FastSLM: Hierarchical Temporal Abstraction for Efficient Long-Form Speech Adaptation

Anonymous ACL submission

Abstract

Scaling Multimodal Large Language Models (MLLMs) to long-form speech is bottlenecked by the explosive growth of input tokens. Unlike images or videos, audio lacks overlapping information, making extreme 1-token compression highly susceptible to the loss of fine-grained acoustic cues. To overcome this, we propose FastSLM, a token-efficient architecture featuring the Hierarchical Temporal Abstractor (HTA). HTA progressively distills non-overlapping acoustic features across multiple temporal scales, achieving an extreme compression rate of 1.67 tokens per second (a 97% reduction) without losing critical context. Experimental results show that FastSLM achieves competitive performance with state-of-the-art models on long-form benchmarks despite operating with significantly fewer FLOPs and parameters. The source code and model checkpoints are available at `https://anonymous.4open.science/r/FastSLM-8BD3`.

1 Introduction

Large Language Models (LLMs) (Achiam et al., 2023; Grattafiori et al., 2024; Comanici et al., 2025; Yang et al., 2025) have demonstrated remarkable reasoning capabilities, prompting research into Multimodal LLMs (MLLMs) for vision, audio, and video (Yin et al., 2024; Lyu et al., 2023). Since speech is the primary interface for human-AI interaction (Sakshi et al., 2024), adapting LLMs to the speech domain via Audio Language Models (ALMs) has become a central focus (Tang et al., 2024; Abouelenin et al., 2025; Chu et al., 2023, 2024; Goel et al., 2025).

However, a fundamental challenge hinders ALM scalability: the granularity mismatch between acoustic features and textual tokens. Typical approaches project dense frame-level features from encoders like Whisper (Radford et al., 2023) directly into the LLM input space. For multi-minute audio, this sequence length explodes, rendering the quadratic computational cost of the autoregressive LLM prohibitive.

A critical challenge in extreme speech compression stems from the audio modality's inherent nature. Unlike images or videos with high spatial redundancy, speech is a sequential, non-overlapping stream of localized acoustic events. Attempting to compress this dense stream into a highly reduced token space using flat pooling or single-stage projection inevitably causes catastrophic loss of fine-grained acoustic cues (e.g., phonemes, intonation) necessary for linguistic understanding.

To overcome this, we propose a shift from flat feature alignment to progressive temporal abstraction. We present FastSLM, a token-efficient architecture designed to distill long-form speech into highly compact representations. We introduce the Hierarchical Temporal Abstractor (HTA), which progressively compresses non-overlapping acoustic features from local details into global contexts.

Crucially, this provides a new conceptual insight into speech modality adaptation: an Acoustic-to-Semantic Phase Transition. By structuring compression hierarchically, HTA aligns with human cognitive processing—progressively transitioning from phonemic perception at early stages to semantic understanding at deeper stages. Through this, we achieve an optimal compression rate of 1.67 tokens per second. This rate closely matches typical human speech rates for clear articulation (Tauroza and Allison, 1990; Coupé et al., 2019), effectively establishing a natural information bottleneck that synchronizes continuous audio streams with the discrete semantic processing pace of LLMs.

Our main contributions are summarized as follows:

- **Acoustic-to-Semantic Phase Transition Framework:** We conceptualize speech-LLM alignment not as flat compression, but as a hi-

erarchical abstraction process that maps non-overlapping acoustic frames into discrete semantic units, closely aligning with human utterance rates (1.67 tokens/sec).

- **Hierarchical Temporal Abstractor (HTA):** We introduce a novel architecture that achieves an extreme 97% token reduction compared to frame-level adapters, successfully preserving fine-grained details for long-form reasoning without the catastrophic information loss inherent to single-stage methods.
- **Data-Efficient Adaptation Strategy:** We demonstrate a three-stage training pipeline that aligns pre-trained LLMs with long-form speech using only accessible standard automatic speech recognition (ASR) corpora, circumventing the dependency on scarce long-context instruction data.
- **Empirical Efficiency and Scalability:** FastSLM achieves competitive or superior performance to SOTA models on multi-task benchmarks while exhibiting near-linear memory scaling, enabling efficient processing of hour-long audio on consumer-grade hardware.

2 Related Work

2.1 Audio and Speech Language Models

To advance large language models (LLMs) toward comprehensive multimodal understanding, numerous studies have explored extending them to the audio modality. Models such as AudioPaLM (Rubenstein et al., 2023), Kimi-Audio (Ding et al., 2025), and Qwen-Audio (Chu et al., 2023, 2024) have demonstrated that integrating frame-level speech features into the LLM embedding space enables effective end-to-end spoken understanding.

However, most of these models are primarily trained on short-form speech (typically under 60 seconds), severely limiting their ability to process multi-minute inputs. To address this limitation, recent models like Voxtral (Liu et al., 2025) and Audio-Flamingo3 (Goel et al., 2025) introduced long-context training strategies and frame-level cross-attention for modality adaptation. Despite their strong progress, these approaches still rely on dense frame-level cross-attention or repeated window-level processing. This reliance introduces prohibitive computational overhead when handling multi-minute speech, failing to resolve the fundamental challenge of efficiently aligning long-form speech with LLMs under strict FLOPs constraints.

2.2 Multi-modal Token Compression for Speech Modality Adaptation

To align speech features with an LLM while mitigating the token explosion problem, recent studies have introduced query-based compression modules that convert high-frame-rate speech features into a smaller set of learnable query tokens. For instance, SALMONN (Tang et al., 2024) proposed a window-level compression module that processes speech within fixed temporal windows, while segment-level approaches (Yu et al., 2024) partition the speech sequence into length-based segments before compression.

Existing query-based approaches typically utilize learnable queries to extract static features from fixed, short-term windows or simple segments. However, due to the non-overlapping nature of speech, this design suffers from a severe context-granularity trade-off: applying compression to short windows risks missing the global semantic context, whereas applying it to long segments tends to dilute critical local acoustic details. To resolve this inherent limitation, we propose the HTA, which introduces a progressive abstraction mechanism to efficiently capture both fine-grained local details and broad global context.

3 Methodology

In Section 3.1, we describe the architecture of FastSLM and the inference process. In Section 3.2, we describe the training strategy employed for speech modality adaptation.

3.1 Model Architecture of FastSLM

FastSLM Overview: The overall architecture of FastSLM is illustrated in Fig. 1. Unlike conventional SLMs that directly concatenate dense acoustic features with text embeddings, FastSLM introduces a dedicated compression bottleneck to bridge the information density gap between modalities. The pipeline proceeds as follows: Raw waveforms are converted into Mel spectrograms (25-ms window and a 10-ms stride) and processed by a speech encoder (Radford et al., 2023) to extract high-fidelity frame-level features (50 Hz). Crucially, instead of feeding these dense frames to the LLM, we employ the HTA to distill them into highly compact semantic tokens. This mechanism achieves an extreme compression rate of approx-

imately 1.67 tokens per second, reducing the sequence length by over 97% compared to the original encoder output. These compressed tokens are then concatenated with text prompts, allowing the LLM to perform long-context reasoning with minimal computational overhead.

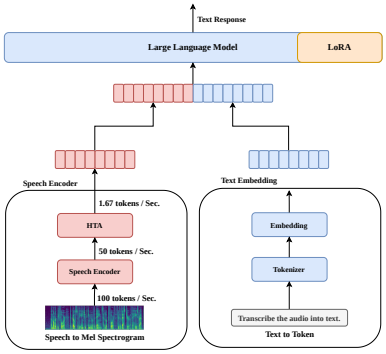


Figure 1: Architecture of FastSLM.

HTA (Hierarchical Temporal Abstractor): The HTA adopts a three-stage hierarchical design to progressively abstract speech information from acoustic to semantic levels. This mimics the cognitive process of progressively summarizing continuous acoustic signals into compact semantic representations. The detailed structure of HTA is shown in Fig. 2.

Stage 1 (Local Acoustic Extraction): The first stage focuses on preserving fine-grained local information. The high-frame-rate features $\hat{\mathbf{X}}_1^A \in \mathbb{R}^F$ from the encoder are fused with a set of learnable queries $\mathbf{Q}_1^A$ via cross-attention. Here, $\mathbf{Q}_1^A$ acts as a local probe, extracting phoneme-level details while maintaining the original temporal resolution. To preserve order during this dense interaction, we inject sinusoidal positional encoding $PE(\cdot)$ (He et al., 2022) into both queries and keys. The compression operation at stage i is defined as:

$$\begin{aligned}\text{Compressor}(\mathbf{Q}_i^A, \hat{\mathbf{X}}_i^A, \hat{\mathbf{X}}_i^A) = \text{Softmax}\left(\frac{\tilde{\mathbf{Q}}_i^A \tilde{\mathbf{X}}_i^{AT}}{\sqrt{d}}\right) \\ \cdot \hat{\mathbf{X}}_i^A, \text{where } \tilde{\mathbf{Q}}_i^A = \mathbf{Q}_i^A + PE(\mathbf{Q}_i^A), \\ \tilde{\mathbf{X}}_i^A = \hat{\mathbf{X}}_i^A + PE(\hat{\mathbf{X}}_i^A),\end{aligned} \tag{1}$$

Stages 2 & 3 (Temporal Aggregation & Semantic Abstraction): Stage 1 captures local acoustic details, while Stages 2 and 3 progressively abstract this information into higher-level semantic representations. A Downsampler acts as a temporal filter, reducing redundancy (e.g., silence, stationary noise). Each Downsampler consists of dual convolutional layers with kernel size 3 and GELU activation (Hendrycks and Gimpel, 2016), applying a stride of 2 to halve temporal resolution at each step. Learnable queries $\mathbf{Q}_i^A$ at deeper stages attend to broader temporal contexts, transitioning from phoneme-level (Stage 1) to word/phrase-level (Stage 2), and finally to sentence-level semantics (Stage 3). The hierarchical feature update is formalized as:

$$\begin{aligned}\hat{\mathbf{Q}}_i^A &= \text{Compressor}_i(\mathbf{Q}_i^A, \hat{\mathbf{X}}_i^A, \hat{\mathbf{X}}_i^A), \\ \text{where} \quad \hat{\mathbf{X}}_i^A &= \text{Downsampler}_i(\hat{\mathbf{X}}_{i-1}^A).\end{aligned} \tag{2}$$

where $i \in \{2, 3\}$, and $\hat{\mathbf{X}}_0^A$ denotes the initial encoder output. Multi-scale representations are concatenated as $\hat{\mathbf{Q}}^A = [\hat{\mathbf{Q}}_1^A; \hat{\mathbf{Q}}_2^A; \hat{\mathbf{Q}}_3^A]$ to form a comprehensive feature bank.

Semantic Distillation (Extreme Token Compression): Although $\hat{\mathbf{Q}}^A$ contains rich information, its length scales linearly with audio duration, which remains suboptimal for long-form reasoning. To achieve our target efficiency of 1.67 tokens/sec, we introduce a final Semantic Distillation step. Drawing inspiration from LLaVA-mini (Zhang et al., 2025), which compresses visual features into a single token, we hypothesize that speech segments can also be represented by a highly compact set of latent variables. However, unlike static images, speech represents a dynamic temporal stream with overlapping events. Therefore, instead of a single token, we employ a small, fixed number of learnable queries $\mathbf{Q}_c^A$ to distill the hierarchical information $\hat{\mathbf{Q}}^A$ into the final compressed tokens $\tilde{\mathbf{Q}}_c^A$:

$$\tilde{\mathbf{Q}}_c^A = \text{C.Attn}\left(\mathbf{Q}_c^A, \hat{\mathbf{Q}}^A, \hat{\mathbf{Q}}^A\right). \tag{3}$$

Detail Recovery Mechanism: Extreme compression risks losing subtle but critical acoustic cues (e.g., intonation, speaker identity). To mitigate this, we incorporate a Detail Recovery attention block. This module allows the compressed semantic tokens $\tilde{\mathbf{Q}}_c^A$ to explicitly re-attend to the original multi-scale acoustic features $\hat{\mathbf{X}}^A$ and retrieve any missing fine-grained details before entering the LLM.

$$\hat{\mathbf{Q}}_c^A = \text{Attention Block}\left(\tilde{\mathbf{Q}}_c^A, \hat{\mathbf{X}}^A\right), \tag{4}$$

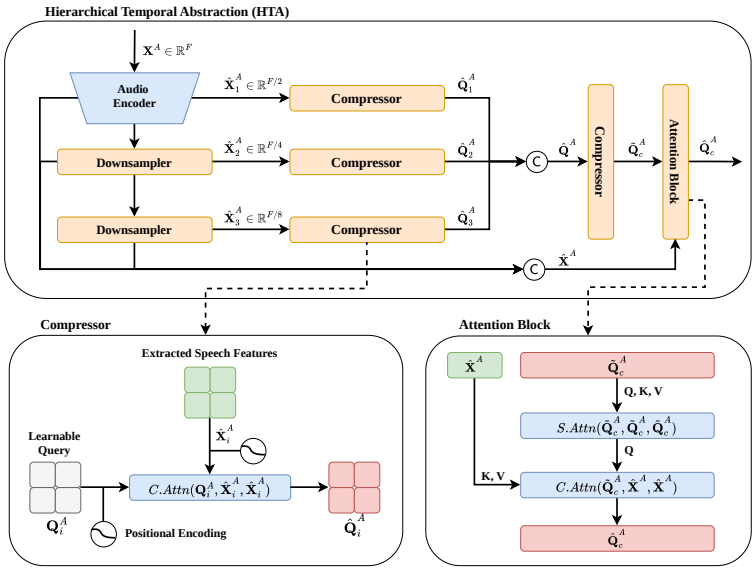


Figure 2: Detailed architecture of the Hierarchical Temporal Abstractor (HTA). C denotes concatenation.

where $\hat{\mathbf{X}}^A = [\hat{\mathbf{X}}_0^A; \hat{\mathbf{X}}_1^A; \hat{\mathbf{X}}_2^A]$. The final output $\hat{\mathbf{Q}}_c^A$ provides a strong balance between semantic density and acoustic fidelity, addressing the granularity mismatch described in Section 1.

To substantiate the efficacy of our hierarchical compression mechanism, we provide a qualitative analysis of cross-attention patterns in Appendix A. The visualizations demonstrate that HTA progressively shifts its attention toward deeper stages as the speech duration increases, confirming that the proposed hierarchical architecture is essential for effective long-range temporal abstraction. Furthermore, we present additional analyses on hierarchical temporal modeling in Appendix B and investigate the impact of the intermediate query size ($\hat{\mathbf{Q}}_i^A$) in Appendix C.

3.2 Three-Stage Training Strategy

To train FastSLM, we propose a three-stage speech modality adaptation strategy, designed to progressively enhance the model's capability to understand and adapt to speech input. Across all stages, we adopt low-rank adaptation (LoRA) (Hu et al., 2022) to ensure cost-efficient training with minimal trainable parameters. Specifically, we set the LoRA rank to 16 and alpha to 64, resulting in a scaling factor of 4.

Pre-training (Short-form Speech Adaptation): In the first stage, the model is trained to adapt to short-form speech inputs. We construct a dataset of approximately 17K hours of speech-text pairs in both Korean and English, with each speech clip restricted to under 30 seconds. This ensures that the model can learn general ASR capabilities and effectively align speech with language. We adopt prompt formats inspired by hierarchical tags (Chu et al., 2023, 2024) to improve language-specific understanding during this stage. A detailed description can be found in Appendix D.

Long-form Speech Adaptation: Pre-trained speech encoders (Gulati et al., 2020; Radford et al., 2023; Elizalde et al., 2023; Chen et al., 2023) are typically limited to processing segments shorter than 30 seconds, hindering performance on long-speech tasks such as speech summarization (SSUM) and spoken query-based question answering (SQQA). While Audio-Flamingo3 (Goel et al., 2025) addressed this by constructing bespoke instruction tuning datasets, such an approach is time-consuming and costly.

To provide a cost-effective alternative, we train the model on a curated ASR-based dataset containing speech–text pairs ranging from 1 to 15 minutes. This stage is designed not to train abstract reasoning directly but to strengthen the model's fundamental ability to process extended speech sequences. Through long-form transcription training, the model learns to maintain temporal coherence and preserve acoustic features over lengthy contexts—a critical prerequisite for complex downstream tasks. The resulting long-context representations supply the LLM backbone with higher-quality, more coherent inputs, enabling superior performance on downstream tasks. Since ASR

datasets are far more accessible than specialized instruction-tuning datasets, this approach offers a scalable path toward long-form speech modeling.

Instruction Tuning: In the final stage, we perform instruction tuning to enable the model to handle a variety of downstream tasks. Due to the scarcity of non-English multi-task datasets, we generated a Korean multi-task dataset using a text-to-speech (TTS) engine (Zhao et al., 2023), covering tasks such as SSUM and SQQA. Unlike previous stages, hierarchical language tags are no longer required as language identification is established; however, hierarchical task tags are employed to explicitly specify the task. Details are provided in Appendix D.

4 Experiment Results

4.1 Dataset Description

Pre-training Dataset: As described in Section 3.2, we constructed a bilingual dataset comprising 17K hours speech-text pairs to adapt ASR capabilities to the LLM during the pre-training stage. Including 9,152 hours of English speech-text pair (LibriSpeech (Panayotov et al., 2015), GigaSpeech-L (Chen et al., 2021), Voxpopuli (Wang et al., 2021), SpgiSpeech-M (O'Neill et al., 2021), Earnings-22 (Rio et al., 2022), AMI (Kraaij et al., 2005), Common Voice 15 (Ardila et al., 2019), AI-HUB ASR-En (The Open AI Dataset Project, 2021)) dataset, and 7,812 hours of Korean speech-text pair (AI-Hub-ASR-Ko (The Open AI Dataset Project, 2021)) dataset. A detailed description of the pre-training dataset can be found in Appendix E Table 7.

Long-form speech Dataset: To enhance the model capacity to process long-form speech input, particularly for tasks such as SSUM and SQQA, we constructed a dedicated long-form speech dataset. For English, we curated a total of 1,224 hours of long-form speech. For Korean, we curated a total of 1,012 hours of long-form speech.

Instruction Tuning Dataset: To enable robust instruction-following capabilities, we constructed a multi-task instruction tuning dataset covering four representative speech-language tasks: ASR, AST, SSUM, and SQQA. For ASR, we randomly sampled Korean and English speech-text pairs from the during pre-training. For SSUM, our dataset includes 1,600 hours of long-form dialogue from the MNSC corpus (Wang et al., 2025). As no public Korean SSUM dataset was available, we synthesized one by applying a TTS engine to the KMSS text summarization dataset (Kim et al., 2022). For SQQA, we used the English LibriSQA dataset (Zhao et al., 2024). For Korean, we constructed a parallel dataset by converting the text-based KorQuAD dataset (Lim et al., 2019) into speech via TTS. A detailed breakdown of datasets used for each instruction tuning task is provided in Table 1.

Evaluation Datasets: To evaluate the speech understanding capabilities of FastSLM, we conducted experiments across a variety of benchmark tasks.

- **ASR**: For English, we used the OpenASR evaluation datasets (Srivastav et al., 2023). For Korean, we used the Common Voice 15 (Ardila et al., 2019) and Fleurs (Conneau et al., 2022) datasets, which are open datasets, for fair comparison of results. We evaluate transcription quality using character error rate (CER) for Korean and word error rate (WER) for English to reflect the linguistic characteristics of each language.

- **AST**: We evaluated translation on the Fleurs (En2Ko/Ko2En) (Conneau et al., 2022) and Minds14 (Ko2En) (Gerz et al., 2021) datasets. We measured translation performance using the BLEU score (Post, 2018).

- **SSUM**: Evaluation was conducted on SDS-PART6 (He et al., 2025) and KMSS-speech (Kim et al., 2022). Summarization quality was assessed using GPT-4 scoring with the LLM-as-a-judge framework (Zheng et al., 2023). To ensure evaluation fairness and prevent prompt engineering bias, we strictly adopted the zero-shot evaluation prompt and scoring rubric proposed in recent work (Abouelenin et al., 2025). Please refer to Appendix F for the exact SSUM judge prompt.

- **SQQA**: We measured accuracy on the LibriSQA and KorQuAD-speech datasets to evaluate SQQA performance.

4.2 Experimental Setup

Model Architecture: FastSLM employs the encoder from Whisper-large-v3 (Radford et al., 2023) for speech feature extraction and adopts Qwen3-4B (Yang et al., 2025) as the backbone LLM for text generation. Despite its relatively compact size, Qwen3-4B exhibits sufficient capacity for comprehending speech-derived representations. In contrast

Table 1: Details of the instruction tuning dataset. “En" denotes English, “Ko" denotes Korean, and “En2Ko", “Ko2En" indicate the translation directions.

Task	Dataset	Duration (hours)	#Samples	speech Language
ASR	LibriSpeech	960	281,241	En
	GigaSpeech-S	250	230,068	En
	AI-HUB ASR	1500	320,000	Ko
AST	AI-HUB AST (En2Ko)	1,209	400,000	En
	AI-HUB AST (Ko2En)	1,152	400,000	Ko
SSUM	SDS-PART6	1,600	103,935	En
	KMSS	668	84,000	Ko
SQQA	LibriSQA	364	104,014	En
	KorQuAD-speech	483	100,243	Ko
Total	-	8,186	2,023,501	-

to prior SLMs that typically utilize LLM backbone with 7 to 14 billion (B) parameters (Chu et al., 2023, 2024; Tang et al., 2024; Yu et al., 2024; Rubenstein et al., 2023; Ding et al., 2025; Liu et al., 2025), FastSLM achieves a favorable cost-performance trade-off by leveraging lightweight architecture without compromising performance (Abouelenin et al., 2025; Ghosh et al., 2025). The HTA module within FastSLM compresses frame-level features via a hierarchical query-based mechanism. The number of learnable queries of $\mathbf{Q}_i^A$ was set to 80 cost-effectively (Yu et al., 2024), and the number of learnable queries of $\mathbf{Q}_c^A$ used as contextual input in LLM was set to 50 through the experiment. For further ablation studies and justifications for parameter selection, please refer to Section 4.5 Fig. 3.

Training: FastSLM was trained on an NVIDIA A100 GPU-80GB$\times$4 with a global batch size of 256. We used mixed precision training (Micikevicius et al., 2017) to maintain model performance while improving computational efficiency, with BF16 used as data type.

The model implementation details and the training setup are summarized in Appendix H.

4.3 Comparison with Baseline Abstractors

To evaluate the performance of HTA, we compared it against two baseline methods: the segment-level Temporal Abstractor (STA) (Yu et al., 2024) and the window-level Temporal Abstractor (WTA) (Tang et al., 2024), which explored speech modality adaptation using temporal windows. We additionally include an average pooling (AvgPool) baseline, in which frame-level speech features are first downsampled through AvgPool and subsequently passed through a multi-layer perceptron (MLP) for linear projection into the LLM embedding space. This baseline allows us to evaluate whether direct downsampling combined with a simple projection layer can effectively replace a more complex, learnable compression module.

For evaluation, we employed the ASR task, as it provides a direct measure of how accurately the model can understand speech content. In addition, to assess the computational load imposed on the LLM when processing long-form speech, we measured the FLOPs of the LLM using a 5-minute speech input. All baselines were trained and evaluated under the same pre-training dataset, LLM backbone, LoRA configuration, and embedding dimensions to ensure a fully fair comparison. The detailed results are presented in Table 2.

Table 2: Comparison of WER, and complexity across baseline methods. LS denotes the LibriSpeech.

Dataset \ Method	AvgPool	STA	WTA	HTA (ours)
LS-clean	1.88	2.32	2.14	2.09
LS-other	4.12	4.87	4.51	4.67
Voxpopuli	7.14	8.37	7.26	6.55
#Speech Tokens/Sec.	25.0	2.67	2.93	1.67
LLM FLOPs (T)	30.6	3.32	3.65	$\mathbf{2.15}(-\mathbf{92}\%)$

As shown in Table 2, HTA achieves the best WER on VoxPopuli and remains highly competitive on LS-clean, while using the fewest speech tokens per second (1.67 tokens/sec). Although AvgPool attains slightly lower WER on the LibriSpeech benchmarks, it requires a vastly larger number of speech tokens (25 tokens/sec) and incurs a substantially higher LLM computation cost (30.6 TFLOPs).

In contrast, HTA achieves a strong balance between accuracy and efficiency: it reduces the token rate by 38% compared to WTA and lowers LLM FLOPs by 41.1% (3.65T $\rightarrow$ 2.15T), while still improving WER on VoxPopuli. These results indicate that HTA provides a significantly more efficient speech-to-LLM alignment mechanism, greatly reducing the computational burden of autoregressive decoding for long-form speech without compromising recognition quality.

Table 3: Comparison of FastSLM with other SLMs on various tasks. Underline indicates the second-best performance. WER and CER are lower-is-better (↓); Accuracy (ACC), BLEU, and Score are higher-is-better (↑). N/A indicates the model lacks the corresponding capability. '*' denotes results fine-tuned on an additional Korean dataset provided in the official Hugging Face supplementary material. Detailed ASR benchmark WER for each dataset is reported in Appendix I.

Task	Metric	Dataset	FastSLM 4.8B	Whisper 1.5B	Qwen2-Audio 8B	Phi4-MM 5.8B	Voxtral-mini 4.7B	Gemini-2.5-Flash
#speech tokens/30 Sec.			50	1500	101	375	750	960
ASR (En)	WER ↓	OpenASR	<u>6.47</u>	7.44	7.43	**6.14**	7.05	9.29
ASR (Ko)	CER ↓	Fleurs Common Voice 15	**3.82**	7.92	N/A	N/A	N/A	<u>4.55</u>
AST (En2Ko)	BLEU ↑	Fleurs	<u>7.39</u>	N/A	N/A	*2.62	N/A	**13.4**
AST (Ko2En)	BLEU ↑	Fleurs	**19.5**	18.6	N/A	*10.4	N/A	<u>19.2</u>
AST (Ko2En)	BLEU ↑	Minds14	<u>28.9</u>	**29.5**	N/A	*14.8	N/A	26.3
SSUM (En)	Score (1-7) ↑	SDS-PART6	5.40	N/A	4.54	5.30	<u>5.48</u>	**5.87**
SSUM (Ko)	Score (1-7) ↑	KMSS	<u>4.12</u>	N/A	N/A	N/A	N/A	**4.37**
SQQA (En)	ACC ↑	LibriSQA	**69.5**	N/A	57.2	64.5	48.9	<u>67.0</u>
SQQA (Ko)	ACC ↑	KorQuAD-speech	**64.9**	N/A	N/A	N/A	N/A	<u>64.8</u>

4.4 Quantitative Results

We primarily compare FastSLM with strong speech-centric baselines such as WhisperV3 and AST, which directly align with our speech-only setting. For completeness, we additionally evaluate several multimodal models (Qwen2-Audio (Chu et al., 2024), Phi-4-Multimodal (MM) (Abouelenin et al., 2025), Gemini-2.5-Flash (Comanici et al., 2025), and Voxtral-Mini (Liu et al., 2025)) in their speech-only mode. These models were not originally designed as SLMs, but we include them for an upper-bound comparison.

FastSLM demonstrates a powerful combination of efficiency and performance, achieving top-tier results with just 50 speech tokens per 30-second input—a fraction of that used by models like Whisper (1,500) and Gemini-2.5-Flash (960). As detailed in Table 3, its key achievements include:

- **ASR**: Achieves a SOTA CER of 3.82 on Korean benchmarks and a competitive WER of 6.47 on English OpenASR.
- **AST**: Achieves SOTA performance on the Fleurs Ko2En task with 19.5 BLEU, outperforming both Whisper and Gemini-2.5-Flash. Furthermore, it demonstrates competitive performance on the Minds14 dataset (28.9 BLEU), comparable to the strong Whisper baseline.
- **SSUM**: Delivers competitive scores of 5.40 (English) and 4.12 (Korean), performing on par with several larger SLMs.
- **SQQA**: Achieves SOTA performance on the evaluated benchmarks, attaining accuracies of 69.5% on LibriSQA (English) and 64.9% on KorQuAD-speech (Korean)

In summary, FastSLM provides a highly effective and efficient solution for diverse speech-language tasks, proving that a compact speech representation can drive SOTA performance.

4.5 Ablation Study

Effect of Speech Token Compression Ratio on ASR Performance: To determine the optimal speech token compression ratio, we conducted an ablation study that evaluates the trade-off between ASR performance (WER) and computational cost. As illustrated in Fig. 3, a clear relationship emerges. While a high token rate (2.67 tokens/sec) yields the best ASR performance, it incurs a substantial computational cost. In contrast, an overly compressed representation (1.33 tokens/sec) results in a significant degradation of performance.

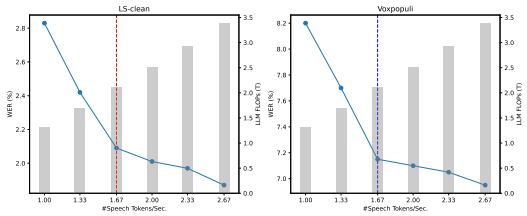


Figure 3: ASR performance of FastSLM with various speech tokens. (left) LS-clean decoding result, and (right) Voxpopuli decoding result.

Our analysis identifies 1.67 tokens/sec as an optimal operating point that balances these competing

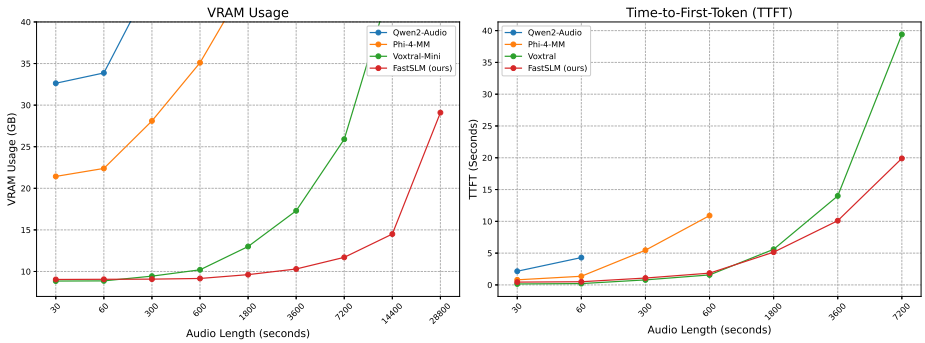


Figure 4: Comparison VRAM usage and time-to-first-token (TTFT) according to speech length.

factors, achieving strong performance while minimizing computational demands. This decision is supported by a mathematical analysis of the point of diminishing returns, detailed in Appendix I.

Scaling Limits of Long-Form Speech Input: To provide a practical assessment beyond indirect metrics like FLOPs (Ma et al., 2018), we empirically evaluated FastSLM's scaling properties by measuring VRAM consumption and time-to-first-token (TTFT) on a single 40GB A100 GPU.

As shown in Fig. 4, FastSLM demonstrates significant scalability advantages. While benchmark models exhibit exponential VRAM growth, FastSLM scales near-linearly, processing an 8-hour speech stream using under 30GB of memory. Furthermore, FastSLM maintains a competitive TTFT that increases only minimally with speech length, avoiding the sharp latency spikes observed in similarly-sized baselines like Voxtral-Mini. These findings confirm that FastSLM enables robust ultra-long-form inference in constrained environments. Additionally, its minimal memory footprint allows for significantly larger batch sizes, maximizing throughput for parallelized workloads.

Impact of Hierarchical Modules and Training Strategy: To validate the efficacy of our architectural design and the three-stage training strategy, we conducted an ablation study by selectively removing key components: the Downsampler stage, the hierarchical attention mechanism, and Training Stage 2 (long-form speech adaptation). As summarized in Table 4, removing the Downsampler or hierarchical attention caused a severe performance degradation, with WER on LS-Long (Park et al., 2024) surging to 12.4 and 10.8, respectively. This underscores the structural necessity of our hierarchical compression in handling long sequences. Furthermore, omitting Training Stage 2, which aligns the model with long-context speech via Long-form speech adaptation, resulted in significant regression across tasks—specifically, increasing LS-Long WER from 5.98 to 6.81 and dropping KorQuAD-Speech accuracy from 64.9% to 59.0%. The full FastSLM configuration consistently outperforms all ablated variants, confirming that both the hierarchical architecture and the dedicated adaptation stage are integral to achieving efficient and robust long-form understanding.

Table 4: Comparison of long-form speech adaptation strategy.

Method \ Dataset	LS-Long WER ↓	KorQuAD-Speech ACC ↑	SDS-PART6 Score (1-7) ↑
w/o Downsample Stage	12.4	56.7	4.12
w/o Hierarchical Attention	10.8	56.9	4.92
w/o Training Stage 2	6.81	59.0	5.07
FastSLM	**5.78**	**64.9**	**5.40**

5 Conclusion

In this paper, we introduced FastSLM, a lightweight architecture designed to overcome the scaling bottlenecks of long-form speech processing. By employing the Hierarchical Temporal Abstractor (HTA) and a cost-effective three-stage training strategy, FastSLM compresses high-frame-rate acoustic features into compact semantic tokens—reducing sequence length by up to 97%. Our results demonstrate that this extreme token compression achieves competitive or state-of-the-art performance across diverse benchmarks, providing a highly scalable and practical building block for real-time multimodal systems.

Limitation

While FastSLM demonstrates significant improvements in inference latency and robust performance across diverse acoustic environments, it has notable limitations. First, due to the lack of public Korean long-form datasets, our Korean instruction-tuning

relied on TTS-generated speech. Although FastSLM achieved SOTA performance on these benchmarks, a synthetic-to-real domain gap may exist when applying these specific reasoning tasks (e.g., Korean SSUM and SQQA) to highly spontaneous, in-the-wild conversational audio. Second, while we empirically validated the near-linear memory scaling of FastSLM for up to 8 hours of continuous speech, comprehensive quantitative evaluations of linguistic robustness (e.g., measuring insertion errors over extreme multi-hour decoding) remain for future work. Finally, FastSLM is currently optimized exclusively for speech-language tasks; extending its hierarchical abstraction to broader auditory scenes (e.g., environmental sounds, music) remains unverified.

Ethical Considerations

Our model was trained exclusively on publicly available ASR corpora and synthetic data, avoiding proprietary or sensitive personal audio. Given its efficiency in processing long-form speech, FastSLM necessitates responsible deployment to prevent potential misuse, such as unauthorized mass surveillance. We acknowledge potential demographic biases stemming from the training data, which may affect performance across diverse accents. Finally, FastSLM's extreme token compression significantly reduces the computational and environmental costs of long-form speech processing, promoting sustainable AI deployment.

References

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, and 1 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, and 1 others. 2021. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. In *Proceedings of the INTERSPEECH*, pages 4376–4380.

Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, Wanxiang Che, Xiangzhan Yu, and Furu Wei. 2023. BEATs: Audio pre-training with acoustic tokenizers. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202, pages 5178–5193.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, and 1 others. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

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

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2022. Fleurs: Few-shot learning evaluation of universal representations of speech. *arXiv preprint arXiv:2205.12446*.

Christophe Coupé, Yoon Mi Oh, Dan Dediu, and François Pellegrino. 2019. Different languages, similar encoding efficiency: Comparable information rates across the human communicative niche. *Science advances*, 5(9):eaaw2594.

Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, and 1 others. 2025. Kimi-audio technical report. *arXiv preprint arXiv:2504.18425*.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2023. Clap learning audio concepts from natural language supervision. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Daniela Gerz, Pei-Hao Su, Razvan Kusztos, Avishek Mondal, Michał Lis, Eshan Singhal, Nikola Mrkšić, Tsung-Hsien Wen, and Ivan Vulić. 2021. Multilingual and cross-lingual intent detection from spoken data. *arXiv preprint arXiv:2104.08524*.

Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Dinesh Manocha, and Bryan Catanzaro. 2025. Audio flamingo 2: An audio-language model with long-audio understanding and expert reasoning abilities. *arXiv preprint arXiv:2503.03983*.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and 1 others. 2025. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *arXiv preprint arXiv:2507.08128*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and 1 others. 2020. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009.

Yingxu He, Zhuohan Liu, Geyu Lin, Shuo Sun, Bin Wang, Wenyu Zhang, Xunlong Zou, Nancy Chen, and Aiti Aw. 2025. Meralion-audiollm: Advancing speech and language understanding for singapore. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 22–30.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceeding of the International Conference on Learning Representations*.

Bong-Su Kim, Hye-Jin Jun, Hyun-Kyu Jeon, Hye-in Jung, and Jung-Hoon Jang. 2022. Kmss: Korean media script dataset for dialogue summarization. In *Annual Conference on Human and Language Technology*, pages 198–204.

Wessel Kraaij, Thomas Hain, Mike Lincoln, and Wilfried Post. 2005. The ami meeting corpus. In *Proceeding of the International Conference on Methods and Techniques in Behavioral Research*, pages 1–4.

Seungyoung Lim, Myungji Kim, and Jooyoul Lee. 2019. Korquad1. 0: Korean qa dataset for machine reading comprehension. *arXiv preprint arXiv:1909.07005*.

Alexander H Liu, Andy Ehrenberg, Andy Lo, Clément Denoix, Corentin Barreau, Guillaume Lample, Jean-Malo Delignon, Khyathi Raghavi Chandu, Patrick von Platen, Pavankumar Reddy Muddireddy, and 1 others. 2025. Voxtral. *arXiv preprint arXiv:2507.13264*.

Chenyang Lyu, Minghao Wu, Longyue Wang, Xinting Huang, Bingshuai Liu, Zefeng Du, Shuming Shi, and Zhaopeng Tu. 2023. Macaw-llm: Multi-modal language modeling with image, audio, video, and text integration. *arXiv preprint arXiv:2306.09093*.

Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and 1 others. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740*.

Patrick K O'Neill, Vitaly Lavrukhin, Somshubra Majumdar, Vahid Noroozi, Yuekai Zhang, Oleksii Kuchaiev, Jagadeesh Balam, Yuliya Dovzhenko, Keenan Freyberg, Michael D Shulman, and 1 others. 2021. Spgispeech: 5,000 hours of transcribed financial audio for fully formatted end-to-end speech recognition. *arXiv preprint arXiv:2104.02014*.

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

Se Jin Park, Julian Salazar, Aren Jansen, Keisuke Kinoshita, Yong Man Ro, and R. J. Skerry-Ryan. 2024. Long-form speech generation with spoken language models. *CoRR*, abs/2412.18603.

Matt Post. 2018. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International conference on machine learning (ICML)*, pages 28492–28518. PMLR.

Miguel Del Rio, Peter Ha, Quinten McNamara, Corey Miller, and Shipra Chandra. 2022. Earnings-22: A practical benchmark for accents in the wild. *arXiv preprint arXiv:2203.15591*.

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, and 1 others. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. 2024. Mmau: A massive multi-task audio understanding and reasoning benchmark. *arXiv preprint arXiv:2410.19168*.

Vaibhav Srivastav, Somshubra Majumdar, Nithin Koluguri, Adel Moumen, Sanchit Gandhi, and 1 others. 2023. Open automatic speech recognition leaderboard. https://huggingface.co/spaces/hf-audio/open_asr_leaderboard.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. 2024. Salmonn: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Steve Tauroza and Desmond Allison. 1990. Speech rates in british english. *Applied linguistics*, 11(1):90–105.

The Open AI Dataset Project. 2021. The open ai dataset project. https://www.aihub.or.kr.

Bin Wang, Xunlong Zou, Shuo Sun, Wenyu Zhang, Yingxu He, Zhuohan Liu, Chengwei Wei, Nancy F Chen, and AiTi Aw. 2025. Advancing singlish understanding: Bridging the gap with datasets and multimodal models. *arXiv preprint arXiv:2501.01034*.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. *arXiv preprint arXiv:2101.00390*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. A survey on multimodal large language models. *National Science Review*, 11(12):nwae403.

Wenyi Yu, Changli Tang, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2024. Connecting speech encoder and large language model for asr. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12637–12641.

Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. 2025. Llava-mini: Efficient image and video large multimodal models with one vision token. *arXiv preprint arXiv:2501.03895*.

Wenliang Zhao, Xumin Yu, and Zengyi Qin. 2023. Melotts: High-quality multilingual multi-accent text-to-speech. https://github.com/myshell-ai/MeloTTS.

Zihan Zhao, Yiyang Jiang, Heyang Liu, Yu Wang, and Yanfeng Wang. 2024. Librisqa: A novel dataset and framework for spoken question answering with large language models. *IEEE Transactions on Artificial Intelligence*, pages 1–12.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Qualitative Analysis of HTA Attention Map

Fig. 5 shows the cross-attention patterns of HTA for short-form (top), mid-form (middle), and long-form (bottom) speech across the three hierarchical stages.

For short-form speech, the final queries attend broadly to all stages, indicating that local and mid-term features remain useful when the sequence is short. For mid-form speech, attention begins to shift away from Stage 1 and is increasingly concentrated on Stage 2 and Stage 3, reflecting the need for broader temporal context. For long-form speech, attention becomes strongly dominated by Stage 3, while Stage 1 and Stage 2 receive minimal attention. This shows that the model relies on high-level, compressed representations for long-form speech reasoning.

Overall, as speech length increases, the attention distribution progressively moves from local (Stage 1) to global (Stage 3) features, demonstrating that HTA adaptively adjusts its focus based on speech length.

B Additional Analysis of Hierarchical Temporal Modeling

To further substantiate the necessity of hierarchical temporal abstraction in HTA, we present additional analyses examining how different stages attend to both local and global speech context and compare HTA against a larger single-stage abstractor operating under the same token budget. Local temporal cues correspond to short-range phonetic transitions, formant dynamics, and prosodic micro-patterns within 50 to 200 ms. In contrast, global temporal cues capture long-range structure such as discourse flow, topic progression, and speaker turns, which are especially important for long-form reasoning tasks including SSUM and SQQA. A single-stage abstractor must simultaneously compress thousands of speech frames, forcing local and global cues to compete within a single attention scale, which leads to representational bottlenecks.

HTA alleviates this limitation through progressive temporal abstraction: early stages attend to fine-grained acoustic details, while deeper stages integrate increasingly broader temporal context. This hierarchical design enables efficient multi-scale modeling without increasing the token budget. Our cross-attention map visualizations (Appendix A) further confirm that attention shifts toward deeper stages as input duration increases, highlighting the importance of multi-stage processing for long-range temporal understanding. The empirical comparison between HTA and the larger single-stage abstractor under the same token budget is presented in Table 5.

Table 5: Comparison between a larger single-stage abstractor and our HTA under the same token budget.

Dataset / Metric \ Method	Larger Single-stage Abstractor	HTA (ours)
LS-clean (WER ↓)	2.11	2.09
Voxpopuli (WER ↓)	7.11	6.55
SDS-PART6-Speech Score (1-7) ↑	5.04	5.40
KorQuAD-Speech ACC ↑	61.2	64.9

Although the ASR quality (LS-clean) is comparable across the two models, HTA achieves substantially higher performance on long-form and reasoning-intensive tasks. These results empirically demonstrate that hierarchical temporal decomposition is critical for robust long-form speech understanding.

C Effect of Hierarchical Staging in HTA

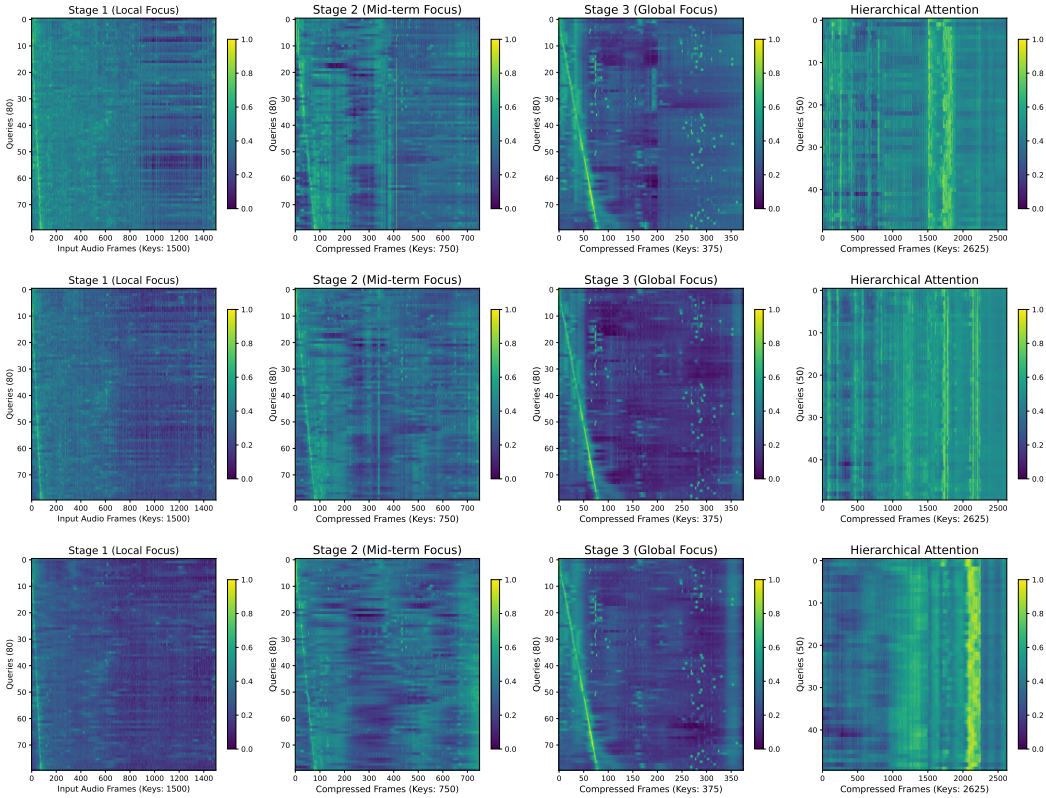
To directly evaluate the benefit of the hierarchical design in HTA, we compare three structural variants: utilizing only Stage 1 (w/o Stage 2/3), utilizing Stages 1 and 2 (w/o Stage 3), and the full hierarchical pipeline (HTA).

Table 6: Effect of hierarchical downsampling stages on speech understanding performance.

Method \ Dataset	LS-clean WER ↓	SDS-PART6-Speech Score (1-7) ↑	KorQuAD-Speech ACC ↑
w/o Stage 2/3	2.23	4.12	56.7
w/o Stage 3	2.15	4.98	62.2
HTA	2.09	5.40	64.9

As shown in Table 6, performance consistently improves across all metrics as more downsampling stages are incorporated. While the foundational speech recognition capability (LS-clean WER) shows a steady reduction from 2.23 to 2.09, the most substantial enhancements are observed in long-form and reasoning-intensive tasks. Specifically, advancing from a single-stage to the full three-stage hierarchy significantly boosts the SSUM score (SDS-PART6-Speech) from 4.12 to 5.40 and increases the SQQA accuracy (KorQuAD-Speech) by an absolute 8.2% (from 56.7% to 64.9%). These empirical results explicitly demonstrate that progressive temporal abstraction is es-

Figure 5: Visualization of HTA attention patterns across speech duration (logarithmic scale). Top: short-form speech ($< 30\,\mathrm{s}$). Middle: mid-form speech ($< 60\,\mathrm{s}$). Bottom: long-form speech ($> 15\,\mathrm{min}$).



sential for effectively capturing the global semantic context required in long-form speech processing.

D Prompt Template and Hierarchical Tags for Training

We applied a unified prompt template and task/language control tokens during training to support multiple speech-language tasks:

$$\text{User}:< |\text{audio_bos}| >< |\text{AUDIO}| >< |\text{audio_eos}| > \{\text{Prompt/Question}\}\backslash\text{n Assistant}:$$

To improve task specialization and language awareness, we employed hierarchical task and language tokens, which enabled robust detection and performance across languages and tasks:

$$\text{Language Token}:< |\text{KO}| >< |\text{EN}| >$$
$$\text{Task Token}:< |\text{ASR}| >< |\text{AST}| >$$
$$< |\text{SQQA}| >< |\text{SSUM}| >$$

E Pre-training Dataset

Table 7 is a dataset used for pre-training of Fast-SLM. The dataset consists of English and Korean, and consists of a total of 10M speech-text pairs.

F Prompt for GPT-4 as a Judge on Speech Benchmarks

The following is the exact prompt template used for evaluating SSUM output via an LLM-as-a-Judge.

Listing 1: LLM-as-a-Judge Prompt for SSUM Evaluation

```
You are a skilled evaluator for
    summaries generated based on user-
    provided instructions.
Your task is to rate how well the
    summary follows the user's
    instructions on a 1-7 scale.

Scoring Rubric:
- 7 (Excellent): Fully follows all
    instructions. Accurate, fluent, and
    coherent with the correct level of
    detail and structure.
- 6 (Good): Almost perfect, with very
    minor issues that do not affect
    usability (e.g., tiny structural
    deviation, trivial omission).
- 5 (Mostly Correct): Fulfills the main
    instruction but has noticeable
    issues (e.g., includes some
    unimportant extras, misses a few
    details).
- 4 (Acceptable): Adheres to the
    instruction partially but has
    significant issues like
    inconsistencies or irrelevant
```

Table 7: Pre-training dataset details. En denote the English, and Ko denote the Korean.

Dataset	Duration (hours)	# Samples	Speech Language
LibriSpeech	960	281,241	En
TED-LIUM-release3	454	268,263	En
GigaSpeech-L	2,500	2,266,371	En
Voxpopuli	523	182,482	En
SpgiSpeech-M	1,000	385,361	En
Earnings-22	105	52,006	En
AMI	78	108,502	En
Common Voice 15	2,532	1,070,066	En
AI-HUB ASR-En	1,000	1,020,265	En
AI-HUB ASR-Ko	7,812	4,557,512	Ko
Total	16,964	10,212,348	-

```
    content.
- 3 (Poor): Minimally adheres to the
    instruction, missing most required
    details or containing significant
    irrelevant/hallucinated content.
- 2 (Very Poor): Fails to follow the
    core instruction. Mostly irrelevant,
     fabricated, or ignores requested
    structure/tone.
- 1 (Fails): Completely fails to follow
    instructions.

Input:
- User Instruction: {USER_INSTRUCTION}
- Reference (gold): {REFERENCE_ANSWER}
- Model Summarization: {SUMMARY_TO_
    EVALUATE}

Notes:
- It helps to read the Summary first,
    then compare with the Reference and
    Instruction.
- If the summary is missing or empty,
    return N/A as the score.

Output:
Note: Use the following JSON format for
    easy downstream consumption.
{
    explanation: Brief reasoning for the
         score based on the rubric.,
    score: <Float, 1-7>
}
```

G Generation and Decoding Configuration

We use the following decoding hyperparameters for all LLM-based generation tasks.

Table 8: Decoding configuration used for LLM-based generation in FastSLM.

Parameter	Value
Decoding Strategy	Sampling
Temperature	0.2
Top-p	0.95
Top-k	20
Repetition Penalty	1.0

H Model and Training Parameters

The model implementation details and the training setup for each stage are presented in Table 9 and Table 10.

Table 9: Model configuration for FastSLM.

Module	Component	Configuration
Encoder	Backbone	Whisper-large-v3
	Parameters	635M
	Hidden Size	1280
	Context Length	1500
Abstractor	Backbone	HTA
	Parameters	56M
	Hidden Size	1280
	Queries per Stage	80
	Compressed Speech Token	50
	Downsampling Factors	2
LLM	Backbone	Qwen3-4B
	Parameters	4.06B
	Hidden Size	2560
	Context Length	4096
LoRA	Rank (r)	16
	Alpha (α)	64
	Scaling Factor	4
	LoRA Target Modules	q/k/v_proj, gate/up/down_proj

Table 10: Training settings across stages

Setting	Stage1	Stage2	Stage3
Learning Rate	1e-4	5e-5	5e-5
Learning Rate Scheduler	Linear Decay		
Weight Decay	0	1e-4	1e-4
Epoch	1	1	2
Data Type	BF16		
DeepSpeed Stage	Zero2		

I Information-Theoretic Justification for the Compression Bottleneck

To rigorously justify our optimal compression rate of 1.67 speech tokens/sec, we analyze the capacity of the HTA module through the lens of information bottleneck theory. We quantify the trade-off between reasoning performance and computational cost by defining the *Marginal Semantic Utility* (η). This metric measures the effective semantic gain per unit of computational overhead:

$$\eta(r) = \left| \frac{\partial \text{WER}(r)}{\partial \text{FLOPs}(r)} \right| \approx \frac{\Delta \text{WER}}{\Delta \text{FLOPs}}$$

where r represents the speech token rate. We apply this metric to the LS-clean decoding results, as summarized in Table 11.

The analysis in Table 11 reveals a precipitous drop in utility—from 1.00 to 0.08—when expanding the token rate beyond 1.67 tokens/sec. This non-linear degradation quantitatively demonstrates that the interval beyond 1.67 tokens/sec marks a stark point of diminishing returns, acting as a **critical saturation point** for the acoustic-to-semantic phase transition.

In the $1.33 \rightarrow 1.67$ interval, additional tokens actively resolve linguistic ambiguities (e.g., recovering missing words), yielding high utility. Conversely, beyond the 1.67 tokens/sec bottleneck, the additional tokens no longer contribute to semantic understanding. Instead, they merely encode redundant acoustic variations (e.g., micro-pauses, pitch fluctuations) that quadratically inflate the LLM's computational cost without improving reasoning.

Crucially, this empirical saturation point theoretically aligns with the optimal human speaking rate of 100 to 120 words per minute (approximately 1.67 to 2.0 words per second) observed in spontaneous and clear speech (Tauroza and Allison, 1990). This confirms that our chosen token rate is not an arbitrary threshold, but rather acts as a natural information bottleneck (Coupé et al., 2019), effectively synchronizing the continuous acoustic stream with the discrete semantic processing pace of the LLM.

J Details of ASR Benchmark Results

Table 12 presents a detailed comparison of FastSLM and SOTA models across multiple ASR benchmarks. We report WER for English datasets and CER for Korean datasets. The results demonstrate that FastSLM achieves competitive performance while using significantly fewer speech tokens per second.

Table 11: Marginal Semantic Utility (η) across different temporal compression intervals. A sharp drop in η indicates the saturation of semantic information.

Token Rate Interval (r)	ΔWER (Reduction)	ΔFLOPS (Increase)	Marginal Semantic Utility (η) $\uparrow$
$1.33 \rightarrow 1.67$	$\approx 0.3\%$	≈ 0.3 T	≈ 1.00
$1.67 \rightarrow 2.00$	$\approx 0.1\%$	≈ 1.3 T	≈ 0.08

Table 12: Comparison of WER between FastSLM and state-of-the-art (SOTA) models. This results representation ASR Benchmark Dataset WER and CER.

Dataset	Sub-Category	Metric	FastSLM 4.8B	Qwen2-Audio 8B	Phi4-Multimodal 5.8B	Whisper 1.5B	Voxtral-mini 4.7B	Gemini-2.5-Flash
OpenASR	AMI	WER	10.8	15.2	11.7	16.0	16.3	21.6
	Earnings22	WER	10.7	14.1	10.2	11.3	10.7	13.1
	GigaSpeech	WER	10.7	10.3	9.78	10.0	10.2	10.7
	SpgiSpeech	WER	2.33	3.00	3.13	2.01	2.37	3.82
	TEDLIUM	WER	3.97	4.05	2.90	3.91	3.68	3.01
	LS-clean	WER	2.09	1.74	1.68	2.94	1.88	2.49
	LS-other	WER	4.67	4.03	3.83	3.86	4.10	5.84
	Voxpopuli	WER	6.55	7.05	5.91	9.54	7.14	7.89
Fleurs	En	WER	5.26	5.27	3.38	4.10	3.77	6.20
	Ko	CER	2.79	N/A	N/A	5.32	N/A	3.00
Common Voice 15	En	WER	10.9	8.68	7.61	9.30	10.2	11.2
	Ko	CER	4.55	N/A	N/A	5.74	N/A	6.09