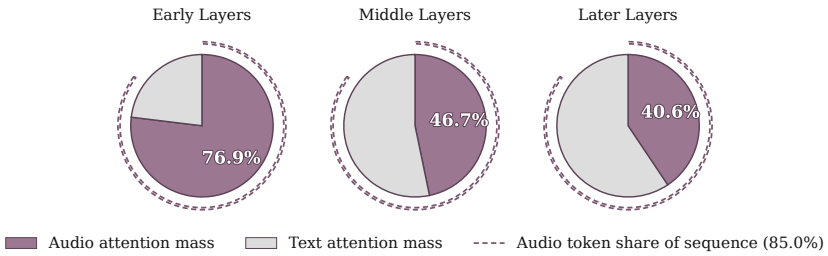


Figure 2: Distribution of decoder attention mass allocated to audio tokens across model depth in Qwen2-Audio on LibriSpeech. Despite audio tokens comprising 84.98% of the input sequence (dashed ring), early layers are dominated by intra-audio attention (76.91%), which drops sharply in middle (46.73%) and late layers (40.60%) as attention shifts to text-conditioned representations. This shift motivates our layer-dependent pooling rule in Eq. 3.

self-attention weight matrix as $A^{(\ell)} \in \mathbb{R}^{H \times N^{(\ell)} \times N^{(\ell)}}$, where H is the number of heads and $N^{(\ell)}$ is the sequence length at layer ℓ (which shrinks as audio tokens are pruned).

3.2 Stage 1: Phoneme Transition Score

Motivation. Linguistic information in speech is concentrated near acoustic-phonetic transitions (Furui, 1986b). Because these transitions carry the most acoustic and phonetic information, we hypothesize that tokens spanning them undergo significantly larger representational shifts in latent space than tokens within stable phonetic segments. We leverage this geometric property to construct an acoustic gate that identifies informative tokens without external supervision.

Phonetic Transition Score Formulation. For each audio token t, we compute the token-to-token derivative d_t as the squared Euclidean distance between consecutive hidden states $h_t, h_{t-1} \in \mathbb{R}^d$ taken from a single intermediate encoder layer (layer selection in §5.1):

$$d_t = \|h_t - h_{t-1}\|_2^2, \qquad d_1 := 0, \tag{1}$$

where the first token is assigned zero derivative in the absence of a preceding token. The squared L_2 norm amplifies sharp representational shifts over intra-segment noise at a linear computational cost of $\mathcal{O}(d)$ per token.

To modulate attention during pruning, we convert d_t into a normalized acoustic weight w_t:

$$w_t = \frac{d_t}{\frac{1}{T}\sum_{i=1}^{T} d_i}, \tag{2}$$

where T is the total number of audio tokens in the utterance. The denominator enforces unit mean ($\mathbb{E}_t[w_t] = 1$).

3.3 Stage 2: Depth-Adaptive Attention Scoring

Motivation. Attention mass allocation evolves significantly across LLM depth (Figure 2). Early decoder layers are dominated by intra-audio attention (audio tokens attending to audio tokens) to refine structural audio representations. In contrast, middle and late layers reallocate attention toward cross-modal interaction, where text queries attend directly to audio tokens. Consequently, a single, depth-invariant attention metric fails to capture token importance uniformly across the network.

Layer-Adaptive Scoring. We address this depth dependence by defining the attention score $\tilde{p}_t^{(\ell)}$ for audio token t at layer ℓ over a depth-specific subset of query rows:

$$\tilde{p}_t^{(\ell)} = \begin{cases} \dfrac{1}{H \cdot |\mathcal{Q}_a|} \displaystyle\sum_{h=1}^{H} \sum_{i \in \mathcal{Q}_a} A_{h,i,t}^{(\ell)}, & \ell \in \mathcal{L}_{\text{early}} \quad \text{(intra-audio filter)} \\ \dfrac{1}{H \cdot |\mathcal{Q}_q|} \displaystyle\sum_{h=1}^{H} \sum_{i \in \mathcal{Q}_q} A_{h,i,t}^{(\ell)}, & \ell \in \mathcal{L}_{\text{later}} \quad \text{(cross-modal filter)} \end{cases} \tag{3}$$

where $\mathcal{Q}_a$ and $\mathcal{Q}_q$ denote the index sets of audio-query and text-query rows, H is the number of attention heads, and $\{\mathcal{L}_{\text{early}}, \mathcal{L}_{\text{later}}\}$ partition the decoder layers.

Automated Crossover Discovery. Rather than hand-crafting the layer partition $\{\mathcal{L}_{\text{early}}, \mathcal{L}_{\text{later}}\}$, we determine the crossover layer $\ell^\dagger$ from backbone attention dynamics on a small calibration set, as the first depth at which mean cross-modal attention mass $a_{\text{cross}}(\ell)$ overtakes mean intra-audio attention mass $a_{\text{intra}}(\ell)$:

$$\ell^\dagger = \min\{\ell : a_{\text{cross}}(\ell) \geq a_{\text{intra}}(\ell)\}, \tag{4}$$

This unsupervised procedure identifies $\ell^\dagger$ in a single calibration pass per backbone; full derivation and per-layer statistics are in Appendix A.2.2.

Score Fusion. At each pruning layer ℓ, we fuse the layer-adaptive attention score $\tilde{p}_t^{(\ell)}$ with the static acoustic gate w_t:

$$s_t^{(\ell)} = \tilde{p}_t^{(\ell)} \cdot w_t. \tag{5}$$

At each pruning layer ℓ, we rank tokens by $s_t^{(\ell)}$ and retain a top fraction k_ℓ, discarding the rest (see §5.1 for the pruning schedule and how k_ℓ is set per compression rate).

4 Analysis: Validation of Acoustic State Dynamics

In Section 3.2, we hypothesized that token-to-token encoder state transitions d_t track actual phonetic transitions. Here, we validate this hypothesis both quantitatively against ground-truth forced alignments and label-free via distributional kurtosis.

Phoneme Transition Correlation. To evaluate whether encoder distance spikes align with physical phoneme transitions, we define a continuous diagnostic transition prior b_t. Because audio encoders distribute context across neighboring tokens (Radford et al., 2023), we convolve d_t with a Gaussian kernel of width $\sigma = 2$ and clip to $[0, 1]$:

$$b_t = \text{clip}\Big(\textstyle\sum_\tau d_\tau \cdot \mathcal{N}(t - \tau;\, 0,\, \sigma^2),\; 0,\; 1\Big). \tag{6}$$

We evaluate b_t against forced alignments on LibriSpeech (Panayotov et al., 2015; CorentinJ, 2019). Standard alignment tools yield discrete phonetic transitions y_{hard}. To model continuous speech coarticulation (Furui, 1986b), we apply identical Gaussian smoothing to y_{hard}, producing a continuous target y_{soft}. We measure the Pearson correlations $r_{\text{hard}} = \text{corr}(b_t, y_{\text{hard}})$ and $r_{\text{soft}} = \text{corr}(b_t, y_{\text{soft}})$. Figure 8 in Appendix A.2.3 illustrates this end-to-end pipeline. We provide additional dataset validations in Appendix A.2.4.

As shown in Figure 3 (top), transition alignment varies predictably across depth, peaking in mid-network layers across all backbones: Qwen2-Audio (layer 16, $r_{\text{soft}} = 0.68$) (Chu et al., 2024), Phi-4-Multimodal (layer 14, $r_{\text{soft}} = 0.59$) (Microsoft et al., 2025), and Audio Flamingo 3 (layer 15, $r_{\text{soft}} = 0.62$) (Ghosh et al., 2025a). Appendix A.2.5 provides an ablation over σ.

Label-Free Diagnostic via Kurtosis. To verify this mid-network correlation peak without relying on external alignments, we measure the distributional kurtosis (κ) of d_t across layer depth. High kurtosis indicates that transitions stand out as sharp, heavy-tailed outliers above background intra-segment noise. As shown in Figure 3 (bottom), kurtosis peaks at the exact same mid-network layers ($\kappa \approx 24.0$-26.5), providing an unsupervised method to verify transition isolation.

Trajectory Visualization. A qualitative t-SNE analysis of per-token hidden states (Appendix A.2.6, Figure 11) corroborates this pattern: mid-network layers show tight phoneme clustering separated by sudden trajectory jumps at transitions, while early and late layers show progressively less phonetically consolidated structure.

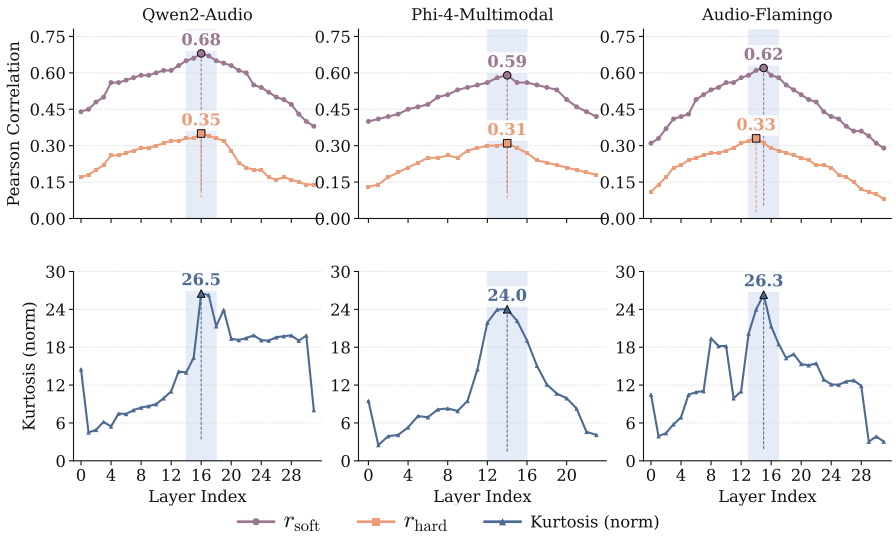


Figure 3: Layer-wise diagnostics across Qwen2-Audio, Phi-4-Multimodal, and Audio Flamingo 3. **Top:** Mean correlation between b_t and ground-truth phoneme transitions (r_{soft} and r_{hard}). **Bottom:** Kurtosis (κ) of the distance distribution d_t. Both metrics identify an intrinsic mid-network sweet spot (shaded) where phonetic transitions are maximally accentuated.

5 Experiments & Results

5.1 Experimental Setup

Backbones and Datasets. We evaluate AudioTETRIS on three LALM backbones with distinct encoder-decoder architectures: Qwen2-Audio-7B (∼8.4B) (Chu et al., 2024), Phi-4-Multimodal (5.6B) (Microsoft et al., 2025), and Audio Flamingo 3 (∼8.3B) (Ghosh et al., 2025a). Our evaluation spans four core tasks: automatic speech recognition on LibriSpeech test-other (Panayotov et al., 2015), emotion recognition on IEMOCAP (Busso et al., 2008), audio question answering on MMAU-mini (Sakshi et al., 2025), and speech translation (English-to-Chinese) on CoVoST 2 (Wang et al., 2021). Together, these benchmarks span lexical transcription, cross-lingual generation, paralinguistic classification, and complex audio reasoning, allowing us to test whether preserving phoneme-transition regions specifically benefits tasks that require precise lexical recovery.

Baselines. We compare against SpeechPrune (Lin et al., 2025), SparseVLM (Zhang et al., 2025b), and LTBM (Luo et al., 2026), representing the two families of prior work discussed in §2. We additionally include a Random Pruning baseline that retains a uniformly random token subset at matched compression rates, providing a lower-bound sanity check. All baselines are implemented under matched compression schedules using released hyperparameters where available.

Implementation Details. We evaluate four target token compression rates - $0.36, 0.51, 0.64$, and 0.75 - representing the fraction of audio tokens removed by the final decoder layer. These correspond to uniform per-stage retention factors $r \in \{0.8, 0.7, 0.6, 0.5\}$ across two pruning layers ($1 - r^2$) in our experiments, ranging from light reduction to aggressive $4\times$ compression. Pruning is scheduled at layer 1[1] and at a backbone-specific later layer $\ell^\dagger$ (Eq. 4) via progressive top-k retention on the fused score $s_t^{(\ell)}$ (Eq. 5). The acoustic prior w_t is computed from a single mid-network encoder layer per backbone: layer 16 for Qwen2-Audio, layer 14 for Phi-4-Multimodal, and layer 15 for Audio Flamingo 3. These layers are selected by the label-free kurtosis criterion of §4 (the depth maximizing κ of d_t); forced alignments are used only to validate this choice post hoc and are not required to apply AudioTETRIS to a new backbone. The diagnostic prior b_t is computed from the same layer. All experiments are executed on a single NVIDIA A100 GPU. Layer choices, calibration, and alternative schedules are detailed in Appendices A.2.2 and A.3.3.

[1]Decoder layers are indexed starting from 0 throughout this paper (matching the axis convention used in Figures 2 and 7); e.g., 'layer 1' denotes the second decoder layer.

5.2 Main Results

We report overall performance and efficiency across four compression rates on Qwen2-Audio in Table 1. Three key observations emerge from these results:

AudioTETRIS maintains high accuracy even under heavy compression. At low compression (rate 0.36), AudioTETRIS matches or improves upon the uncompressed baseline on many benchmarks (4.52 WER, 58.04% IEMOCAP, 42.36% MMAU-mini vs. 4.61, 55.11%, 42.20% uncompressed). Even at rate 0.75, WER stays at 8.50 while IEMOCAP, MMAU-mini, and CoVoST2 remain within 2 points of uncompressed baselines, confirming that our method removes redundant tokens without disrupting essential lexical and semantic transitions.

Performance gaps widen as compression grows aggressive. At rate 0.36, several baselines remain competitive; by rate 0.75, baseline methods degrade sharply (random pruning WER climbs to 31.59), lacking a mechanism to safeguard acoustic transitions. AudioTETRIS outperforms the best baseline by over 1.2 WER points at the highest compression setting.

ASR is inherently more vulnerable to token pruning. Fine-grained transcription (WER) degrades far faster under aggressive compression than high-level understanding (MMAU-mini) or emotion recognition (IEMOCAP): at rate 0.75, IEMOCAP and MMAU-mini stay nearly intact while WER roughly doubles. High-level tasks rely on coarse acoustic features and tolerate token dropout, whereas exact lexical recovery needs fine-grained transition preservation - a vulnerability that speech-specific methods (AudioTETRIS, SpeechPrune) mitigate far better than general-purpose vision baselines or non-speech heuristics.

5.3 Efficiency Analysis

Pruning within the LLM backbone yields direct computational savings, dropping total TFLOPs from 7.89 to 4.32 at a 0.75 compression rate, with latency falling from 1.31 s to 0.62 s over the same range. In-LM methods (AudioTETRIS, SparseVLM) incur slightly higher TFLOPs than single-shot methods (SpeechPrune, LTBM) since early layers still process the full, unpruned sequence before the first pruning stage; this difference is minimal relative to AudioTETRIS's accuracy gains. Real-world latency scales somewhat less than the FLOP drop due to RoPE reindexing and KV-cache trimming overhead at intermediate layers (Appendix A.1.4).

5.4 Generalization Across Backbones

To test whether AudioTETRIS's gains are specific to Qwen2-Audio's architecture, we repeat the main comparison on Phi-4-Multimodal and Audio Flamingo 3 at compression ratio 0.51 (Figure 4). On both backbones, AudioTETRIS remains within 1 point of uncompressed WER and attains the strongest or near-strongest accuracy on MMAU-mini, while baselines that match it on one metric (e.g., WER) fall sharply behind on others (e.g., SpeechPrune loses over 17 BLEU points on CoVoST2); Full numbers are given in Table 4 (Phi-4-Multimodal) and Table 5 (Audio Flamingo 3).

These results suggest the phonetic transition prior transfers across encoders with different subsampling rates and architectures (Phi-4-Multimodal's Conformer encoder vs. the Whisper-based encoders of Qwen2-Audio and Audio Flamingo 3), consistent with encoder-state transitions being a general property of speech representations rather than an artifact of one backbone. Extended results across all four compression ratios and benchmarks are in Appendix A.3.1, where similar patterns hold.

6 Ablation Studies

We isolate the contribution of each component of AudioTETRIS's scoring function and validate the choice of distance metric underlying the transition score §3.2.

6.1 Distance Metric for the Transition Score

The transition signal d_t requires a distance function to measure the transition magnitude between adjacent encoder hidden states. We compare four standard choices: Euclidean (L2), cosine, Manhat-

Table 1: Performance and efficiency comparison across compression ratios on Qwen2Audio evaluated on LibriSpeech (ASR), IEMOCAP (Emotion Recognition), MMAU-mini (Audio Understanding, MCQ), and CoVoST2 (Speech Translation). WER (%) is lower-is-better (↓); Accuracy (%) and BLEU are higher-is-better (↑). Baselines: SpeechPrune (Lin et al., 2025), SparseVLM (Zhang et al., 2025b), LTBM (Luo et al., 2026).

Method	LibriSpeech WER ↓	IEMOCAP Acc. ↑	MMAU-mini Acc. ↑	CoVoST2 BLEU ↑	Latency (s)	TFLOPs
No Compression	$4.61^{\pm 0.12}$	$55.11^{\pm 1.06}$	$42.20^{\pm 0.96}$	$33.03^{\pm 0.72}$	1.31	7.89
Compression Ratio 0.36						
Random	$9.91^{\pm 1.67}$	$53.10^{\pm 0.37}$	$40.18^{\pm 0.23}$	$29.10^{\pm 0.02}$	1.15	6.93
SpeechPrune	$6.75^{\pm 1.44}$	$56.43^{\pm 0.35}$	$41.17^{\pm 0.60}$	$31.34^{\pm 0.07}$	1.21	6.99
SparseVLM†‡	$8.40^{\pm 1.08}$	$52.37^{\pm 0.21}$	$41.09^{\pm 0.73}$	$31.38^{\pm 0.12}$	1.16	7.28
LTBM	$5.90^{\pm 0.48}$	$55.90^{\pm 0.27}$	$40.67^{\pm 0.67}$	$29.97^{\pm 0.06}$	1.25	6.93
AudioTETRIS†	$4.52^{\pm 0.21}$	$58.04^{\pm 0.84}$	$42.36^{\pm 0.21}$	$31.96^{\pm 0.04}$	1.19	7.33
Compression Ratio 0.51						
Random	$14.02^{\pm 0.33}$	$52.08^{\pm 0.49}$	$39.20^{\pm 0.43}$	$26.95^{\pm 0.01}$	0.97	5.61
SpeechPrune	$6.52^{\pm 1.14}$	$57.39^{\pm 0.74}$	$41.83^{\pm 0.06}$	$31.16^{\pm 0.15}$	1.14	5.67
SparseVLM†‡	$9.13^{\pm 1.15}$	$47.10^{\pm 0.69}$	$38.80^{\pm 0.27}$	$31.17^{\pm 0.09}$	1.15	6.10
LTBM	$6.92^{\pm 0.04}$	$54.58^{\pm 0.42}$	$40.33^{\pm 0.40}$	$29.55^{\pm 0.03}$	1.15	5.62
AudioTETRIS†	$5.77^{\pm 0.33}$	$57.03^{\pm 0.41}$	$42.91^{\pm 0.36}$	$31.45^{\pm 0.06}$	1.00	6.13
Compression Ratio 0.64						
Random	$24.08^{\pm 0.26}$	$50.22^{\pm 1.10}$	$39.23^{\pm 0.41}$	$24.64^{\pm 0.02}$	0.84	4.47
SpeechPrune	$8.62^{\pm 0.07}$	$56.10^{\pm 1.09}$	$40.30^{\pm 0.69}$	$30.12^{\pm 0.05}$	0.99	4.53
SparseVLM†‡	$12.40^{\pm 0.55}$	$47.27^{\pm 1.74}$	$37.66^{\pm 0.66}$	$30.79^{\pm 0.12}$	0.95	5.08
LTBM	$8.60^{\pm 0.18}$	$53.38^{\pm 0.86}$	$40.70^{\pm 0.78}$	$29.47^{\pm 0.06}$	1.03	4.49
AudioTETRIS†	$6.62^{\pm 0.29}$	$57.90^{\pm 0.76}$	$41.83^{\pm 0.15}$	$31.24^{\pm 0.11}$	0.86	5.26
Compression Ratio 0.75						
Random	$31.59^{\pm 0.24}$	$47.67^{\pm 0.71}$	$38.57^{\pm 1.16}$	$22.82^{\pm 0.01}$	0.61	3.52
SpeechPrune	$9.75^{\pm 0.10}$	$54.17^{\pm 0.92}$	$40.93^{\pm 0.34}$	$25.77^{\pm 0.05}$	0.69	3.58
SparseVLM†‡	$17.30^{\pm 0.70}$	$46.10^{\pm 3.05}$	$35.83^{\pm 0.12}$	$30.66^{\pm 0.02}$	0.71	4.23
LTBM	$10.58^{\pm 0.42}$	$47.84^{\pm 1.05}$	$39.23^{\pm 0.45}$	$24.26^{\pm 0.05}$	0.70	3.52
AudioTETRIS†	$8.50^{\pm 0.18}$	$55.34^{\pm 1.25}$	$41.46^{\pm 0.32}$	$31.33^{\pm 0.07}$	0.62	4.32

† In-LM pruning (token elimination performed inside the LLM backbone).
‡ Originally developed for the vision modality, adapted here for speech/audio.

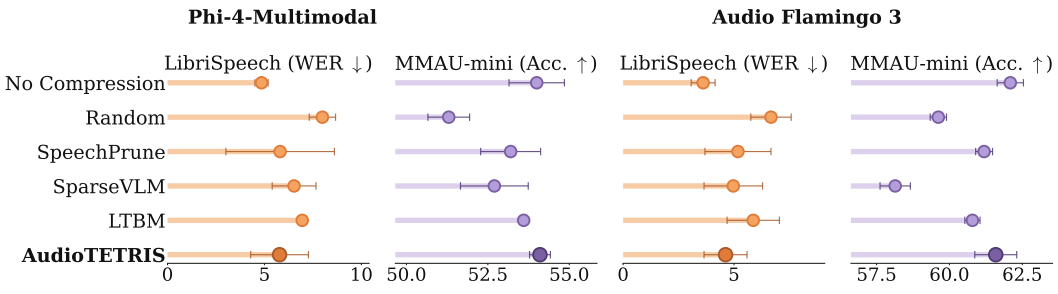


Figure 4: Comparison of LibriSpeech WER (↓) and MMAU-mini accuracy (↑) at compression ratio 0.51, for Phi-4-Multimodal and Audio Flamingo 3. Error bars show standard deviation across runs. AUDIOTETRIS is highlighted in the darker shade.

tan (L1), and Chebyshev (L_∞) distance, holding all other hyperparameters fixed at a compression ratio of 0.51. Results are summarized in Figure 5.

We find that L2 distance is the strongest choice on all four benchmarks, with the largest gap on ASR (cosine increases WER by over 50%: 9.04% vs. 5.77%); hence we use it as the default throughout.

6.2 Component and Stage Ablation Analysis

The fused score $s_t^{(\ell)} = \tilde{p}_t^{(\ell)} \cdot w_t$ (Eq. 5) combines the depth-adaptive attention term with the acoustic prior. Figure 6 isolates each stage's contribution against the full pipeline at compression ratio 0.51; extended results across all ratios and remaining benchmarks are in Table 6 (Appendix A.3.2).

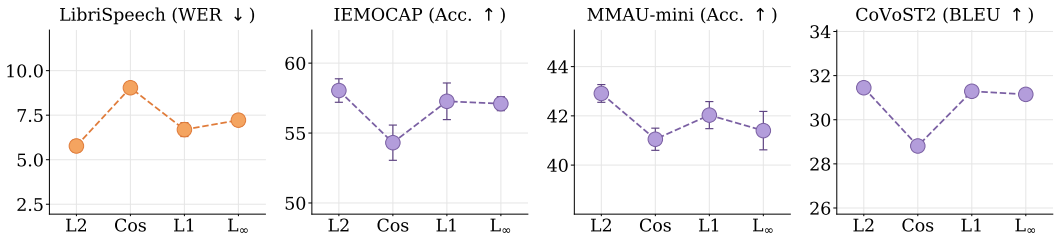


Figure 5: WER/accuracy/BLEU across distance metrics for the transition score, at compression ratio 0.51. Error bars show standard deviation over runs.

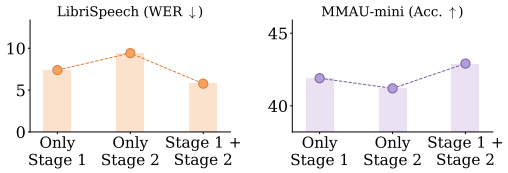


Figure 6: Ablation of AudioTETRIS stages at compression ratio 0.51.

We find that both stages combined consistently outperforms either stage alone across all ratios and benchmarks, with the gap widening sharply under aggressive compression (at ratio 0.75, Stage 2 alone degrades to 15.56% WER vs. 8.50% combined). This reflects complementary signals: the phonetic prior w_t preserves phonetically contiguous spans, while the attention term targets task relevance - most visible on LibriSpeech, where ASR's sensitivity to fine-grained transitions makes the two-stage advantage largest.

7 Conclusion

We presented AudioTETRIS, a training-free framework for compressing audio tokens in LALMs. By leveraging the audio encoder's hidden-state dynamics as a phoneme-transition prior and fusing it with depth-adaptive decoder attention, AudioTETRIS preserves acoustically rich and semantically critical regions without requiring auxiliary training or fine-tuning. Experiments across Qwen2-Audio, Phi-4-Multimodal, and Audio Flamingo 3, demonstrate that AudioTETRIS retains near-lossless performance at light compression, while widening its advantage over training-free baselines as compression gets aggressive. Furthermore, our findings highlight that ASR is highly sensitive to token loss making speech-aware structural priors essential for exact lexical recovery. Ultimately, this work highlights a broader principle for multimodal architecture design and token-compression methods: intermediate representations computed by modality-specific encoders contain valuable structural geometry that should be propagated forward rather than discarded at the LLM interface.

Limitations

AudioTETRIS's acoustic prior assumes speech-encoder hidden states exhibit large-norm transitions at phonetic boundaries - validated for Whisper-based encoders (Qwen2-Audio, Audio Flamingo 3) and the Conformer-based Phi-4-Multimodal encoder, but not confirmed for encoders trained with different objectives or aggressive temporal downsampling (e.g., HTSAT-style audio-event encoders). Second, while the early pruning layer is fixed by construction, the later crossover layer $\ell^\dagger$ requires a short backbone-specific calibration pass (Appendix A.2.2) - a one-time cost, but the method is not fully zero-shot to unseen architectures without it. Third, because pruning happens progressively inside the decoder rather than once before prefill, AudioTETRIS carries a small TFLOP overhead relative to single-shot methods (§5.3), though this is largely offset by its accuracy gains. Finally, our evaluation spans four English-source benchmarks; results on noisier, or multilingual audio remain untested, and pushing compression beyond the rates studied here would likely erode lexical accuracy first, given ASR's sensitivity to aggressive compression.

AI USE STATEMENT

We used LLMs as a general-purpose writing aid - for sentence-level editing, LaTeX formatting, and phrasing suggestions - during preparation of this manuscript. We also used them to generate ideas for paper visualizations as well as to automate, troubleshoot, and debug parts of the analysis code. All research ideas, methodology, experiments, and results reported in this paper are the work of the authors, who take full responsibility for the content of the manuscript.

REPRODUCIBILITY STATEMENT

AudioTETRIS is training-free, so no training procedure, optimizer, or learning-rate schedule needs to be specified for reproduction. All components needed to reproduce our results are given in the main text and appendix: the phoneme transition score (Eq. 1-2), the depth-adaptive attention score and its automated crossover-layer discovery procedure (Eq. 3-4, Appendix A.2.2). A one-time, per-backbone calibration pass (Appendix A.2.2) locates the crossover layer $\ell^\dagger$ from attention statistics; this involves no gradient computation or parameter updates. The full fused-score pruning schedule are summarized as pseudocode in Algorithms 1 and 2 (Appendix A.2.1). Pruning layers, per-stage retention rates, and the closed-form multi-stage schedule are specified in §5.1 and Appendix A.3.3 (Eq. 9, Table 7). Backbone checkpoints, exact dataset splits, and their licenses are listed in Appendix A.1.1 (Tables 2, 3); alignment sources used for the boundary-correlation analysis (§4) are described in Appendix A.2.4. We include our full implementation, including the pruning modules described in Algorithms 1 and 2 and the evaluation scripts used to produce Tables 1-5, in an anonymized repository at https://anonymous.4open.science/r/AudioTETRIS-E87F, and will release the full documented code publicly upon acceptance.

REFERENCES

Matthew Aylett and Alice Turk. The smooth signal redundancy hypothesis: A functional explanation for relationships between redundancy, prosodic prominence, and duration in spontaneous speech. *Language and Speech*, 47(1):31–56, 2004. doi: 10.1177/00238309040470010201.

Swarup Ranjan Behera, Abhishek Dhiman, Karthik Gowda, and Aalekhya Satya Narayani. Fastast: Accelerating audio spectrogram transformer via token merging and cross-model knowledge distillation. In *Interspeech 2024*, pp. 4303–4307, 2024. doi: 10.21437/Interspeech.2024-1185. URL https://www.isca-archive.org/interspeech_2024/behera24_interspeech.html.

Saurabhchand Bhati, Samuel Thomas, Hilde Kuehne, Rogerio Feris, and James Glass. Towards audio token compression in large audio language models. *arXiv preprint arXiv:2511.20973*, 2025.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *International Conference on Learning Representations (ICLR)*, 2023.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower Provost, Samuel Kim, Jeannette Chang, Sungbok Lee, and Shrikanth Narayanan. Iemocap: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42:335–359, 12 2008. doi: 10.1007/s10579-008-9076-6.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision (ECCV)*, pp. 19–35. Springer, 2024. doi: 10.1007/978-3-031-73004-7_2.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, and Junyang Lin. Qwen2-audio technical report, 2024. URL https://arxiv.org/abs/2407.10759.

CorentinJ. Librispeech alignments. https://github.com/CorentinJ/librispeech-alignments, 2019. GitHub repository.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. Pengi: An audio language model for audio tasks. In *Advances in Neural Information Processing Systems*, volume 36, pp. 18090–18108, 2023.

Sadaoki Furui. On the role of spectral transition for speech perception. *The Journal of the Acoustical Society of America*, 80(4):1016–1025, 1986a. doi: 10.1121/1.393842.

Sadaoki Furui. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(1):52–59, 1986b. doi: 10.1109/TASSP.1986.1164788.

Sreyan Ghosh, Arushi Goel, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. doi: 10.52202/085713-1396.

Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Dinesh Manocha, and Bryan Catanzaro. Audio flamingo 2: An audio-language model with long-audio understanding and expert reasoning abilities. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 19358–19405, 2025b.

Chao Gong, Depeng Wang, Zhipeng Wei, Ya Guo, Huijia Zhu, and Jingjing Chen. Echoingpixels: Aliasing-resistant joint token reduction for audio-visual llms. In *Forty-third International Conference on Machine Learning*, 2026.

Peize He, Zichen Wen, Yubo Wang, Yuxuan Wang, Xiaoqian Liu, Jiajie Huang, Zehui Lei, Zhuangcheng Gu, Xiangqi Jin, Jiabing Yang, Kai Li, Zhifei Liu, Weijia Li, Cunxiang Assisting Wang, Conghui He, and Linfeng Zhang. Audiomarathon: A comprehensive benchmark for long-context audio understanding and efficiency in audio llms. *arXiv preprint arXiv:2510.07293*, 2025.

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 *ICLR*, 2022.

Chaeyoung Jung, Youngjoon Jang, Seungwoo Lee, and Joon Son Chung. Fastav: Efficient token pruning for audio-visual large language model inference. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2026. doi: 10.1109/ICASSP5730.2026.11463130.

Hassan Salami Kavaki and Michael I. Mandel. Audio sparse-transformer for speech classification. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025. doi: 10.1109/ICASSP63290.2025.10890475.

Yuang Li, Yu Wu, Jinyu Li, and Shujie Liu. Accelerating Transducers through Adjacent Token Merging. In *Interspeech 2023*, pp. 1379–1383, 2023. doi: 10.21437/Interspeech.2023-599.

Luca Della Libera, Cem Subakan, and Mirco Ravanelli. Beyond fixed frames: Dynamic character-aligned speech tokenization, 2026. URL https://arxiv.org/abs/2601.23174.

Yueqian Lin, Yuzhe Fu, Jingyang Zhang, Yudong Liu, Jianyi Zhang, Jingwei Sun, Hai Li, and Yiran Chen. Speechprune: Context-aware token pruning for speech information retrieval. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, 2025.

Jiale Luo, Xiaoyu Liang, and Haoji Hu. Locality matters for training-free audio token compression in audio-language models. *arXiv preprint arXiv:2605.25179*, 2026.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH 2017)*, pp. 498–502, 2017. doi: 10.21437/Interspeech.2017-1386.

Microsoft, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, Xiyang Dai, Ruchao Fan, Mei Gao, Min Gao, Amit Garg, Abhishek Goswami, Junheng Hao, Amr Hendy, Yuxuan Hu, Xin Jin, Mahmoud Khademi, Dongwoo Kim, Young Jin Kim, Gina Lee, Jinyu Li, Yunsheng Li, Chen Liang, Xihui Lin, Zeqi Lin, Mengchen Liu, Yang Liu, Gilsinia Lopez, Chong Luo, Piyush Madan, Vadim Mazalov, Arindam Mitra, Ali Mousavi, Anh Nguyen, Jing Pan, Daniel Perez-Becker, Jacob Platin, Thomas Portet, Kai Qiu, Bo Ren, Liliang Ren, Sambuddha Roy, Ning Shang, Yelong Shen, Saksham Singhal, Subhojit Som, Xia Song, Tetyana Sych, Praneetha Vaddamanu, Shuohang Wang, Yiming Wang, Zhenghao Wang, Haibin Wu, Haoran Xu, Weijian Xu, Yifan Yang, Ziyi Yang, Donghan Yu, Ishmam Zabir, Jianwen Zhang, Li Lyna Zhang, Yunan Zhang, and Xiren Zhou. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras, 2025. URL https://arxiv.org/abs/2503.01743.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. *ICML*, 2023.

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 *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22857–22867, October 2025.

Kele Shao, Keda Tao, Kejia Zhang, Sicheng Feng, Mu Cai, Yuzhang Shang, Haoxuan You, Can Qin, Yang Sui, and Huan Wang. A survey of token compression for efficient multimodal large language models. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=G2od9JVHkE.

Kenneth N. Stevens. Toward a model for lexical access based on acoustic landmarks and distinctive features. *The Journal of the Acoustical Society of America*, 111(4):1872–1891, 2002. doi: 10.1121/1.1458026.

Changli Tang, Wenyi Yao, Guangke Zhou, Bin Chen, Siyuan Deng, Zhuo Liu, Jiahui Wang, Ziyang Zhou, Xiu Zhang, Zhou Wan, et al. SALMONN: Towards generic hearing abilities for large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *ACM Comput. Surv.*, 55(6), December 2022. ISSN 0360-0300. doi: 10.1145/3530811. URL https://doi.org/10.1145/3530811.

Bin Wang, Xunlong Zou, Geyu Lin, Shuo Sun, Zhuohan Liu, Wenyu Zhang, Zhengyuan Liu, AiTi Aw, and Nancy F. Chen. AudioBench: A universal benchmark for audio large language models. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4297–4316. Association for Computational Linguistics, April 2025a. doi: 10.18653/v1/2025.naacl-long.218. URL https://aclanthology.org.

Changhan Wang, Anne Conneau, Jiatao Gu, and Juan Pino. CoVoST 2 and Massively Multilingual Speech Translation. In *Proc. Interspeech 2021*, pp. 4234–4238, 2021. doi: 10.21437/Interspeech.2021-1773.

Jiatong Wang, Yifan Zhao, Shicong Wang, Hexin Lin, Shugao Wang, Shiyi Feng, and Dongyan Jiang. Say more with less: Variable-frame-rate speech tokenization via adaptive clustering and implicit duration coding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25310–25318, 2025b. URL https://aaai.org.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, and Dahua Lin. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Qian Yang Yang, Hongbin Wang, Zheng Li, Shunian Wang, Shifu Zhang, Meng Liu, Jun Guo, Yan Zhou, Chang Su, Zhao Zhang, Chao Li, and Wei Xu. AIR-bench: Benchmarking large audio-language models via generative comprehension. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1966–1986, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

Xiangyu Zhang, Fuming Fang, Peng Gao, Bin Qin, Beena Ahmed, and Julien Epps. Distinctive feature codec: An adaptive efficient speech representation for depression detection, 2025a. URL https://arxiv.org/abs/2505.18516.

Xiangyu Zhang, Yuxin Li, Haoyang Zhang, Shiqi Han, Hexin Liu, Qiquan Zhang, Beena Ahmed, and Julien Epps. The wer trap: Shattering the illusion of unified tokens in speech language models. *arXiv preprint arXiv:2605.29209*, 2026.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In *International Conference on Machine Learning*, 2025b.

A Appendix

A.1 Implementation Details

A.1.1 Backbones and Checkpoints

Table 2 lists the exact checkpoints and licenses for the three backbones evaluated in this work. All experiments use unmodified pretrained weights; the only backbone-side code changes are load-time monkey-patches to the decoder forward pass implementing AudioTETRIS's pruning (§3). No checkpoint weights were altered or fine-tuned.

Table 2: Pretrained checkpoints used in this work.

Backbone	Checkpoint	License
Qwen2-Audio (Chu et al., 2024)	`Qwen/Qwen2-Audio-7B-Instruct`	Apache 2.0
Phi-4-Multimodal (Microsoft et al., 2025)	`microsoft/Phi-4-multimodal-instruct`	MIT
Audio Flamingo 3 (Ghosh et al., 2025a)	`nvidia/audio-flamingo-3-hf`	NVIDIA OneWay Noncommercial License*

* Non-commercial research use only; see checkpoint card for full terms.

A.1.2 Datasets, Splits, and Evaluation

Table 3 lists the datasets used for evaluation, the exact splits, and their licensing. We accessed all four via the `AudioLLMs` organization's re-packaged mirrors on the Hugging Face Hub, used throughout the AudioBench evaluation harness (Wang et al., 2025a), rather than the original distribution channels for each corpus. For the CoVoST2 phoneme alignments generated for Appendix A.2.4, these are derivative annotations produced by us using the Montreal Forced Aligner and are not redistributed third-party material.

Table 3: Evaluation datasets, splits, and licensing.

Dataset	Split used	Original license
LibriSpeech (test-other) (Panayotov et al., 2015)	`AudioLLMs/librispeech_test_other,test`	CC BY 4.0
IEMOCAP (Busso et al., 2008)	`AudioLLMs/iemocap_emotion_recognition,test`	Custom non-commercial research license (USC SAIL)
MMAU-mini (Sakshi et al., 2025)	`AudioLLMs/MMAU-mini,test`	Not explicitly stated by source
CoVoST2 (En→Zh) (Wang et al., 2021)	`AudioLLMs/covost2_en_zh_test,test`	CC0 (per original release); Hub mirror tagged CC BY-NC 4.0

A.1.3 SparseVLM Adaptation to Fixed-Rate Speech Pruning

SparseVLM (Zhang et al., 2025b) originally proposed an adaptive, dynamic pruning schedule based on the rank of the visual-text attention matrix. However, its public release[2] replaces this dynamic controller with static token-retention schedules. Because the dynamic rank-based ratio is absent from all usable releases (`main`, `v1.5`, `v2.0`), we could not evaluate SparseVLM's originally described adaptive deployment directly. Instead, we adapt its public attention-guided scoring mechanism to our fixed-rate setup for a direct, controlled comparison.

Adaptation. We retain SparseVLM's core scoring mechanism - using the text raters, and text-token attention to rate audio token importance - and apply it at layers $\{1, 15\}$ (Appendix A.2.2). At each stage, we keep the top fraction r of audio tokens by score to match our evaluated compression targets $(0.36, 0.51, 0.64, 0.75)$. This adaptation isolates the impact of SparseVLM's attention-based importance metric under identical layer placements and compute budgets.

A.1.4 Attention Backend for Score-Based Pruning

AudioTETRIS and the attention-score-based baselines we compare against (SparseVLM Zhang et al. (2025b), SpeechPrune (Lin et al., 2025)) all derive token-importance scores from attention probabilities between query and audio tokens. Fused attention kernels such as FlashAttention-2 (Dao, 2024) and PyTorch's scaled-dot-product attention (SDPA) do not materialize these probabilities: both return `attn_weights=None`, since the attention matrix is never explicitly formed.

[2] https://github.com/Gumpest/SparseVLMs, accessed May 2026, commit 87fe4319430e079a778c25d4ddf4588a3c4038ca 2025-12-22.

Eager attention is the only implementation that exposes the full $[B, H, N^{(\ell)}, N^{(\ell)}]$ probability tensor required for scoring.

We therefore run the attention backend in eager mode only at the layers where token scores are computed, and restore the model's configured kernel (FlashAttention-2 (Dao, 2024), in our setup) for the forward pass that produces the hidden states and KV cache used downstream. This substitution is local to the small set of scoring layers (§3) rather than applied model-wide, so the FlashAttention-2 fast path is preserved for the remainder of the network's depth. We apply this identically across AudioTETRIS and all attention-score-based baselines, so the latency and FLOPs figures reported in §5.2 reflect the same hybrid execution path for every method and remain directly comparable.

We note that this eager-attention fallback is a memory-bandwidth-bound cost that is not reflected in the TFLOPs column, which counts compute operations only; it is, however, fully captured in our reported wall-clock latency, since all latency numbers are measured end-to-end on the actual hybrid execution path (eager at scoring layers, FlashAttention-2 elsewhere). Because the eager fallback is confined to exactly two decoder layers per backbone (layer 1 and that backbone's calibrated $\ell^\dagger$; §5.1), its contribution to total latency is small in practice, and it could be eliminated entirely in a production deployment with a custom fused kernel that computes only the row/column marginal attention sums our scoring requires, without materializing the full attention matrix - a one-time kernel-engineering cost rather than a potential structural limitation of the method.

A.2 Method Details and Analysis

A.2.1 Algorithmic Summary

For reproducibility, Algorithms 1 and 2 summarize the full pipeline described in §3. Algorithm 1 is a one-time, offline calibration step run once per backbone to locate the crossover layer $\ell^\dagger$ (§A.2.2); Algorithm 2 is the training-free inference procedure applied per input, using $\ell^\dagger$ from calibration to determine which attention branch of Eq. 3 to use at each pruning layer.

Algorithm 1 Crossover Layer Calibration (offline, run once per backbone)

Require: Calibration set $\mathcal{U}$ (audio-query pairs), backbone with L_{dec} decoder layers
Ensure: Crossover layer $\ell^\dagger$

for each utterance $u \in \mathcal{U}$ **do**
  Run forward pass; record attention tensors $A_u^{(\ell)}$ for all layers $\ell = 1, \ldots, L_{\text{dec}}$
end for
for $\ell = 1, \ldots, L_{\text{dec}}$ **do**
  $a_{\text{intra}}(\ell) \leftarrow$ mean attention mass from $\mathcal{Q}_a$ rows onto audio tokens (Eq. 3, upper case), averaged over $\mathcal{U}$
  $a_{\text{cross}}(\ell) \leftarrow$ mean attention mass from $\mathcal{Q}_q$ rows onto audio tokens (Eq. 3, lower case), averaged over $\mathcal{U}$
end for
$\ell^\dagger \leftarrow \min\{\ell : a_{\text{cross}}(\ell) \geq a_{\text{intra}}(\ell)\}$ ▷ Eq. 4
return $\ell^\dagger$

A.2.2 Pruning Layer Selection

AudioTETRIS prunes at two decoder layers (§5.1), and the two are chosen by different criteria because they serve different roles. The early layer is about saving compute: it is fixed by design as the earliest depth at which the attention term exists, and pruning there is viable because the acoustic prior compensates for weak early attention. The late layer $\ell^\dagger$ is about task relevance: it is calibrated per backbone from attention crossover statistics to locate where cross-modal grounding takes over (§A.2.2). We motivate each choice separately below.

Early Pruning Layer. Because self-attention cost accrues at every subsequent decoder layer, the earlier a token is removed, the larger the cumulative FLOP reduction across the rest of the forward pass. We therefore place the first pruning stage at layer 1 - the earliest layer at which the depth-adaptive attention score $\tilde{p}_t^{(\ell)}$ (Eq. 3) is defined, since it requires at least one decoder self-attention

Algorithm 2 AudioTETRIS Inference (training-free, applied per input)

Require: Audio encoder hidden states $H = \{h_1, \ldots, h_T\}$, pruning layers $\{\ell_1, \ell_2\}$ with $\ell_1 < \ell^\dagger \leq \ell_2$ (in practice $\ell_2 = \ell^\dagger$, the backbone's calibrated crossover layer; §5.1), target retention fractions $\{k_{\ell_1}, k_{\ell_2}\}$, decoder with L_{dec} layers
Ensure: Compressed decoder output

Stage 1: Acoustic Transition Prior *(computed once, before decoding)*

$d_1 \leftarrow 0$
for $t = 2, \ldots, T$ **do**
  $d_t \leftarrow \|h_t - h_{t-1}\|_2^2$ ▷ Eq. 1
end for
$\bar{d} \leftarrow \frac{1}{T} \sum_{i=1}^{T} d_i$
for $t = 1, \ldots, T$ **do**
  $w_t \leftarrow d_t / \bar{d}$ ▷ Eq. 2
end for

Stage 2: Depth-Adaptive Pruning *(during decoder forward pass)*

$\mathcal{T} \leftarrow \{1, \ldots, T\}$ ▷ surviving audio-token indices
for $\ell = 1, \ldots, L_{\text{dec}}$ **do**
  Run decoder layer ℓ on current sequence (audio tokens indexed by $\mathcal{T}$, plus query tokens)
  if $\ell \in \{\ell_1, \ell_2\}$ **then**
    if $\ell < \ell^\dagger$ **then**
      $\tilde{p}_t^{(\ell)} \leftarrow$ mean attention from audio-query rows $\mathcal{Q}_a$ onto token t, $\forall t \in \mathcal{T}$ ▷ Eq. 3, intra-audio
    else
      $\tilde{p}_t^{(\ell)} \leftarrow$ mean attention from text-query rows $\mathcal{Q}_q$ onto token t, $\forall t \in \mathcal{T}$ ▷ Eq. 3, cross-modal
    end if
    $s_t^{(\ell)} \leftarrow \tilde{p}_t^{(\ell)} \cdot w_t, \quad \forall t \in \mathcal{T}$ ▷ Eq. 5
    $\mathcal{T} \leftarrow$ indices of the top $\lceil k_\ell \cdot |\mathcal{T}| \rceil$ tokens in $\mathcal{T}$ ranked by $s_t^{(\ell)}$
    Evict pruned tokens from hidden states and KV cache; update RoPE position indices for $\mathcal{T}$
  end if
end for
return final decoder output

computation to exist. Deferring pruning to a deeper early layer might yield a marginally sharper intra-audio attention signal, but this benefit is outweighed by the compute lost across every additional layer the full, unpruned token sequence has to pass through.

Early pruning works well for AudioTETRIS because the layer-1 score does not rely on attention alone. The fused score $s_t^{(1)} = \tilde{p}_t^{(1)} \cdot w_t$ (Eq. 5) combines the attention-based importance $\tilde{p}_t^{(1)}$, computed from a single decoder layer, with the acoustic transition prior w_t (Eq. 2). The prior comes directly from the audio encoder's hidden states, so it is already fully informative before the decoder even starts (§4).

The prior's independence from decoder depth matters because attention alone is a weak signal at layer 1, where the decoder has not yet built up much context. The acoustic prior compensates for this weakness, making the fused score far more reliable at this depth than attention alone would be. This reliability is what lets us prune aggressively at layer 1 without the accuracy collapse that a purely attention-based method suffers. Our component ablation supports this conclusion (§6.2, Table 6): 'Only Stage 2' (attention alone) degrades much faster as compression increases than either 'Only Stage 1' or the full fused score.

Late Pruning Layer: Crossover Calibration. Section 3.3 defines the crossover layer $\ell^\dagger$ (Eq. 4) as the first decoder depth at which cross-modal attention mass overtakes intra-audio attention mass, and uses it to set the layer partition $\{\mathcal{L}_{\text{early}}, \mathcal{L}_{\text{later}}\}$ for the depth-adaptive attention score $\tilde{p}_t^{(\ell)}$ (Eq. 3). Here we describe the calibration procedure used to estimate $\ell^\dagger$ and report the resulting per-layer attention statistics for each backbone.

Calibration set. We estimate $\ell^\dagger$ from 150 utterances drawn from LibriSpeech test-other (Panayotov et al., 2015), disjoint from the utterances used for downstream evaluation in §5.2. For each calibration utterance, we run a forward pass through the target backbone and record the full decoder attention tensors $A^{(\ell)}$ at every layer, with no gradient updates or parameter changes. Calibration takes under 4 minutes on a single A100 and is amortized across all subsequent inputs for that backbone.

Per-layer attention mass. For each decoder layer ℓ, we define the mean intra-audio and cross-modal attention mass, restating the two branches of Eq. 3 without the depth partition:

$$a_{\text{intra}}(\ell) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{H \cdot |\mathcal{Q}_a| \cdot T_u} \sum_{h=1}^{H} \sum_{i \in \mathcal{Q}_a} \sum_{t=1}^{T_u} A^{(\ell)}_{h,i,t}, \tag{7}$$

$$a_{\text{cross}}(\ell) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{H \cdot |\mathcal{Q}_q| \cdot T_u} \sum_{h=1}^{H} \sum_{i \in \mathcal{Q}_q} \sum_{t=1}^{T_u} A^{(\ell)}_{h,i,t}, \tag{8}$$

where $\mathcal{U}$ is the calibration set, T_u is the number of audio tokens in utterance u, and $\mathcal{Q}_a$, $\mathcal{Q}_q$ are the audio-query and text-query row index sets as in Eq. 3. The crossover layer is then $\ell^\dagger = \min\{\ell : a_{\text{cross}}(\ell) \geq a_{\text{intra}}(\ell)\}$, matching Eq. 4.

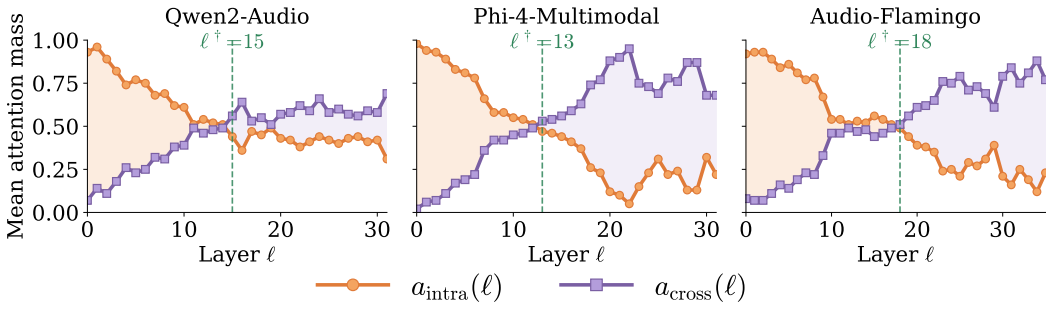


Figure 7: Per-layer intra-audio ($a_{\text{intra}}(\ell)$) and cross-modal ($a_{\text{cross}}(\ell)$) attention mass on the LibriSpeech test-other calibration subset, for Qwen2-Audio (Chu et al., 2024), Phi-4-Multimodal (Microsoft et al., 2025), and Audio Flamingo 3 (Ghosh et al., 2025a) The dashed vertical line marks the identified crossover layer $\ell^\dagger$ for each backbone. Intra-audio attention dominates in early layers and is progressively overtaken by cross-modal attention as depth increases, consistent with Figure 2.

Results. As shown in Figure 7, all three backbones exhibit the same qualitative pattern: intra-audio attention mass is highest in early layers and declines monotonically (on average) with depth, while cross-modal attention mass rises correspondingly, crossing over in the middle third of the network. The calibration procedure identifies $\ell^\dagger = 15$ for Qwen2-Audio, consistent with the pruning schedule used in §5.1, $\ell^\dagger = 13$ for Phi-4-Multimodal, and $\ell^\dagger = 18$ for Audio Flamingo 3. The crossover layer therefore shifts across backbones with differing encoder-decoder depth and subsampling rates.

A.2.3 Transition Prior Computation Pipeline

Figure 8 illustrates the end-to-end computation of the phoneme transition score b_t and its relationship to ground-truth phoneme transitions, using a toy example spanning the phonemes */s/-/t/-/r/-/iy/-/m/*. The raw derivative d_t (Eq. 1) is computed from consecutive encoder hidden states, producing sparse spikes near true phonetic transitions but with substantial token-to-token noise. Gaussian smoothing (Eq. 6) converts this into the continuous transition prior b_t, which we correlate against both the discrete ground-truth transition indicator y_{hard} and its Gaussian-smoothed counterpart y_{soft} to obtain r_{hard} and r_{soft} respectively.

A.2.4 Boundary Correlation on CoVoST2 and IEMOCAP

Section 4 reports boundary-correlation results on LibriSpeech (Figure 3), using the MFA-based forced alignments made available through (CorentinJ, 2019). Here we repeat the same analysis on

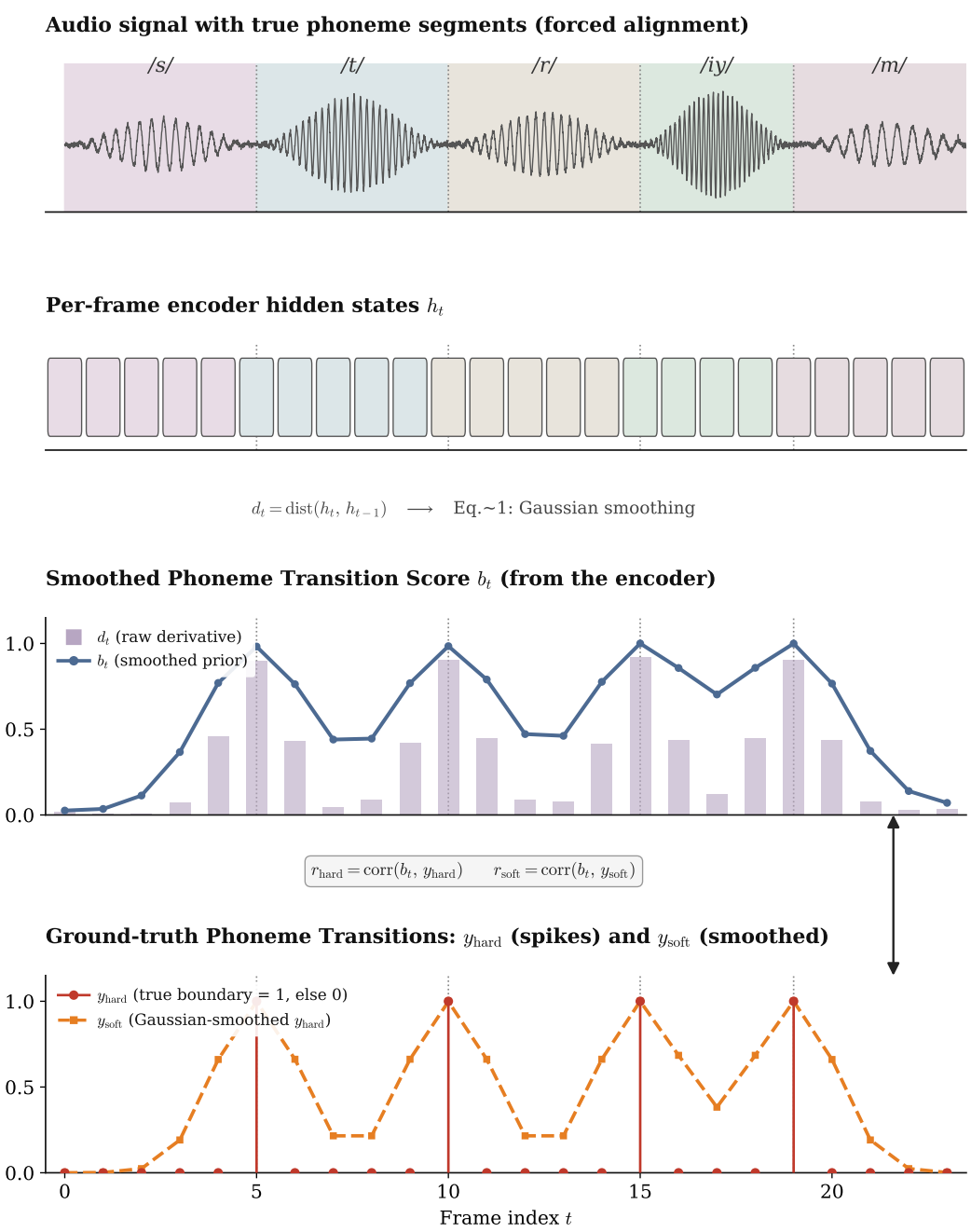


Figure 8: Schematic of the phoneme-transition-aware prior pipeline. **Top:** audio waveform with forced-alignment phoneme segments. **Second row:** per-token encoder hidden states h_t, colored by phoneme membership. **Third row:** raw derivative d_t (bars) and the Gaussian-smoothed transition score b_t (line). **Bottom:** ground-truth transition labels, shown as discrete spikes y_{hard} and their Gaussian-smoothed form y_{soft}, used to compute r_{hard} and r_{soft}.

our two other evaluation datasets, CoVoST2 (speech translation) (Wang et al., 2021) and IEMOCAP (emotion recognition) (Busso et al., 2008), to confirm that the mid-network correlation peak is not an artifact of LibriSpeech's read-speech domain.

Alignment sources. IEMOCAP ships with official word-, syllable-, and phoneme-level forced alignments as part of its dataset release, generated with Sphinx-III (Busso et al., 2008); we use

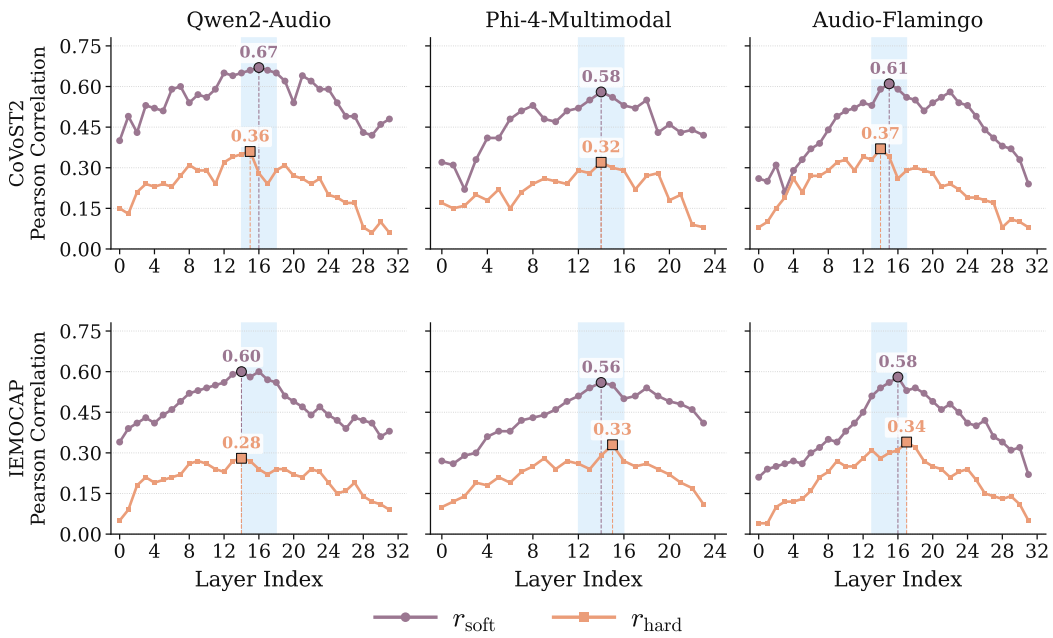


Figure 9: Layer-wise boundary correlation (r_{soft}, r_{hard}) across Qwen2-Audio, Phi-4-Multimodal, and Audio Flamingo 3, computed on CoVoST2 (**top**, MFA alignments generated in-house) and IEMOCAP (**bottom**, official released alignments). Shaded regions mark each backbone's mid-network sweet spot, consistent with Figure 3. Dashed vertical lines and annotated values mark the peak layer for each curve.

these directly. No comparable off-the-shelf alignment resource exists for CoVoST2, so we generated phoneme alignments ourselves using the Montreal Forced Aligner (MFA) (McAuliffe et al., 2017) on CoVoST2's English-source audio and transcripts, matching the alignment methodology used for the LibriSpeech release to keep the comparison consistent across datasets.[3] For both datasets we recompute b_t per layer and correlate against the same $y_{\text{hard}}/y_{\text{soft}}$ construction as in Section 4, holding $\sigma = 2$ fixed.

As shown in Figure 9, both datasets reproduce the mid-network peak observed on LibriSpeech across all three backbones, with r_{soft} peaking in the same layer range identified by the sweet-spot shading (shared with Figure 3). Peak r_{soft} is consistently higher on CoVoST2 than on IEMOCAP for every backbone (e.g., Qwen2-Audio: 0.67 vs. 0.60; Audio Flamingo 3: 0.61 vs. 0.58), which is consistent with our finding in Section 5.2 that fine-grained lexical tasks depend more heavily on precise boundary localization than paralinguistic classification does.

A.2.5 Sensitivity to the Smoothing Width σ

Eq. 6 defines the transition prior b_t by convolving the raw distance signal d_t with a Gaussian kernel of width σ, and we apply the same σ to smooth the discrete forced-alignment labels y_{hard} into the continuous target y_{soft}. In the main text we fix $\sigma = 2$ (encoder tokens). Here we ablate this choice to characterize how r_{soft} and r_{hard} vary as σ moves away from the selected smoothing width.

We sweep $\sigma \in \{0.5, 1, 2, 3, 4, 5, 6, 8\}$ encoder tokens and recompute r_{soft} and r_{hard} at each value, holding the layer fixed at the mid-network sweet spot identified in Section 4 for each backbone. Because Phi-4-Multimodal's audio encoder subsamples more aggressively than Qwen2-Audio's or Audio Flamingo's, the same σ in token units corresponds to roughly twice the real-time smoothing

[3] IEMOCAP's original release notes that alignment accuracy was screened but not formally evaluated by the corpus authors; we flag this as a caveat on alignment quality relative to the MFA-based LibriSpeech and CoVoST2 alignments used elsewhere in this work.

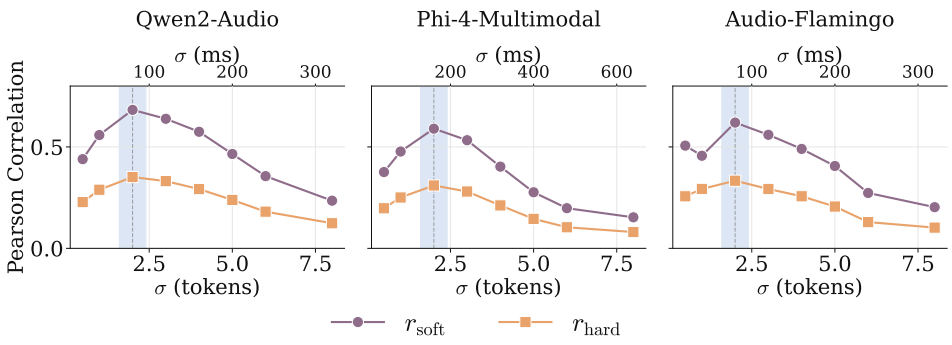


Figure 10: Sensitivity of the transition-prior correlation to the Gaussian smoothing width σ, on all backbones. Each panel sweeps σ (encoder tokens, with milliseconds on the secondary top axis) and reports r_{soft} and r_{hard} at that backbone's mid-network sweet-spot layer. The shaded band and dashed line mark $\sigma = 2$, the value used throughout the main text.

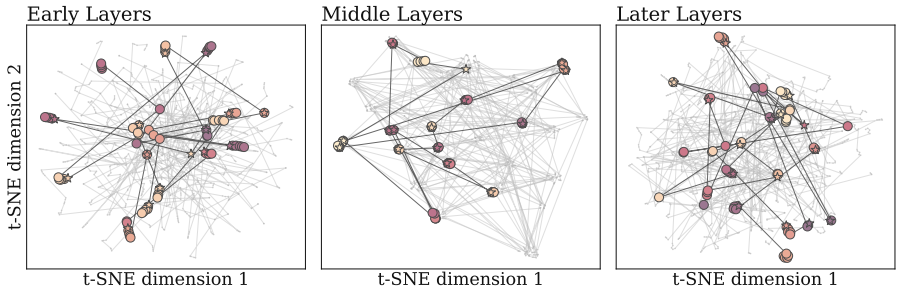


Figure 11: Qualitative analysis of Qwen2-Audio encoder token embedding trajectories for a single LibriSpeech utterance, projected via t-SNE and colored by phoneme identity, at early, middle, and late encoder depths. Colored points ● denote individual tokens where different colors represent different phonemes; Star markers ★ indicate frames at phoneme transitions per forced alignment; Faint gray points ● show other utterances' frames for spatial context.

window for Phi-4-Multimodal; Figure 10 therefore also reports σ in milliseconds on a secondary axis for each backbone.

As shown in Figure 10, both r_{soft} and r_{hard} consistently exhibit an inverted U-shaped curve across all backbones, peaking cleanly at $\sigma = 2$ tokens. Performance degrades on either side: smaller values fail to account for context distribution, while larger values ($\sigma > 2$) excessively blur distinct phonetic transitions.

A.2.6 Qualitative Encoder Embedding Trajectories

Figure 11 complements the quantitative diagnostics of Section 4 with a qualitative view of the same phenomenon. We project the per-token encoder hidden states h_t of a single LibriSpeech test-other utterance into two dimensions with t-SNE at an early (layer 4), middle (layer 16), and late encoder layer (layer 28) of Qwen2-Audio, and trace the utterance's token sequence through the projected space. Frames from other utterances are shown in grey to provide a shared spatial reference.

At the middle layer, which coincides with the peak of both r_{soft} and kurtosis (Figure 3), tokens belonging to the same phoneme form compact clusters. At the early and late layers, tokens are less consolidated by phoneme identity and boundary frames are less clearly separated from within-segment frames, consistent with the lower correlation and kurtosis observed at those depths. Since t-SNE preserves local neighbourhoods but not global distances, we treat this figure as illustrative only. The quantitative evidence for the mid-network sweet spot is given by the correlation and kurtosis analyses in Section 4 and Appendix A.2.4.

Table 4: Extended results for Phi-4-Multimodal across all compression ratios.

Method	LibriSpeech WER ↓	IEMOCAP Acc. ↑	MMAU-mini Acc. ↑	CoVoST2 BLEU ↑	Latency (s)	TFLOPs
No Compression	$4.85^{\pm 0.34}$	$39.34^{\pm 0.29}$	$54.0^{\pm 0.85}$	$36.06^{\pm 0.07}$	1.15	1.403
Compression Ratio 0.36						
Random	$5.64^{\pm 1.92}$	$39.00^{\pm 0.51}$	$52.60^{\pm 0.89}$	$19.05^{\pm 0.05}$	0.97	0.96
SpeechPrune	$5.55^{\pm 0.06}$	$38.28^{\pm 1.10}$	$53.70^{\pm 0.44}$	$22.60^{\pm 0.07}$	1.01	0.96
SparseVLM†‡	$5.43^{\pm 0.79}$	$39.75^{\pm 0.36}$	$52.70^{\pm 0.43}$	$22.05^{\pm 0.07}$	1.02	1.00
LTBM	$6.60^{\pm 0.00}$	$40.40^{\pm 0.84}$	$54.40^{\pm 0.00}$	$26.76^{\pm 0.19}$	1.00	0.96
AudioTETRIS†	$4.84^{\pm 1.03}$	$40.93^{\pm 0.56}$	$55.20^{\pm 0.31}$	$31.60^{\pm 0.35}$	1.02	1.03
Compression Ratio 0.51						
Random	$7.99^{\pm 0.68}$	$38.03^{\pm 0.90}$	$51.30^{\pm 0.64}$	$16.34^{\pm 0.02}$	0.77	0.77
SpeechPrune	$5.81^{\pm 2.80}$	$36.91^{\pm 0.99}$	$53.20^{\pm 0.92}$	$13.57^{\pm 0.01}$	0.80	0.78
SparseVLM†‡	$6.53^{\pm 1.13}$	$39.34^{\pm 0.72}$	$52.70^{\pm 1.04}$	$17.92^{\pm 0.08}$	0.81	0.84
LTBM	$6.95^{\pm 0.07}$	$39.74^{\pm 0.95}$	$53.60^{\pm 0.00}$	$20.23^{\pm 0.02}$	0.79	0.77
AudioTETRIS†	$5.78^{\pm 1.49}$	$39.90^{\pm 0.20}$	$54.10^{\pm 0.32}$	$31.13^{\pm 0.30}$	0.81	0.88
Compression Ratio 0.64						
Random	$8.49^{\pm 6.06}$	$35.84^{\pm 0.25}$	$50.90^{\pm 0.89}$	$25.29^{\pm 0.02}$	0.61	0.62
SpeechPrune	$6.38^{\pm 2.01}$	$35.38^{\pm 0.40}$	$51.90^{\pm 0.83}$	$6.21^{\pm 0.03}$	0.64	0.63
SparseVLM†‡	$7.53^{\pm 1.09}$	$39.04^{\pm 0.56}$	$52.30^{\pm 0.63}$	$15.67^{\pm 0.04}$	0.65	0.70
LTBM	$6.80^{\pm 0.46}$	$38.97^{\pm 0.77}$	$52.10^{\pm 0.45}$	$15.24^{\pm 0.10}$	0.63	0.62
AudioTETRIS†	$6.04^{\pm 0.87}$	$39.44^{\pm 0.52}$	$53.50^{\pm 0.61}$	$30.67^{\pm 0.09}$	0.67	0.74
Compression Ratio 0.75						
Random	$8.92^{\pm 4.33}$	$33.54^{\pm 0.34}$	$50.60^{\pm 1.04}$	$9.56^{\pm 0.17}$	0.40	0.49
SpeechPrune	$7.85^{\pm 1.52}$	$35.91^{\pm 0.48}$	$51.30^{\pm 0.39}$	$2.50^{\pm 0.02}$	0.46	0.49
SparseVLM†‡	$8.19^{\pm 2.08}$	$38.94^{\pm 0.43}$	$51.80^{\pm 0.48}$	$13.97^{\pm 0.03}$	0.47	0.60
LTBM	$6.89^{\pm 2.34}$	$38.04^{\pm 0.80}$	$51.60^{\pm 0.91}$	$10.80^{\pm 0.09}$	0.45	0.49
AudioTETRIS†	$6.71^{\pm 2.08}$	$39.07^{\pm 0.52}$	$52.50^{\pm 0.38}$	$30.04^{\pm 0.32}$	0.45	0.64

† In-LM pruning (token elimination performed inside the LLM backbone).
‡ Originally developed for the vision modality, adapted here for speech/audio.

A.3 Additional Experiments

A.3.1 Extended Results on Phi-4-Multimodal and Audio Flamingo 3

Section 5.2 and Figure 4 reports results on Phi-4-Multimodal and Audio Flamingo 3 at a single compression ratio (0.51) for brevity. Here we provide the extended results for both backbones across all four evaluated compression ratios (0.36, 0.51, 0.64, 0.75), reported in Table 4 for Phi-4-Multimodal and Table 5 for Audio Flamingo 3. Consistent with the Qwen2-Audio results in Table 1, AudioTETRIS maintains near-lossless performance across both additional backbones at low compression and shows a widening advantage over baselines as compression increases, indicating that the method generalizes across encoder-decoder architectures rather than being tuned to a single backbone.

A.3.2 Extended Component-wise Ablation

Section 6.2 reports the component and stage ablation of AudioTETRIS's fused score $s_t^{(\ell)} = \tilde{p}_t^{(\ell)} \cdot w_t$ at compression ratio 0.51 only. Here we provide the extended results across all four evaluated compression ratios (0.36, 0.51, 0.64, 0.75) and all four benchmarks (LibriSpeech, IEMOCAP, MMAU-mini, CoVoST2), comparing the isolated single-stage variants (Only Stage 1, Only Stage 2) against the full two-stage AudioTETRIS pipeline (Stage 1 + Stage 2). Table 6 shows that combining both stages consistently outperforms either stage in isolation across every compression ratio and benchmark, with the gap widening substantially at higher compression - confirming that the phonetic transition prior and the depth-adaptive attention score contribute complementary, rather than redundant information.

A.3.3 Single-Stage and Alternative Multi-Stage Schedules

Section 5.1 fixes AudioTETRIS's schedule at two stages, layers $\{1, 15\}$, uniform per-stage split, with layer 1 placed for compute savings (Appendix A.2.2) and layer 15 set by crossover calibration on Qwen2Audio (Appendix A.2.2). Here we ablate two orthogonal design choices: *where* in the

Table 5: Extended results for Audio Flamingo 3 across all compression ratios.

Method	LibriSpeech WER ↓	IEMOCAP Acc. ↑	MMAU-mini Acc. ↑	CoVoST2 BLEU ↑	Latency (s)	TFLOPs
No Compression	$3.61^{\pm 0.54}$	$59.11^{\pm 0.87}$	$62.09^{\pm 0.45}$	$37.02^{\pm 0.71}$	1.29	7.93
Compression Ratio 0.36						
Random	$4.02^{\pm 1.21}$	$58.43^{\pm 0.63}$	$60.97^{\pm 0.23}$	$36.82^{\pm 0.10}$	1.13	6.92
SpeechPrune	$4.01^{\pm 2.10}$	$58.23^{\pm 0.59}$	$61.42^{\pm 0.83}$	$36.73^{\pm 0.05}$	1.15	7.09
SparseVLM†‡	$3.78^{\pm 1.45}$	$57.34^{\pm 0.32}$	$58.83^{\pm 0.68}$	$36.18^{\pm 0.06}$	1.14	7.31
LTBM	$3.83^{\pm 0.38}$	$58.11^{\pm 0.41}$	$61.00^{\pm 0.65}$	$35.89^{\pm 0.02}$	1.21	6.98
AudioTETRIS†	$3.71^{\pm 0.45}$	$58.94^{\pm 0.59}$	$61.77^{\pm 0.37}$	$37.31^{\pm 0.19}$	1.15	7.34
Compression Ratio 0.51						
Random	$6.67^{\pm 0.91}$	$56.45^{\pm 0.62}$	$59.62^{\pm 0.28}$	$34.72^{\pm 0.15}$	0.80	5.61
SpeechPrune	$5.18^{\pm 1.49}$	$57.81^{\pm 0.97}$	$61.19^{\pm 0.29}$	$35.18^{\pm 0.16}$	0.87	5.71
SparseVLM†‡	$4.97^{\pm 1.32}$	$57.39^{\pm 0.49}$	$58.14^{\pm 0.52}$	$36.30^{\pm 0.32}$	0.92	6.18
LTBM	$5.87^{\pm 0.18}$	$58.91^{\pm 0.56}$	$60.79^{\pm 0.26}$	$35.17^{\pm 0.13}$	0.91	5.63
AudioTETRIS†	$4.62^{\pm 0.97}$	$58.85^{\pm 0.34}$	$61.59^{\pm 0.72}$	$36.26^{\pm 0.09}$	0.85	6.16
Compression Ratio 0.64						
Random	$10.91^{\pm 1.05}$	$53.29^{\pm 0.33}$	$58.19^{\pm 0.52}$	$33.05^{\pm 0.31}$	0.73	4.53
SpeechPrune	$6.59^{\pm 1.92}$	$56.71^{\pm 0.63}$	$60.48^{\pm 0.64}$	$34.87^{\pm 0.26}$	0.78	4.56
SparseVLM†‡	$6.08^{\pm 1.94}$	$53.98^{\pm 0.62}$	$56.39^{\pm 0.93}$	$35.39^{\pm 0.09}$	0.82	5.14
LTBM	$6.49^{\pm 0.71}$	$57.18^{\pm 0.21}$	$59.23^{\pm 0.20}$	$33.90^{\pm 0.04}$	0.83	4.60
AudioTETRIS†	$5.72^{\pm 0.63}$	$58.31^{\pm 0.26}$	$59.85^{\pm 0.13}$	$35.93^{\pm 0.15}$	0.77	5.14
Compression Ratio 0.75						
Random	$15.69^{\pm 1.12}$	$51.89^{\pm 1.08}$	$54.58^{\pm 0.42}$	$27.72^{\pm 0.20}$	0.51	3.59
SpeechPrune	$11.48^{\pm 2.63}$	$54.78^{\pm 0.53}$	$59.59^{\pm 0.97}$	$31.98^{\pm 0.29}$	0.63	3.68
SparseVLM†‡	$9.59^{\pm 1.27}$	$52.60^{\pm 0.49}$	$53.21^{\pm 0.28}$	$34.25^{\pm 0.02}$	0.62	4.25
LTBM	$8.94^{\pm 0.33}$	$57.31^{\pm 0.29}$	$57.13^{\pm 0.40}$	$33.01^{\pm 0.06}$	0.66	3.52
AudioTETRIS†	$6.39^{\pm 1.03}$	$58.09^{\pm 0.37}$	$60.92^{\pm 0.29}$	$35.94^{\pm 0.14}$	0.60	4.26

† In-LM pruning (token elimination performed inside the LLM backbone).
‡ Originally developed for the vision modality, adapted here for speech/audio.

Table 6: Ablation results comparing individual stages against full AudioTETRIS across different compression ratios. The best-performing combination per compression ratio block is highlighted.

Method	LibriSpeech WER ↓	IEMOCAP Acc. ↑	MMAU-mini Acc. ↑	CoVoST2 BLEU ↑
Compression Ratio 0.36				
Only Stage 1	$5.50^{\pm 0.25}$	$57.45^{\pm 0.39}$	$41.80^{\pm 0.62}$	$31.21^{\pm 0.09}$
Only Stage 2	$9.21^{\pm 0.36}$	$57.30^{\pm 1.19}$	$41.70^{\pm 0.80}$	$30.93^{\pm 0.04}$
Stage 1 + Stage 2	$\mathbf{4.52^{\pm 0.21}}$	$\mathbf{58.04^{\pm 0.84}}$	$\mathbf{42.36^{\pm 0.21}}$	$\mathbf{31.96^{\pm 0.04}}$
Compression Ratio 0.51				
Only Stage 1	$7.39^{\pm 0.09}$	$56.37^{\pm 1.55}$	$41.90^{\pm 0.42}$	$31.18^{\pm 0.08}$
Only Stage 2	$9.42^{\pm 0.32}$	$56.22^{\pm 0.21}$	$41.20^{\pm 0.85}$	$30.55^{\pm 0.13}$
Stage 1 + Stage 2	$\mathbf{5.77^{\pm 0.33}}$	$\mathbf{57.03^{\pm 0.41}}$	$\mathbf{42.91^{\pm 0.36}}$	$\mathbf{31.45^{\pm 0.06}}$
Compression Ratio 0.64				
Only Stage 1	$8.92^{\pm 0.46}$	$56.68^{\pm 1.43}$	$41.50^{\pm 0.14}$	$30.11^{\pm 0.04}$
Only Stage 2	$9.62^{\pm 0.22}$	$55.77^{\pm 0.57}$	$38.70^{\pm 1.13}$	$30.85^{\pm 0.19}$
Stage 1 + Stage 2	$\mathbf{6.62^{\pm 0.29}}$	$\mathbf{57.90^{\pm 0.76}}$	$\mathbf{41.83^{\pm 0.15}}$	$\mathbf{31.24^{\pm 0.11}}$
Compression Ratio 0.75				
Only Stage 1	$13.78^{\pm 0.23}$	$55.02^{\pm 0.14}$	$40.80^{\pm 0.42}$	$30.24^{\pm 0.09}$
Only Stage 2	$15.56^{\pm 0.57}$	$50.94^{\pm 1.06}$	$36.10^{\pm 0.57}$	$24.29^{\pm 0.11}$
Stage 1 + Stage 2	$\mathbf{8.50^{\pm 0.18}}$	$\mathbf{55.34^{\pm 1.25}}$	$\mathbf{41.46^{\pm 0.32}}$	$\mathbf{31.33^{\pm 0.07}}$

decoder the pruning stages are placed and *how* the compression budget is allocated across stages once placement is fixed.

Schedule parameterization. For stages at layers $(\ell_1, \ldots, \ell_n)$ and target keep-fraction $K = 1 - \text{rate}$, we assign each stage a weight w_i with $\sum_i w_i = 1$ and set the per-stage retained fraction

$$r_i = K^{w_i}, \qquad \prod_{i=1}^{n} r_i = K, \tag{9}$$

so every variant reaches the same target compression rate regardless of how w_i is split. Equal weights $w_i = 1/n$ give the uniform split used by the main text; $w_1 > w_2$ front-loads the budget toward the earlier stage, and the reverse back-loads it. Dense variants keep n large and the split

uniform, spreading small drops across many stages instead of a few large ones. Table 7 lists all variants tested. Table 8 reports the resulting per-stage compression rate $(1 - r_i) \times 100\%$ at each evaluated compression rate. For uniform-split variants this is a single number repeated n times; for front-/back-loaded it differs stage to stage, with front-loaded removing a larger share early and back-loaded the reverse.

Table 7: Candidate pruning schedules. Placement variants use the uniform split; front-/back-loaded vary only w_i at the default layers $\{1, 15\}$.

Schedule	Layers	n stages	Split
Single-stage @ 1	1	1	–
Single-stage @ 15	15	1	–
Two-stage $\{1, 15\}$ (default)	1, 15	2	Uniform
Two-stage $\{1, 19\}$	1, 19	2	Uniform
Three-stage $\{1, 8, 15\}$	1, 8, 15	3	Uniform
Front-loaded	1, 15	2	$w = (0.7, 0.3)$
Back-loaded	1, 15	2	$w = (0.3, 0.7)$
Dense (every ∼3 layers, full depth)	1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31	11	Uniform

Table 8: Per-stage token-drop rate (% of currently live tokens removed at that stage) for each schedule, at every evaluated compression rate. Equal per-stage rates are written once as "$X\%$ $(\times n)$"; front-/back-loaded list rates stage by stage.

Schedule	Layers	rate 0.36	rate 0.51	rate 0.64	rate 0.75
Single-stage @ 1	1	36%	51%	64%	75%
Single-stage @ 15	15	36%	51%	64%	75%
Two-stage (default)	1, 15	20% (×2)	30% (×2)	40% (×2)	50% (×2)
Two-stage $\{1, 19\}$	1, 19	20% (×2)	30% (×2)	40% (×2)	50% (×2)
Three-stage	1, 8, 15	14% (×3)	21% (×3)	29% (×3)	37% (×3)
Front-loaded	1, 15	27%/13%	39%/19%	51%/26%	62%/34%
Back-loaded	1, 15	13%/27%	19%/39%	26%/51%	34%/62%
Dense (11 stages)	1,4,...,31	4% (×11)	6% (×11)	9% (×11)	12% (×11)

Table 9: Schedule ablation, LibriSpeech WER (%, lower is better), Qwen2-Audio, across compression rates.

Schedule	rate 0.36	rate 0.51	rate 0.64	rate 0.75
Single-stage @ 1	$6.51^{\pm 0.14}$	$8.92^{\pm 0.17}$	$12.40^{\pm 0.36}$	$18.78^{\pm 0.41}$
Single-stage @ 15	$5.44^{\pm 0.61}$	$6.93^{\pm 0.42}$	$8.94^{\pm 0.48}$	$12.89^{\pm 0.82}$
Two-stage $\{1, 15\}$ (default)	$4.52^{\pm 0.21}$	$\mathbf{5.77^{\pm 0.33}}$	$6.62^{\pm 0.29}$	$8.50^{\pm 0.18}$
Two-stage $\{1, 19\}$	$4.78^{\pm 0.38}$	$6.02^{\pm 0.35}$	$6.94^{\pm 0.62}$	$8.96^{\pm 0.31}$
Three-stage $\{1, 8, 15\}$	$\mathbf{4.36^{\pm 0.71}}$	$5.93^{\pm 0.71}$	$\mathbf{6.47^{\pm 0.72}}$	$\mathbf{8.12^{\pm 0.73}}$
Front-loaded	$4.99^{\pm 0.69}$	$6.72^{\pm 0.76}$	$7.68^{\pm 0.29}$	$10.73^{\pm 0.28}$
Back-loaded	$5.73^{\pm 0.31}$	$7.34^{\pm 0.62}$	$9.38^{\pm 0.34}$	$14.53^{\pm 0.42}$
Dense (11 stages)	$7.13^{\pm 0.45}$	$8.79^{\pm 0.57}$	$10.09^{\pm 0.62}$	$12.92^{\pm 0.36}$

Results. We find that single-stage compression at layer 15 (6.93% WER at rate 0.51; Table 9) consistently outperforms layer 1 (8.92%), highlighting the role of text-conditioned attention in token ranking. Splitting the compression budget across two stages improves upon single-stage baselines across all rates, regardless of the chosen layers. Within two-stage setups, our default $\{1, 15\}$ schedule achieves the strongest performance (5.77% at rate 0.51), outperforming $\{1, 19\}$ (6.02%) by 0.2-0.5 percentage points across the sweep.

The three-stage schedule $\{1, 8, 15\}$ yields small gains over the two-stage default at three of the four evaluated rates - 4.36% vs. 4.52% WER at rate 0.36, 6.47% vs. 6.62% at rate 0.64, and 8.12% vs.

Table 10: Efficiency and downstream-task trade-offs at compression rate 0.51. Latency is relative to the uncompressed baseline ($1.00\times$).

Schedule	WER	IEMOCAP	MMAU	CoVoST	Latency	TFLOPs
Single-stage @ 1	$8.92^{\pm 0.17}$	$53.55^{\pm 0.50}$	$39.81^{\pm 0.41}$	$27.05^{\pm 0.05}$	**0.98**	**5.51**
Single-stage @ 15	$6.93^{\pm 0.42}$	$55.83^{\pm 0.52}$	$41.52^{\pm 0.24}$	$29.53^{\pm 0.09}$	0.99	5.92
Two-stage $\{1, 15\}$ (default)	$\mathbf{5.77^{\pm 0.33}}$	$57.03^{\pm 0.41}$	$42.91^{\pm 0.36}$	$31.45^{\pm 0.06}$	1.00	6.13
Two-stage $\{1, 19\}$	$6.02^{\pm 0.35}$	$57.41^{\pm 0.47}$	$43.18^{\pm 0.18}$	$\mathbf{31.69^{\pm 0.07}}$	1.03	6.13
Three-stage $\{1, 8, 15\}$	$5.93^{\pm 0.71}$	$55.68^{\pm 0.37}$	$41.67^{\pm 0.72}$	$31.45^{\pm 0.10}$	1.04	6.32
Front-loaded	$6.72^{\pm 0.76}$	$56.91^{\pm 0.19}$	$41.48^{\pm 0.39}$	$30.20^{\pm 0.18}$	1.02	5.71
Back-loaded	$7.34^{\pm 0.62}$	$55.93^{\pm 0.31}$	$41.44^{\pm 0.53}$	$26.68^{\pm 0.04}$	1.06	6.21
Dense (11 stages)	$8.79^{\pm 0.57}$	$\mathbf{57.62^{\pm 0.81}}$	$\mathbf{43.30^{\pm 0.64}}$	$30.90^{\pm 0.07}$	1.21	6.93

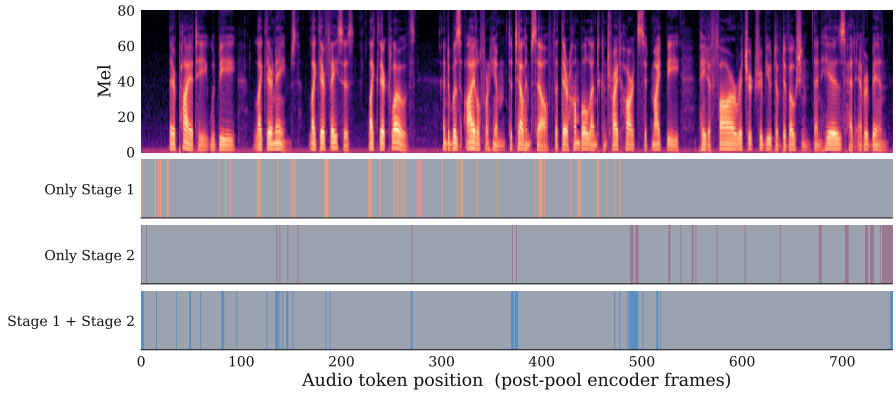
8.50% at rate 0.75 - but this advantage is not monotonic: at rate 0.51 the ordering reverses, with the two-stage default outperforming three-stage (5.77% vs. 5.93%). Given this inconsistency and the added latency overhead of a third pruning stage (Table 10), we retain the two-stage $\{1, 15\}$ schedule as the default configuration used throughout the main text, rather than the marginally stronger but less consistent three-stage variant.

Redistributing the budget within two-stage setups reveals a clear asymmetry: front-loading (larger early cut) remains competitive with the default (4.99% vs. 4.52% at rate 0.36), whereas back-loading degrades sharply under higher compression (14.53% vs. 10.73% at rate 0.75). This aligns with our analysis in Appendix A.2.2: heavy pruning in later layers, where cross-modal attention dominates, incurs a higher penalty than pruning early while representations remain largely intra-audio. Finally, the 11-stage dense schedule underperforms at all rates, showing that frequent, fine-grained pruning could accumulate more errors for the ASR task.

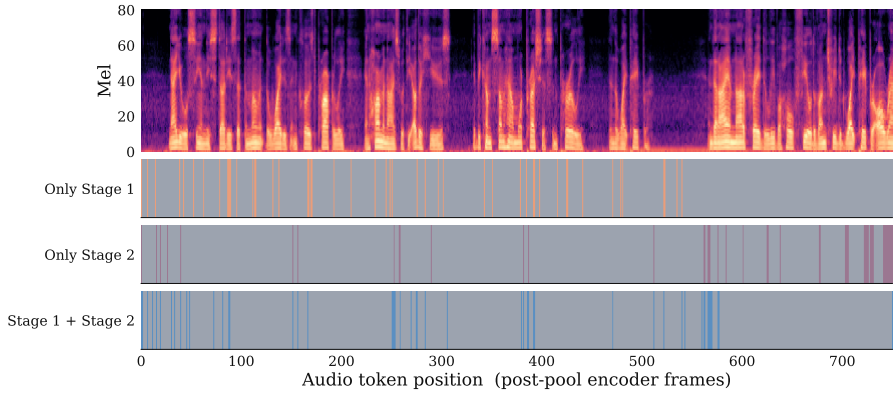
Table 10 highlights the efficiency trade-offs at rate 0.51. We observe that the two-stage compression schedule offers one of the most favorable balance between accuracy and computational cost. In contrast, three-stage and dense schedules introduce a slight overhead ($1.04\times$ and $1.21\times$ latency, respectively) without consistent downstream gains, confirming the 2-stage compression is an effective overall configuration.

A.3.4 Qualitative Examples of Token Survival

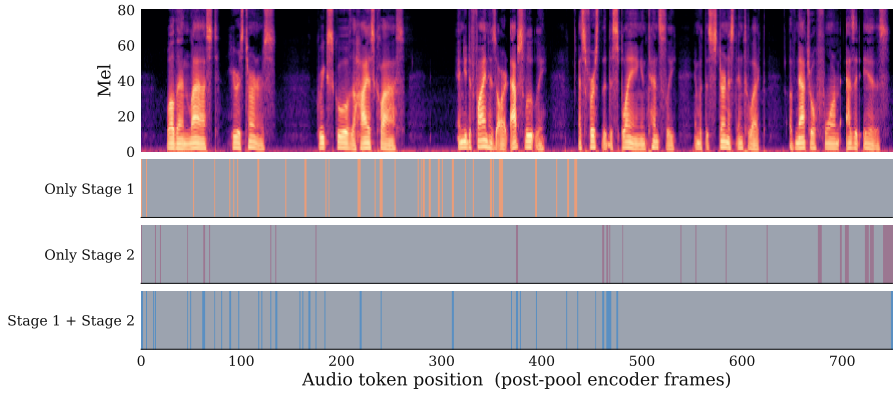
Figure 12 shows token survival at the final pruning layer for three representative LibriSpeech utterances. The examples are consistent with our quantitative finding (§6) that the two stages contribute complementary rather than redundant information: Stage 1 supplies dense, distributed protection against losing phonetic detail, while Stage 2 contributes a smaller number of additional, task-relevant tokens that Stage 1 alone would not have prioritized.



(a) Example 1: *"their piety would be like their names, like their faces, like their clothes… a gift tenfold more acceptable than his elaborate adoration"*



(b) Example 2: *"now you will see in these studies that the moment the white is inclosed properly… the little diamonds in the round window will tell as jewels if they are gradated justly"*



(c) Example 3: *"well then, last, here is turner's greek school of the highest class, and you define his art absolutely as first the displaying intensely and with the sternest intellect of natural form as it is, and then the envelopment of it with cloud and fire"*

Figure 12: Token survival at the final pruning layer for three representative utterances (ground-truth transcripts shown under each panel). Rows within each panel (top to bottom): mel spectrogram, Only Stage 1 (phoneme-transitions only), Only Stage 2 (attention-only), Stage 1 + Stage 2 (attention $\times$ phoneme-transitions, our full method). Orange = kept uniquely by Only Stage 1; mauve = kept uniquely by Only Stage 2; blue = kept uniquely by Stage 1 + Stage 2; grey = pruned.