

MoST: MIXING SPEECH AND TEXT WITH MODALITY-AWARE MIXTURE OF EXPERTS

Anonymous authors

Paper under double-blind review

ABSTRACT

We present MoST (Mixture of Speech and Text), a novel multimodal large language model that seamlessly integrates speech and text processing through our proposed Modality-Aware Mixture of Experts (MAMoE) architecture. While current multimodal models typically process diverse modality representations with identical parameters—disregarding their inherent representational differences, we introduce specialized routing pathways that direct tokens to modality-appropriate experts based on input type. MAMoE simultaneously enhances modality-specific learning and cross-modal understanding through two complementary components: modality-specific expert groups that capture domain-specific patterns and shared experts that facilitate information transfer between modalities. Building on this architecture, we develop an efficient transformation pipeline that adapts the pretrained MoE language model through strategic post-training on ASR and TTS datasets, followed by fine-tuning with a carefully curated speech-text instruction dataset. A key feature of this pipeline is that it relies exclusively on fully accessible, open-source datasets to achieve strong performance and data efficiency. Comprehensive evaluations across ASR, TTS, audio language modeling, and spoken question answering benchmarks show that MoST consistently outperforms existing models of comparable parameter counts. Our ablation studies confirm that the modality-specific routing mechanism and shared experts design significantly contribute to performance gains across all tested domains. To our knowledge, MoST represents the first fully open-source speech-text LLM built on a Mixture of Experts architecture.¹

1 INTRODUCTION

The seamless integration of speech and text modalities represents a cornerstone objective in the pursuit of versatile and natural human-computer interaction. Foundation models, particularly Large Language Models (LLMs), have demonstrated remarkable capabilities in text understanding and generation (OpenAI et al., 2024; DeepSeek-AI et al., 2024), motivating efforts to extend their prowess to encompass spoken language. However, unifying inherently different modalities like continuous, high-dimensional speech waveforms and discrete, symbolic text sequences within a single model presents significant architectural and training challenges.

A prevalent approach in multimodal learning involves mapping different modalities into a shared representational space, often processed by a common set of transformer parameters (Nguyen et al., 2024; Défossez et al., 2024). While conceptually simple, forcing diverse modality representations through identical parameters disregards their inherent representational differences. Specifically for speech and text, this uniform treatment risks representational interference, where the distinct statistical properties and structural nuances of each modality are diluted. Furthermore, it can hinder the development of highly specialized processing pathways necessary to capture the fine-grained details unique to each domain—such as phonetic variations in speech or complex semantic relations in text.

Mixture of Experts (MoE) architectures have emerged as a powerful technique for scaling model capacity while maintaining computational efficiency (Shazeer et al., 2017; Lepikhin et al., 2020;

¹We release MoST model, training code, inference code, and training data at <https://github.com/anonymous29008/MoST/tree/main>

Fedus et al., 2022). By sparsely activating only a subset of parameters (experts) for each input token, MoE models can significantly increase their parameter count without a proportional increase in inference cost. While promising for handling complex data, standard MoE routing mechanisms are typically modality-agnostic; tokens are routed based on learned gating functions over their representations, without explicit consideration of their originating modality. This leaves untapped the potential to dedicate specialized expert capacity based on known input characteristics.

In this work, we propose MoST (Mixture of Speech and Text), a novel speech-text LLM built upon a Modality-Aware Mixture of Experts (MAMoE) architecture. MAMoE directly addresses the limitations of uniform parameter processing by incorporating two principal components: **modality-specific expert groups** (e.g., text-only, audio-only) governed by a modality-aware routing mechanism, and **shared experts** accessible to all modalities, which facilitate cross-modal interaction. The modality-aware routing explicitly leverages token modality information to direct inputs towards appropriate modality-specific experts. Text tokens are primarily routed to text and shared experts, while audio tokens target audio and shared experts. This design allows for: (1) **Specialized Learning**: Modality-specific experts can capture the intricate patterns unique to either speech or text without interference. (2) **Efficient Cross-Modal Interaction**: Shared experts act as a bridge, facilitating knowledge transfer and cross-modal alignment crucial for tasks like Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) synthesis.

To effectively realize MoST, we develop an efficient LLM to Speech-Text Transformation Pipeline. Starting from a pretrained MoE LLM, we first conduct targeted post-training on large ASR and TTS datasets. This stage primes the modality-specific experts and adapts the shared experts for cross-modal tasks. Subsequently, we perform instruction fine-tuning using a carefully curated dataset comprising diverse speech and text instructions, enhancing the model’s versatility and controllability. A key aspect of our approach is the exclusive use of fully accessible, open-source training data to achieve strong performance. This strategic choice distinguishes MoST from some prominent speech-text models, such as SpiritLM (Nguyen et al., 2024) and Moshi (Défossez et al., 2024), whose development incorporated datasets that are not publicly accessible.

Comprehensive evaluations across a range of speech-text benchmarks, including Automatic Speech Recognition (ASR), Text-to-Speech (TTS) synthesis, audio language modeling, and spoken question answering, demonstrate that MoST achieves state-of-the-art or competitive performance compared to existing models with similar parameter counts. Our ablation studies rigorously validate the effectiveness of the MAMoE architecture, confirming that the modality-aware routing mechanism is a key contributor to performance gains across both modality-specific and cross-modal tasks. To the best of our knowledge, MoST is the first fully open-source speech-text LLM leveraging a Mixture of Experts architecture with modality-specific routing.

Our main contributions are:

- **Modality-Aware Mixture of Experts (MAMoE)**: A novel multimodal MoE architecture that directs tokens to modality-specific expert groups and shared experts, enhancing specialized learning and cross-modal interaction.
- **Efficient Text-Speech Transformation Pipeline**: A data-efficient pipeline for adapting a pretrained LLM into a speech-text LLM via targeted post-training and instruction tuning.
- **Open Release of MoST Model**: We release the MoST model weights, training and inference code, and training dataset to facilitate future research and development in multimodal AI.

2 RELATED WORK

End-to-end Speech-Text Modeling. Existing end-to-end speech-text models like Qwen2-Audio (Chu et al., 2024) (text output only) and AudioLM (Borsos et al., 2023) (no direct text integration) have specific limitations. Others such as SpiritLM (Nguyen et al., 2024) and Moshi (Défossez et al., 2024) employ dense transformers, which process all tokens uniformly, unlike MoST’s Modality-Aware Mixture of Experts (MAMoE), which uses specialized and shared experts for modality-specific processing and cross-modal understanding.

Mixture of Experts. Standard Mixture of Experts (MoE) architectures (Lepikhin et al., 2020; Fedus et al., 2022; Du et al., 2022) scale models efficiently but typically feature modality-agnostic routing,

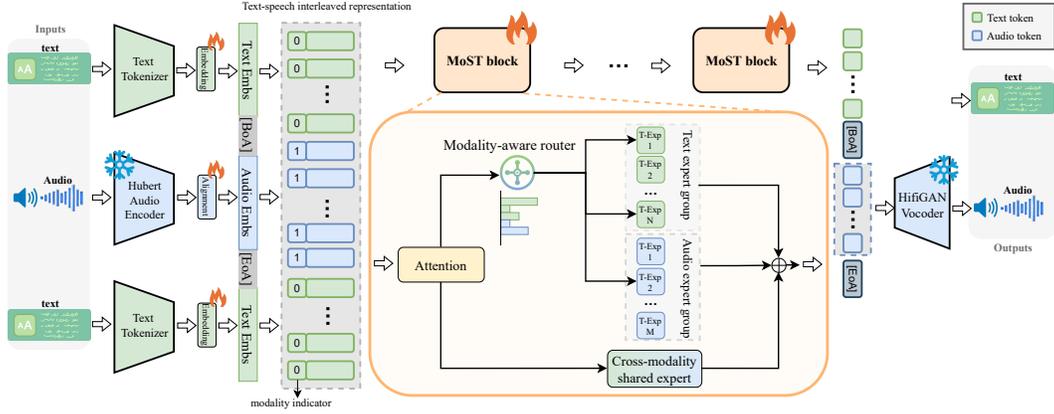


Figure 1: MoST overall architecture. MoST supports interleaved text and audio as input and can generate both text and audio. MoST blocks are transformer blocks featured with Modality-Aware Mixture of Experts (MAMoE). MAMoE is comprised of three critical components: modality-specific expert groups, cross-modal shared expert, and the modality-aware router. Modality-aware router utilizes the modality indicator information for modality-specific routing.

limiting their multimodal effectiveness. While some research adapts MoE for general multimodal tasks (e.g., image-text) by grouping experts (Lin et al., 2024; Shen et al., 2024), MoST is the first to introduce MAMoE specifically for speech-text. Another key difference is that MoST’s MAMoE employs shared experts for cross-modal interaction, which is novel.

Multimodal Adaptation from LLMs. LLMs are often adapted for multimodality via fine-tuning (Zhang et al., 2024), instruction tuning (Liu et al., 2023), or speech-specific methods (Fathullah et al., 2023). MoST presents a unique transformation pipeline: starting from a pretrained MoE LLM, it undergoes targeted ASR/TTS post-training, followed by fine-tuning with a curated speech-text instruction dataset. This approach offers significant data efficiency over conventional adaptation strategies.

3 MOST: MIXTURE OF SPEECH AND TEXT

3.1 MODEL OVERVIEW

MoST (Mixture of Speech and Text) is a unified foundation model built on a Mixture of Experts (MoE) architecture for seamless speech and text processing. Unlike discrete tokenization approaches (Borsos et al., 2023; Nguyen et al., 2024), MoST directly processes continuous audio waveforms via audio encoder E_{audio} : $\mathbf{H}_{audio} = E_{audio}(\mathbf{A}) \in \mathbb{R}^{L_{audio} \times d_{model}}$. Text representations are obtained through standard tokenization and embedding: $\mathbf{H}_{text} = E_{text}(\mathbf{T}) \in \mathbb{R}^{L_{text} \times d_{model}}$. These modalities are combined into unified input \mathbf{H}_{input} .

The core transformer decoder D_{MoST} , adapted from a pretrained MoE LLM, incorporates our Modality-Aware Mixture of Experts (MAMoE) in designated layers. MAMoE routes tokens to modality-specific experts and shared experts for cross-modal interaction: $\mathbf{H}_{output} = D_{MoST}(\mathbf{H}_{input})$.

MoST generates both text via language model head P_{text} and audio via vocoder G_{audio} conditioned on \mathbf{H}_{output} and speaker embeddings, enabling diverse end-to-end speech-text tasks. Figure 1 illustrates the architecture.

3.2 AUDIO ENCODER AND DECODER

MoST processes continuous audio waveforms directly, preserving richer acoustic information than discrete tokenization methods (Borsos et al., 2023; Nguyen et al., 2024).

Audio Encoder Input waveforms \mathbf{A} are encoded using frozen HuBERT (Hsu et al., 2021), yielding features $\mathbf{F}_{hubert} \in \mathbb{R}^{L_{audio} \times d_{hubert}}$, then projected to model dimension:

$$\mathbf{H}_{audio} = \mathbf{F}_{hubert} \mathbf{W}_{proj} \in \mathbb{R}^{L_{audio} \times d_{model}}. \quad (1)$$

These continuous embeddings replace placeholder tokens in the input sequence.

Audio Decoder For synthesis, MoST employs HifiGAN vocoder G_{audio} (Kong et al., 2020), generating waveforms $\hat{\mathbf{A}}$ conditioned on predicted HuBERT tokens $\mathbf{C}_{audio} = P_{hubert_tokens}(\mathbf{H}_{output})$ and speaker embeddings \mathbf{s} :

$$\hat{\mathbf{A}} = G_{audio}(\mathbf{C}_{audio}, \mathbf{s}). \quad (2)$$

Algorithm 1 Modality-Aware Expert Routing

Require: Token representation \mathbf{h}_t , modality indicator \mathbf{m}_t , gating weights \mathbf{W}_g , expert groups \mathcal{E}_{text} , \mathcal{E}_{audio} , number of experts to select K

Ensure: Routed output $\mathbf{y}_{routed,t}$

```

1: Step 1: Compute Raw Gating Scores
2:  $\mathbf{s}_t \leftarrow \text{softmax}(\mathbf{h}_t \mathbf{W}_g) \in \mathbb{R}^N$  ▷ Raw scores over all  $N$  experts
3: Step 2: Create Modality-Specific Mask
4: for  $i = 1$  to  $N$  do
5:   if  $m_t = 0$  (text) and  $E_i \in \mathcal{E}_{text}$  then
6:      $M_{m_t,i} \leftarrow 1$ 
7:   else if  $m_t = 1$  (audio) and  $E_i \in \mathcal{E}_{audio}$  then
8:      $M_{m_t,i} \leftarrow 1$ 
9:   else
10:     $M_{m_t,i} \leftarrow 0$ 
11:   end if
12: end for
13: Step 3: Apply Modality Mask
14:  $\mathbf{s}'_t \leftarrow \mathbf{s}_t \odot \mathbf{M}_{m_t}$  ▷ Element-wise multiplication
15: Step 4: Select Top-K Experts
16:  $\mathcal{I}_t \leftarrow \text{TopK}(\mathbf{s}'_t, K)$  ▷ Indices of top-K experts
17:  $\mathbf{w}'_t \leftarrow \{\mathbf{s}'_t[i] : i \in \mathcal{I}_t\}$  ▷ Corresponding weights
18: Step 5: Compute Routed Output
19:  $\mathbf{y}_{routed,t} \leftarrow \sum_{k=1}^K w'_{t,k} \cdot E_{i_t,k}(\mathbf{h}_t)$  ▷ Weighted expert outputs
20: return  $\mathbf{y}_{routed,t}$ 

```

3.3 MODALITY-AWARE MIXTURE OF EXPERTS (MAMOE)

3.3.1 MODALITY-SPECIFIC EXPERT GROUPS

In the MAMoE layers, the set of N available routed experts, denoted $\mathcal{E} = \{E_1, E_2, \dots, E_N\}$, is partitioned into disjoint modality-specific groups. Based on the primary modalities in MoST (speech and text), we define: **Text Expert Group** (\mathcal{E}_{text}), a subset of experts designated to primarily process text-derived token representations, and **Audio Expert Group** (\mathcal{E}_{audio}), a subset of experts designated to primarily process audio-derived token representations (specifically, features derived from the HuBERT encoder).

These sets are disjoint, $\mathcal{E}_{text} \cap \mathcal{E}_{audio} = \emptyset$, and their union constitutes the full set of routed experts, $\mathcal{E}_{text} \cup \mathcal{E}_{audio} = \mathcal{E}$. Each expert E_i is an MLP, consisting of gate, up, and down projection layers. This partitioning allows experts to specialize in capturing the distinct statistical patterns and representations characteristic of either speech or text.

3.3.2 CROSS-MODAL SHARED EXPERTS

While modality-specific experts promote specialized processing, effective multimodal understanding requires mechanisms for information integration across modalities. In MoST, this is facilitated by incorporating shared experts, implemented as a separate, parallel MLP block (E_{shared}). Unlike the routed experts \mathcal{E} , the shared expert block processes *all* tokens passing through the MAMoE layer.

Given the input hidden state \mathbf{h} to the MAMoE layer, the output \mathbf{y}_{routed} from the modality-specific routed experts (detailed in Sec 3.3.3) is combined with the output of the shared expert block:

$$\mathbf{y}_{mamoE} = \mathbf{y}_{routed} + E_{shared}(\mathbf{h}). \tag{3}$$

This parallel shared expert acts as a common pathway, allowing for the exchange and integration of information learned from different modalities, complementing the specialized processing occurring within the modality-specific expert groups.

3.3.3 MODALITY-AWARE ROUTER

MAMoE’s core routing mechanism directs each token \mathbf{h}_t to modality-specific experts $E_i \in \mathcal{E}$ based on its content and modality indicator m_t (e.g., $m_t = 0$ for text, $m_t = 1$ for audio). Algorithm 1 presents the step-by-step procedure for computing modality-aware routing scores.

During training, an auxiliary load balancing loss, based on \mathbf{s}'_t , promotes balanced expert utilization within each modality group. This modality-aware routing directs tokens to specialized pathways, enhancing expert capacity use and modality-specific representation learning.

4 EFFICIENT TRAINING RECIPE OF MOST

We develop an efficient training recipe to transform a MoE LLM into MoST. Our approach consists of carefully designed data preparation procedures followed by a two-stage training protocol. Comprehensive details of all procedures are provided in Appendix A.

4.1 DATA PREPARATION

ASR and TTS Data Curation. We curate large-scale ASR and TTS datasets from open-source resources, primarily utilizing Common Voice (Ardila et al., 2020b) and LibriHeavy (Kang et al., 2024). For ASR data, we use the standard format with speech input and text transcription output. For TTS data, we reverse the modalities to create paired speech-text samples.

Speech-Text Instruction Dataset Construction. We construct a speech-text instruction-following dataset through two complementary strategies: (1) **Interrupted Dialogue Synthesis:** We augment existing dialogue datasets (Allal et al., 2025) with realistic interruptions using LLM-based synthesis to capture natural conversational dynamics and turn-taking patterns. (2) **Text-to-Speech Instruction Conversion:** We convert text-based instruction datasets into speech formats using our intermediate TTS-trained MoST, significantly expanding speech instruction data without manual collection.

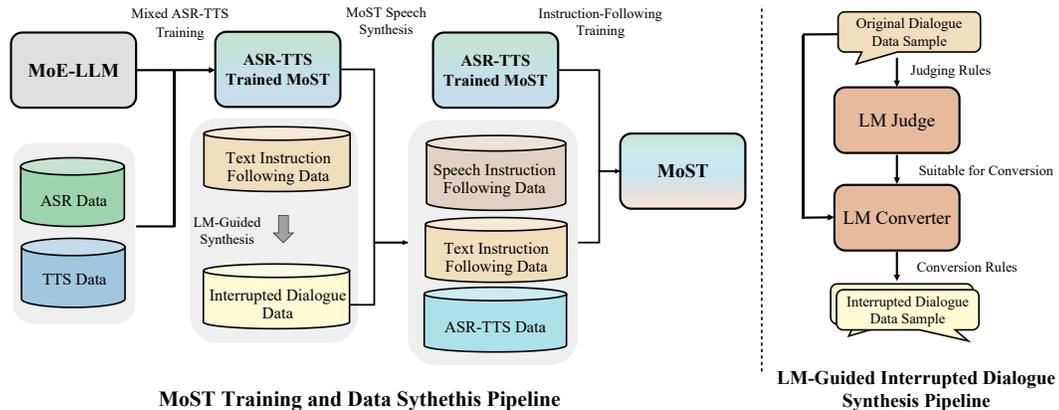


Figure 2: MoST training recipe overview. The left panel illustrates the systematic transformation pipeline from MoE LLM to MoST, encompassing data preparation and two-stage training protocol. The right panel details the interrupted dialogue synthesis process for instruction dataset construction.

4.2 TRAINING PROTOCOL

Stage 1: Cross-Modal Post-Training. We initialize MoST from DeepSeek-v2 Lite (DeepSeek-AI et al., 2024), a pretrained MoE LLM. The initialized model undergoes targeted post-training on the curated ASR and TTS datasets. This stage serves dual purposes: priming modality-specific experts within MAMoE for speech processing and adapting shared experts for fundamental cross-modal alignment between speech and text representations.

Stage 2: Mixed Instruction Fine-Tuning. The model is fine-tuned using our constructed speech-text instruction dataset, strategically mixed with a substantial portion of ASR/TTS data from Stage 1. This mixed-task learning approach enhances complex instruction-following capabilities while preventing catastrophic forgetting of foundational speech processing abilities through continual exposure to core tasks.

5 EMPIRICAL RESULTS

5.1 EVALUATION SETTINGS

To rigorously assess the performance of MoST, we establish a comprehensive evaluation framework encompassing standard benchmarks and strong baseline models.

Baselines. We compare MoST against several state-of-the-art open-source speech and text models, including **AudioLM** (Borsos et al., 2023), a prominent model for high-fidelity audio generation; **SpeechGPT** (Zhang et al., 2023), an LLM adapted for speech tasks; **SpiritLM** (Nguyen et al., 2024), a recent speech-language model with strong performance; **Moshi** (Défossez et al., 2024), another relevant model in speech processing; **Qwen2-Audio** (Chu et al., 2024), a powerful open-source model which has shown strong performance in various speech tasks; **Phi-4 Multimodal** (Microsoft et al., 2025), a compact yet powerful multimodal language model that achieves very strong ASR and AST performance; **SeamlessM4T-v2** (Communication et al., 2023), an advanced model for speech translation and other speech-related tasks; **MinMo** (Chen et al., 2025), a recent multimodal model for speech and text; and **LLaMA-Omni2** (Fang et al., 2025), a recent omnimodal language model with speech capabilities.

Benchmarks and Metrics. Evaluation spans ASR, TTS, audio language modeling (ALM), and spoken question answering (SQA). For ASR/TTS, we use LibriSpeech-clean and -other (Kang et al., 2024), VoxPopuli-V1.0-en (Wang et al., 2021), and Common Voice 15-en (WER for ASR, TTS; lower is better) (Ardila et al., 2020a). ALM is assessed on sWUGGY, sBLIMP (Both from Nguyen et al. (2020)); sTopic-StoryCloze and sStoryCloze (Both from Hassid et al. (2024)) (accuracy). SQA uses Llama Q (Nachmani et al., 2024) (S→T, S→S), Trivial QA (Joshi et al., 2017) (S→T, S→S), and WebQ (Berant et al., 2013) (S→T, S→S) (task-specific scores/accuracy).

5.2 EVALUATION ON TTS AND ASR TASKS

We evaluate MoST on ASR and TTS across multiple datasets in Table 1. MoST demonstrates competitive ASR performance, achieving 2.0% WER on LS-Clean and 3.7% WER on LS-Other, while excelling in challenging cross-dataset scenarios with 6.2% WER on VoxPopuli-V1.0-en and 8.4% WER on Common Voice 15-en. In TTS synthesis, MoST achieves state-of-the-art performance across multiple datasets: 6.0% WER on LS-Clean, 10.1% CER on VoxPopuli-V1.0-en, and 11.5% CER on Common Voice 15-en, consistently outperforming all baselines including the newly added MinMo and LLaMA-Omni2. These results highlight MAMoE’s effectiveness in cross-modal tasks and our pipeline’s role in creating a versatile speech-text model with strong generalization capabilities.

5.3 EVALUATION ON AUDIO LANGUAGE MODELING TASKS

MoST and baseline models are evaluated on four audio language modeling benchmarks—sWUGGY, sBLIMP (Lakhotia et al., 2021), sTopic-StoryCloze, and sStoryCloze (Hassid et al., 2023)—designed to probe audio language modeling capabilities such as grammatical acceptability and discourse coherence. We report accuracies derived from negative log-likelihood (NLL) scores normalized by sequence length, ensuring a fair comparison across variable-length inputs and tasks.

Table 1: ASR (WER % ↓) and TTS (WER % ↓) performance comparison on LibriSpeech clean and other test sets, VoxPopuli-V1.0-en, and Common Voice 15-en. Lower values are better. **MoST (Ours)** results are highlighted with green background.

Model	LS-Clean		LS-Other		VoxPopuli-V1.0-en		Common Voice 15-en	
	ASR	TTS	ASR	TTS	ASR	TTS	ASR	TTS
SpeechGPT	11.0	14.1	16.7	15.3	18.2	21.3	19.4	23.2
AudioLM	9.5	9.2	12.0	12.1	15.0	22.1	17.6	25.1
SpiritLM	6.0	6.7	11.0	9.5	14.3	19.4	15.4	22.4
Moshi	5.5	7.0	12.0	7.2	8.8	10.6	9.4	14.2
Qwen2-Audio	1.8	/	3.6	/	7.1	/	8.6	/
Phi-4 Multimodal	2.1	4.8	3.5	4.1	6.3	11.5	9.2	10.8
SeamlessM4T-v2	3.3	6.3	4.2	5.3	7.5	10.3	10.3	12.1
MinMo	1.8	6.7	3.9	7.5	6.7	10.9	8.0	13.5
LLaMA-Omni2	3.5	10.1	4.0	9.2	9.5	12.4	11.3	17.2
MoST (Ours)	2.0	6.0	3.7	7.2	6.2	10.1	8.4	11.5

Table 2: Performance on Audio Language Modeling Tasks. MoST demonstrates superior performance across most benchmarks compared to existing models. **MoST (Ours)** results are highlighted with green background and improvement notes.

Model	sWUGGY	sBLIMP	sTopic-StoryCloze	sStoryCloze	Average
AudioLM	71.50	64.70	-	-	-
SpeechGPT	51.82	49.75	60.13	53.13	53.71
spiritLM	40.14	48.28	83.32	58.95	57.67
Moshi	51.14	53.31	46.34	45.16	48.99
Phi-4 Multimodal	71.84	60.21	81.55	62.39	69.00
MinMo	68.59	55.43	75.43	61.29	65.19
LLaMA-Omni2	73.21	53.59	78.21	68.55	68.39
MoST (Ours)	75.28 ↑2.1%	63.42	83.64 ↑0.3%	65.43	71.94 ↑2.9%

As shown in Table 2, MoST achieves competitive performance with an average score of 71.18, comparable to Phi-4 Multimodal (71.84) and outperforming MinMo (65.19) and LLaMA-Omni2 (68.39). MoST achieves state-of-the-art performance on sTopic-StoryCloze (83.64), demonstrating strong performance across multiple benchmarks. This strong performance underscores MAMoE’s efficacy in learning nuanced audio-specific linguistic patterns for enhanced spoken language understanding.

5.4 EVALUATION ON SPOKEN QUESTION ANSWERING TASKS

MoST’s Spoken Question Answering (SQA) capabilities are evaluated on Llama Q (Nachmani et al., 2024) (S→T and S→S), Trivial QA (Joshi et al., 2017) (S→T and S→S), and WebQ (Berant et al., 2013) (S→T and S→S). As shown in Figure 3, MoST demonstrates strong performance across all SQA tasks, achieving the best or competitive results in most settings. On Llama Q, MoST achieves 74.8 (S→T) and 62.6 (S→S), demonstrating robust speech-to-text understanding while maintaining competitive speech-to-speech performance. On Trivial QA, MoST scores 43.5 (S→T) and a notably high 32.1 (S→S), substantially exceeding most competitors. On WebQ, MoST achieves the best performance with 58.2 (S→T) and 44.7 (S→S), significantly outperforming all baselines including MinMo and LLaMA-Omni2. This strong performance, particularly in challenging S→S settings, highlights MoST’s robust end-to-end speech understanding and generation, attributed to the MAMoE architecture.

378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431

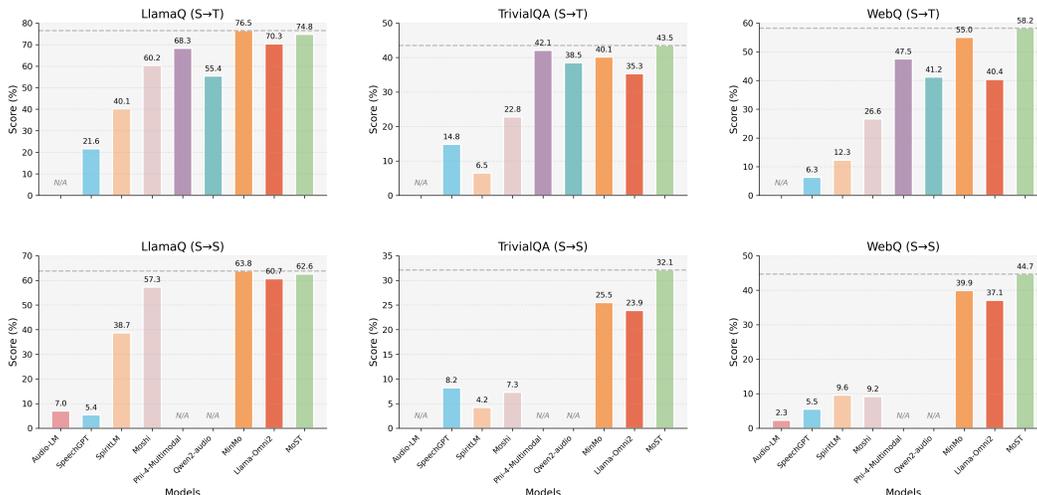


Figure 3: Performance comparison on Spoken Question Answering tasks. MoST achieves competitive or superior performances on all 3 datasets.

6 ABLATION STUDY: ARCHITECTURE BENEFITS WITH CONTROLLED COMPARISONS

6.1 CONTROLLED INITIALIZATION COMPARISON

To rigorously isolate the architectural contributions of MAMoE from LLM initialization quality, we conduct a controlled comparison using identical Llama3.2 3B (Grattafiori et al., 2024) initialization across three architectural variants: **Dense Baseline** (standard transformer without MoE), **Traditional MoE Upcycling** (Komatsuzaki et al., 2023) (modality-agnostic routing where all tokens compete for all experts), and **MoST-Style Upcycling** (the proposed MAMoE with modality-specific expert groups and shared experts). This controlled experimental design ensures that performance differences stem solely from architectural choices rather than varying initialization quality.

Figure 4 demonstrates the clear architectural progression benefits. Both MoE variants significantly outperform the dense baseline when starting from identical initialization, confirming MoE’s fundamental effectiveness for multimodal tasks. Most importantly, our MoST-style upcycling substantially surpasses traditional upcycling across all evaluation metrics, providing compelling evidence for MAMoE’s architectural advantages independent of initialization quality. The consistent performance gains—ranging from 7.3% improvement in ASR to 21.8% in SQA—validate that our proposed modality-aware routing mechanism delivers robust benefits regardless of the starting model. While stronger initialization (DeepSeek-V2 Lite) yields superior absolute performance for the full MoST model, this controlled comparison establishes that our architectural innovations provide consistent improvements across diverse speech-text tasks.

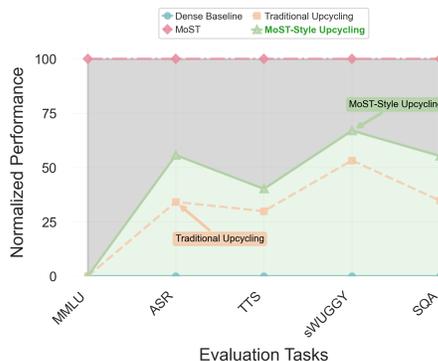


Figure 4: Controlled comparison with identical Llama3.2 3B initialization. We report (100%-WER) for ASR and TTS tasks and accuracy for other tasks.

6.2 MAMoE DESIGN VARIANTS ANALYSIS

To validate the individual contributions of MAMoE’s core components—modality-aware routing and shared experts—we conduct systematic ablations. We compare MoST against two variants: **MoST-**

VanillaMoE (standard MoE routing where all tokens compete for all experts) and **MoST-NoShared** (MAMoE routing but without shared experts, making all experts modality-specific). Both variants maintain the same total number of routed experts as full MoST for fair comparison, precisely isolating the contributions of modality-specific routing and shared expert capacity.

All three models were trained on the combined speech-text instruction following dataset for 10,000 steps (batch size 256). As illustrated in Figure 5, the full MoST model consistently outperforms both ablated variants across the comprehensive task suite. The comparison with MoST-VanillaMoE highlights the significant benefits of MAMoE’s modality-specific routing mechanism, while MoST’s advantage over MoST-NoShared underscores the crucial role of shared experts in enhancing model generalization and facilitating cross-modal understanding. The training and validation loss curves (Figure 5, top row) further corroborate these findings, showing more favorable learning dynamics for the full MAMoE architecture.

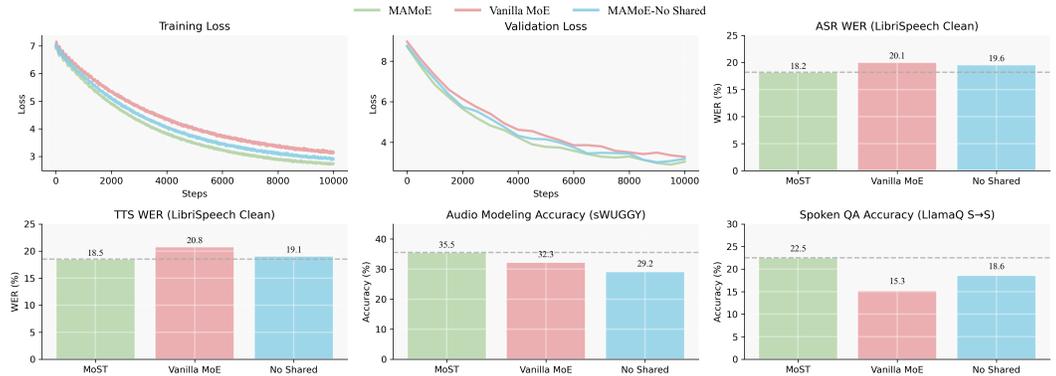


Figure 5: MAMoE design variants analysis. The 2x3 grid displays: Training loss curves; Validation loss curves; ASR, TTS performance (WER) on LibriSpeech dev-clean; Audio language modeling performance (Accuracy) on sWUGGY; Spoken QA performance (Accuracy) on LlamaQ. All metrics are reported at 10,000 training steps. Full MAMoE consistently outperforms ablated versions.

6.3 EXPERT ROUTING ANALYSIS

To provide deeper insights into MAMoE’s routing mechanism, we analyze expert utilization patterns and quantitative routing metrics. Figure 6 presents a comprehensive visualization combining routing frequency heatmaps (left) with quantitative routing metrics (right) for MoST with MAMoE versus vanilla MoE.

The routing heatmaps reveal that MAMoE exhibits distinct modality-specific routing patterns that become increasingly pronounced during training. Text and audio tokens are systematically directed to their respective expert groups, fostering specialized processing pathways. The quantitative analysis (right panel) confirms that MAMoE achieves progressively lower routing entropy during training, indicating enhanced expert specialization. Besides, MAMoE maintains superior load balancing with consistently lower Gini coefficients compared to vanilla MoE, demonstrating the effectiveness of our routing design.

7 CONCLUSION

We introduced MoST, a novel multimodal LLM integrating speech and text with a Modality-Aware Mixture of Experts architecture. MAMoE’s modality-specific routing and shared experts enhance specialized processing and cross-modal interaction. Our efficient pipeline transforms a pretrained MoE LLM into a data-efficient speech-text model. Empirical evaluations confirm MoST achieves state-of-the-art or competitive results across ASR, TTS, audio language modeling, and SQA, outperforming strong baselines.

486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539

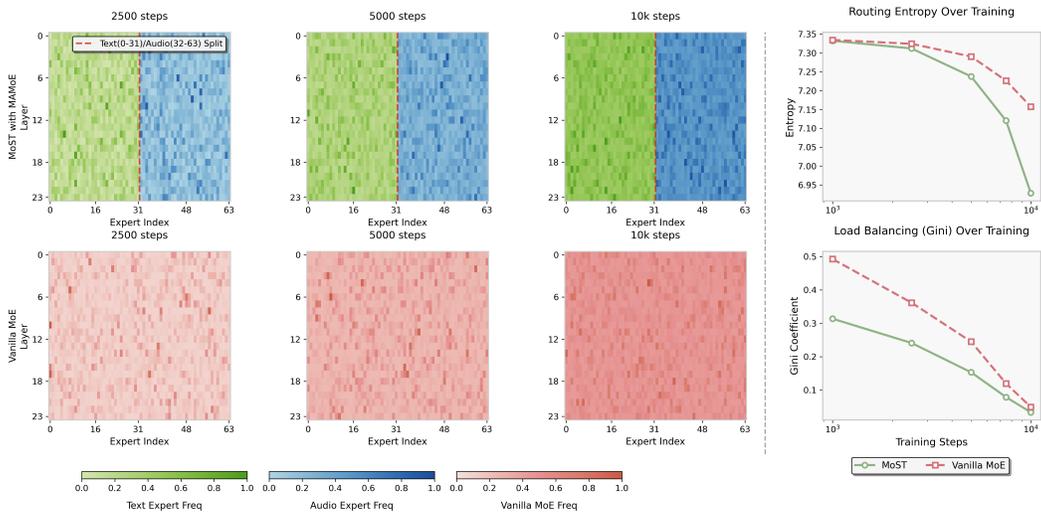


Figure 6: Expert routing analysis for MAMoE versus vanilla MoE. **Left:** Routing frequency heatmaps showing MAMoE’s modality-specific routing (top) versus vanilla MoE’s diffuse patterns (bottom) at 2,500, 5,000, and 10,000 training steps. **Right:** Quantitative metrics showing MAMoE’s superior specialization (lower entropy) and load balancing (lower Gini coefficient) throughout training.

While our index-based 50% expert partition proves effective, it represents a simple initialization strategy. More sophisticated approaches—such as clustering-based or knowledge-preserving partitioning—offer promising directions for future modality-aware MoE research (detailed in Appendix D).

We release MoST’s model checkpoints, codebase, and curated data to encourage further research and development in efficient and effective multimodal models.

REPRODUCIBILITY STATEMENT

To ensure transparency and facilitate reproducibility, we release the full resources required to replicate our experiments. Specifically, we provide the MoST model implementation, including both training and inference code, as well as the complete training datasets, at <https://github.com/anonymous29008/MoST/tree/main>.

All model configurations used for MoST are reported in Appendix C, detailing architecture choices, hyperparameters, and training schedules. The released code includes scripts for data preprocessing, model training, and evaluation, enabling end-to-end reproduction of our results without additional modifications.

We have verified that running the released code with the provided configurations reproduces the main results presented in the paper. Furthermore, instructions are included in the repository to guide users through environment setup, dependency installation, and hardware requirements.

REFERENCES

- Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. Smollm2: When smol goes big – data-centric training of a small language model, 2025. URL <https://arxiv.org/abs/2502.02737>.
- Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus, 2020a. URL <https://arxiv.org/abs/1912.06670>.
- Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus, 2020b. URL <https://arxiv.org/abs/1912.06670>.
- Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1533–1544, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/D13-1160>.
- Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audioldm: a language modeling approach to audio generation, 2023. URL <https://arxiv.org/abs/2209.03143>.
- Qian Chen, Yafeng Chen, Yanni Chen, Mengzhe Chen, Yingda Chen, Chong Deng, Zhihao Du, Ruize Gao, Changfeng Gao, Zhifu Gao, Yabin Li, Xiang Lv, Jiaqing Liu, Haoneng Luo, Bin Ma, Chongjia Ni, Xian Shi, Jialong Tang, Hui Wang, Hao Wang, Wen Wang, Yuxuan Wang, Yunlan Xu, Fan Yu, Zhijie Yan, Yexin Yang, Baosong Yang, Xian Yang, Guanrou Yang, Tianyu Zhao, Qinglin Zhang, Shiliang Zhang, Nan Zhao, Pei Zhang, Chong Zhang, and Jinren Zhou. Minmo: A multimodal large language model for seamless voice interaction, 2025. URL <https://arxiv.org/abs/2501.06282>.
- Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-audio technical report, 2024. URL <https://arxiv.org/abs/2407.10759>.
- Seamless Communication, Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, John Hoffman, Min-Jae Hwang, Hirofumi Inaguma, Christopher Klaiber, Iliia Kulikov, Pengwei Li, Daniel Licht, Jean Maillard, Ruslan Mavlyutov, Alice Rakotoarison, Kaushik Ram Sadagopan, Abinesh Ramakrishnan, Tuan Tran, Guillaume Wenzek, Yilin Yang, Ethan Ye, Ivan Evtimov, Pierre Fernandez, Cynthia Gao, Prangthip Hansanti, Elahe Kalbassi, Amanda Kallet, Artyom Kozhevnikov, Gabriel Mejia Gonzalez, Robin San Roman, Christophe Touret, Corinne Wong,

- 594 Carleigh Wood, Bokai Yu, Pierre Andrews, Can Balioglu, Peng-Jen Chen, Marta R. Costa-jussà,
595 Maha Elbayad, Hongyu Gong, Francisco Guzmán, Kevin Heffernan, Somya Jain, Justine Kao,
596 Ann Lee, Xutai Ma, Alex Mourachko, Benjamin Peloquin, Juan Pino, Sravya Popuri, Christophe
597 Ropers, Safiyyah Saleem, Holger Schwenk, Anna Sun, Paden Tomasello, Changhan Wang, Jeff
598 Wang, Skyler Wang, and Mary Williamson. Seamless: Multilingual expressive and streaming
599 speech translation, 2023. URL <https://arxiv.org/abs/2312.05187>.
- 600 DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi
601 Dengr, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li,
602 Fangyun Lin, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Hao
603 Yang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian
604 Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jin Chen, Jingyang Yuan, Junjie Qiu, Junxiao Song, Kai
605 Dong, Kaige Gao, Kang Guan, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Liyue
606 Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming
607 Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J.
608 Chen, R. L. Jin, Ruiqi Ge, Ruizhe Pan, Runxin Xu, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan
609 Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou,
610 Shuiping Yu, Shunfeng Zhou, Size Zheng, T. Wang, Tian Pei, Tian Yuan, Tianyu Sun, W. L.
611 Xiao, Wangding Zeng, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wentao Zhang, X. Q. Li,
612 Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang
613 Chen, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Liu, Xin Xie, Xingkai
614 Yu, Xinnan Song, Xinyi Zhou, Xinyu Yang, Xuan Lu, Xuecheng Su, Y. Wu, Y. K. Li, Y. X. Wei,
615 Y. X. Zhu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui
616 Wang, Yi Zheng, Yichao Zhang, Yiliang Xiong, Yilong Zhao, Ying He, Ying Tang, Yishi Piao,
617 Yixin Dong, Yixuan Tan, Yiyuan Liu, Yongji Wang, Yongqiang Guo, Yuchen Zhu, Yudian Wang,
618 Yuheng Zou, Yukun Zha, Yunxian Ma, Yuting Yan, Yuxiang You, Yuxuan Liu, Z. Z. Ren, Zehui
619 Ren, Zhangli Sha, Zhe Fu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhewen Hao, Zhihong Shao,
620 Zhiniu Wen, Zhipeng Xu, Zhongyu Zhang, Zhuoshu Li, Zihan Wang, Zihui Gu, Zilin Li, and Ziwei
621 Xie. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model, 2024.
URL <https://arxiv.org/abs/2405.04434>.
- 622 Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim
623 Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten Bosma,
624 Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson,
625 Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc V Le, Yonghui Wu, Zhifeng
626 Chen, and Claire Cui. Glam: Efficient scaling of language models with mixture-of-experts, 2022.
627 URL <https://arxiv.org/abs/2112.06905>.
- 628 Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou,
629 Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue,
630 2024. URL <https://arxiv.org/abs/2410.00037>.
- 631 Qingkai Fang, Yan Zhou, Shoutao Guo, Shaolei Zhang, and Yang Feng. Llama-omni2: Llm-
632 based real-time spoken chatbot with autoregressive streaming speech synthesis, 2025. URL
633 <https://arxiv.org/abs/2505.02625>.
- 634 Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Junteng Jia, Yuan Shangguan, Ke Li, Jinxi Guo,
635 Wenhan Xiong, Jay Mahadeokar, Ozlem Kalinli, Christian Fuegen, and Mike Seltzer. Prompting
636 large language models with speech recognition abilities, 2023. URL <https://arxiv.org/abs/2307.11795>.
- 637 William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter
638 models with simple and efficient sparsity, 2022. URL <https://arxiv.org/abs/2101.03961>.
- 639 Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad
640 Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan,
641 Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev,
642 Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru,
643 Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak,
644 Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu,

648 Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle
649 Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego
650 Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova,
651 Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel
652 Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon,
653 Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan
654 Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet,
655 Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde,
656 Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie
657 Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua
658 Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak,
659 Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley
660 Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yearly, Laurens van der Maaten, Lawrence
661 Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas
662 Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri,
663 Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie
664 Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes
665 Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne,
666 Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajwal
667 Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong,
668 Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic,
669 Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie
670 Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana
671 Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie,
672 Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon
673 Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan,
674 Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas
675 Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami,
676 Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti,
677 Vitor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier
678 Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao
679 Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song,
680 Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe
681 Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya
682 Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei
683 Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu,
684 Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit
685 Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury,
686 Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer,
687 Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu,
688 Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido,
689 Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu
690 Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer,
691 Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu,
692 Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, DingKang Wang, Duc
693 Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily
694 Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers,
695 Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank
696 Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee,
697 Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan,
698 Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph,
699 Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog,
700 Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James
701 Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny
Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings,
Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai
Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik
Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle
Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng

- 702 Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish
703 Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim
704 Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle
705 Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang,
706 Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam,
707 Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier,
708 Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia
709 Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro
710 Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani,
711 Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy,
712 Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin
713 Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu,
714 Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh
715 Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay,
716 Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang,
717 Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie
718 Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta,
719 Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman,
720 Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun
721 Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria
722 Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru,
723 Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz,
724 Will Constable, Xiaocheng Tang, Xiaojuan Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv
725 Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi,
726 Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait,
727 Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The
728 llama 3 herd of models, 2024. URL <https://arxiv.org/abs/2407.21783>.
- 729 Michael Hassid, Tal Remez, Tu Anh Nguyen, Itai Gat, Alexis Conneau, Felix Kreuk, Jade Copet,
730 Alexandre Defossez, Gabriel Synnaeve, Emmanuel Dupoux, et al. Textually pretrained speech
731 language models. *Advances in Neural Information Processing Systems*, 36:63483–63501, 2023.
- 732 Michael Hassid, Tal Remez, Tu Anh Nguyen, Itai Gat, Alexis Conneau, Felix Kreuk, Jade Copet,
733 Alexandre Defossez, Gabriel Synnaeve, Emmanuel Dupoux, Roy Schwartz, and Yossi Adi. Text-
734 tually pretrained speech language models, 2024. URL <https://arxiv.org/abs/2305.13009>.
- 735 Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov,
736 and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked
737 prediction of hidden units, 2021. URL <https://arxiv.org/abs/2106.07447>.
- 738 Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale
739 distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and
740 Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Com-
741 putational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July
742 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL <https://aclanthology.org/P17-1147/>.
- 743 Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and
744 Daniel Povey. Libriheavy: a 50,000 hours asr corpus with punctuation casing and context, 2024.
745 URL <https://arxiv.org/abs/2309.08105>.
- 746 Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua
747 Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. Sparse upcycling: Training mixture-of-
748 experts from dense checkpoints, 2023. URL <https://arxiv.org/abs/2212.05055>.
- 749 Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for
750 efficient and high fidelity speech synthesis, 2020. URL <https://arxiv.org/abs/2010.05646>.
- 751 Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte,
752 Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, et al. On generative spoken
753
754
755

- 756 language modeling from raw audio. *Transactions of the Association for Computational Linguistics*,
757 9:1336–1354, 2021.
- 758
- 759 Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang,
760 Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional
761 computation and automatic sharding, 2020. URL <https://arxiv.org/abs/2006.16668>.
- 762 Xi Victoria Lin, Akshat Shrivastava, Liang Luo, Srinivasan Iyer, Mike Lewis, Gargi Ghosh, Luke
763 Zettlemoyer, and Armen Aghajanyan. Moma: Efficient early-fusion pre-training with mixture of
764 modality-aware experts, 2024. URL <https://arxiv.org/abs/2407.21770>.
- 765 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. URL
766 <https://arxiv.org/abs/2304.08485>.
- 767
- 768 Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL <https://arxiv.org/abs/1711.05101>.
- 769
- 770 Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen
771 Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen,
772 Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, Xiyang
773 Dai, Ruchao Fan, Mei Gao, Min Gao, Amit Garg, Abhishek Goswami, Junheng Hao, Amr Hendy,
774 Yuxuan Hu, Xin Jin, Mahmoud Khademi, Dongwoo Kim, Young Jin Kim, Gina Lee, Jinyu Li,
775 Yunsheng Li, Chen Liang, Xihui Lin, Zeqi Lin, Mengchen Liu, Yang Liu, Gilsinia Lopez, Chong
776 Luo, Piyush Madan, Vadim Mazalov, Arindam Mitra, Ali Mousavi, Anh Nguyen, Jing Pan, Daniel
777 Perez-Becker, Jacob Platin, Thomas Portet, Kai Qiu, Bo Ren, Liliang Ren, Sambuddha Roy,
778 Ning Shang, Yelong Shen, Saksham Singhal, Subhojit Som, Xia Song, Tetyana Sych, Praneetha
779 Vaddamanu, Shuohang Wang, Yiming Wang, Zhenghao Wang, Haibin Wu, Haoran Xu, Weijian
780 Xu, Yifan Yang, Ziyi Yang, Donghan Yu, Ishmam Zabir, Jianwen Zhang, Li Lyna Zhang, Yunan
781 Zhang, and Xiren Zhou. Phi-4-mini technical report: Compact yet powerful multimodal language
782 models via mixture-of-loras, 2025. URL <https://arxiv.org/abs/2503.01743>.
- 783 Eliya Nachmani, Alon Levkovitch, Roy Hirsch, Julian Salazar, Chulayuth Asawaroengchai, Soroosh
784 Mariooryad, Ehud Rivlin, RJ Skerry-Ryan, and Michelle Tadmor Ramanovich. Spoken question
785 answering and speech continuation using spectrogram-powered llm, 2024. URL <https://arxiv.org/abs/2305.15255>.
- 786
- 787 Tu Anh Nguyen, Maureen de Seyssel, Patricia Rozé, Morgane Rivière, Evgeny Kharitonov, Alexei
788 Baevski, Ewan Dunbar, and Emmanuel Dupoux. The zero resource speech benchmark 2021:
789 Metrics and baselines for unsupervised spoken language modeling, 2020. URL <https://arxiv.org/abs/2011.11588>.
- 790
- 791 Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussa, Maha Elbayad, Sravya Popuri,
792 Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary
793 Williamson, Gabriel Synnaeve, Juan Pino, Benoit Sagot, and Emmanuel Dupoux. Spirit lm:
794 Interleaved spoken and written language model, 2024. URL <https://arxiv.org/abs/2402.05755>.
- 795
- 796
- 797 OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni
798 Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor
799 Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian,
800 Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny
801 Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks,
802 Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea
803 Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen,
804 Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung,
805 Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch,
806 Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty
807 Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Justin Forte,
808 Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel
809 Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua
Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike
Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon

- 810 Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne
811 Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo
812 Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar,
813 Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik
814 Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich,
815 Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy
816 Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie
817 Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini,
818 Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne,
819 Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David
820 Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie
821 Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély,
822 Ashvin Nair, Reiichiro Nakano, Rajeef Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo
823 Noh, Long Ouyang, Cullen O’Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano,
824 Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng,
825 Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto,
826 Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power,
827 Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis
828 Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted
829 Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel
830 Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon
831 Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky,
832 Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie
833 Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng,
834 Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun
835 Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang,
836 Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lillian
837 Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren
838 Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming
839 Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao
840 Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL
841 <https://arxiv.org/abs/2303.08774>.
- 842 Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and
843 Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, 2017.
844 URL <https://arxiv.org/abs/1701.06538>.
- 845 Leyang Shen, Gongwei Chen, Rui Shao, Weili Guan, and Liqiang Nie. Mome: Mixture of multimodal
846 experts for generalist multimodal large language models, 2024. URL <https://arxiv.org/abs/2407.12709>.
- 847 Changhan Wang, Morgane Rivière, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary
848 Williamson, Juan Pino, and Emmanuel Dupoux. Voxpopuli: A large-scale multilingual speech
849 corpus for representation learning, semi-supervised learning and interpretation, 2021. URL
850 <https://arxiv.org/abs/2101.00390>.
- 851 Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu.
852 Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities,
853 2023. URL <https://arxiv.org/abs/2305.11000>.
- 854 Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms:
855 Recent advances in multimodal large language models, 2024. URL <https://arxiv.org/abs/2401.13601>.
- 856
857
858
859
860
861
862
863

A DETAILED TRAINING RECIPE

To effectively realize MoST, we develop an efficient LLM-to-Speech-Text Transformation Pipeline. Starting from a pretrained MoE LLM, we first conduct targeted post-training on large ASR and TTS datasets. This stage primes the modality-specific experts and adapts the shared experts for cross-modal tasks. Subsequently, we perform instruction fine-tuning using a carefully curated speech-text instruction dataset. Our training strategy involves mixed-task learning and incorporates techniques to prevent catastrophic forgetting of foundational abilities during fine-tuning. Critically, this pipeline demonstrates remarkable data efficiency, achieving strong performance with significantly fewer training tokens compared to prominent open-source speech-text models like SpiritLM (Nguyen et al., 2024) and Moshi (Défossez et al., 2024).

Table 3 provides a comprehensive overview of our two-stage training protocol, including key hyper-parameters, task mixing ratios, and dataset composition details for each stage.

Table 3: Training Configuration Overview. Detailed breakdown of training stages, configurations, task mixing ratios, and dataset compositions.

Training Stage	Configuration	Task Mixing Ratios	Dataset Composition Details
Stage 1: Cross-Modal Post-Training	Steps: 500k Batch Size: 512	ASR: 0.4 TTS: 0.4 Text LM: 0.2	ASR/TTS: LibriHeavy (0.6), Common Voice (0.2), VoxPopuli (0.2) LM: RefinedWeb (to prevent forgetting)
Stage 2: Mixed Instruction Tuning	Steps: 10k Batch Size: 128	Speech Instruct: 0.4 Text Instruct: 0.4 ASR: 0.1 TTS: 0.1	Speech Instruct: Synthesized from SmolTalk (via MoST Stage 1) Text Instruct: Standard open-source instruction data

A.1 ASR AND TTS DATA PREPARATION (STAGE 1)

The initial stage of our LLM-to-Speech-Text transformation pipeline involves post-training on large-scale Automatic Speech Recognition (ASR) and Text-to-Speech (TTS) datasets. This step is crucial for priming the modality-specific experts within the MAMoE architecture and adapting the shared experts for fundamental cross-modal alignment tasks.

We utilize two primary open data sources for this stage: the Common Voice corpus (Ardila et al., 2020b) and the full LibriHeavy dataset (Kang et al., 2024), selected for their scale and diversity in speech patterns and linguistic content. The ASR data follows the standard format where the input consists of speech (represented as acoustic features or tokens) and the output is the corresponding text transcription, guided by an instruction like "Transcribe the following speech into text."

To generate the necessary TTS data, we systematically transform the ASR datasets. For each ASR sample `{instruction: "Transcribe...", input: speech, output: text}`, we create a corresponding TTS sample by reversing the input and output modalities and modifying the instruction. The resulting TTS sample format is `{instruction: "Convert the following text into speech.", input: text, output: speech}`. This conversion process, (further details were originally noted to be in supplementary materials, which is consistent with an appendix location), ensures that the model is trained on paired speech-text data for both recognition and synthesis tasks, effectively leveraging existing ASR resources to create aligned TTS training data. This phase prepares the model for more complex instruction-following tasks in the subsequent fine-tuning stage.

A.2 SPEECH-TEXT INSTRUCTION-FOLLOWING DATA CONSTRUCTION (STAGE 2)

Following the initial post-training phase, MoST undergoes instruction fine-tuning to enhance its ability to follow complex commands and engage in nuanced interactions involving both speech and text. This requires a diverse dataset of speech-text instructions. We construct this dataset through two

918 complementary strategies: synthesizing interrupted dialogues and converting text-based instructions
919 into a speech format.
920

921 A.2.1 INTERRUPTED DIALOGUE SYNTHESIS PIPELINE 922

923 Standard instruction datasets often lack the dynamic turn-taking and interruptions characteristic of
924 natural human conversation. To address this, we develop a pipeline to augment existing dialogue
925 datasets with realistic interruptions. We utilize the open-source SmolTalk dataset (Allal et al., 2025)
926 as our base, which contains multi-turn user-assistant conversations.
927

928 Our synthesis pipeline employs a large language model (LLM) in a two-step process. First, the
929 LLM analyzes each dialogue from the text instruction following dataset to determine its suitability
930 for adding a user interruption. Suitability criteria include clear role separation (user/assistant),
931 the presence of longer assistant turns amenable to interruption, and contextual appropriateness for
932 an interruption. Second, for dialogues identified as suitable, the LLM is prompted to rewrite the
933 conversation by inserting a natural user interruption during one of the assistant’s longer responses.
934 This interruption typically takes the form of a brief clarifying question related to the assistant’s
935 immediately preceding statement. The LLM is instructed to modify the assistant’s response to briefly
936 address the interruption before seamlessly transitioning back to its original point (e.g., "A host cell
937 is... Now, as I was saying..."). The remainder of the dialogue remains unchanged. This process yields
938 a dataset enriched with more complex conversational flows, better preparing MoST for interactive
939 scenarios.

940 A.2.2 TEXT-TO-SPEECH INSTRUCTION DATASET CONVERSION 941

942 To enable our model with both speech and text instruction following abilities, we employ a strategy
943 where we leverage the capabilities of our own MoST model, specifically after its initial post-training
944 on ASR and TTS data (as described in Appendix A.1).

945 For a given text-based instruction sample (e.g., {instruction: text_prompt, input:
946 text_input, output: text_response}), we use the ASR-TTS trained MoST model
947 to synthesize the speech equivalent of the text_response. This converts the sample into
948 a speech-output format (e.g., {instruction: text_prompt, input: text_input,
949 output: speech_response}). Depending on the target task, we can also synthesize speech
950 for the text_input or even the instruction itself, creating varied multimodal instruction
951 formats. This approach allows us to efficiently repurpose vast quantities of high-quality text instruc-
952 tions for speech-based fine-tuning without the need for manual speech data collection, significantly
953 broadening the range of tasks MoST is trained on.
954

955 A.3 MIXED SPEECH-TEXT INSTRUCTION FINE-TUNING AND HYPER-PARAMETERS (STAGE 3) 956

957 Our training process for MoST (after Stage 1 post-training described in Appendix A.1 and Stage 2
958 data construction in Appendix A.2) proceeds to instruction fine-tuning.

959 **Instruction Fine-tuning Strategy.** This stage uses the curated dataset comprising synthesized
960 interrupted dialogues and text-to-speech converted instructions. The goal is to teach the model
961 complex reasoning, interaction, and command-following involving both modalities. Crucially, to
962 mitigate catastrophic forgetting of the foundational ASR and TTS capabilities learned in Stage 1,
963 the instruction fine-tuning data is mixed with a significant portion of the original ASR and TTS
964 data. This ensures that MoST retains its core speech processing abilities while acquiring higher-level
965 instruction-following competence. The mixing ratio between new instruction data and the ASR/TTS
966 data is carefully managed throughout this stage.
967

968 B TRAINING SETTINGS 969

970 This section details the hyper-parameters used for training MoST and the computing resources utilized
971 for the experiments.

972 B.1 TRAINING HYPER-PARAMETERS

973
974 Across all training stages (post-training and fine-tuning), we utilize the AdamW optimizer (Loshchilov
975 & Hutter, 2019) with a cosine learning rate schedule, including a warm-up phase. Specific values for
976 the peak learning rate, weight decay, batch size, and the exact number of training steps or epochs for
977 each stage, along with the mixing ratios used during instruction fine-tuning are as follows:

978 Peak Learning Rate: 5e-5, Weight Decay: 0.01, Batch Size for ASR-TTS Post-training: 256, Batch
979 Size for Mixed Instruction Following Fine-tuning: 128, Fine-tuning Steps: 10000, Warmup Steps:
980 1000, Mixing Ratio (Instruction:ASR/TTS): 1:1. These parameters were chosen based on preliminary
981 experiments to ensure stable and efficient convergence.

982
983 B.2 EXPERIMENT COMPUTING RESOURCES

984
985 The training experiments for MoST were conducted using a distributed setup comprising 48 NVIDIA
986 A100 GPUs. Specifically, this involved 6 nodes, each equipped with 8 A100 GPUs.

988 C MODEL CONFIGURATIONS

989
990 Table 4 details the model configurations used for MoST.

991
992 Table 4: Model configurations from `config_mostc.json`.

993
994
995
996

	Value
	["MoSTCForCausalLM"]
<code>_bias</code>	false
<code>_dropout</code>	0.0
<code>_expert_indices</code>	[32, 33, ..., 62, 63]
<i>auto_map:</i>	
AutoConfig	configuration_MoST.MoSTCConfig
AutoModel	modeling_most.MoSTCModel
AutoModelForCausalLM	modeling_most.MoSTCForCausalLM
<code>_loss_alpha</code>	0.001
<code>_token_id</code>	100000
<code>_token_id</code>	100001
<code>_k_dense_replace</code>	1
<code>_act</code>	silu
<code>_size</code>	2048
<code>_range</code>	0.02

1025

1026 Table 4: Model configurations from `config_mostc.json` (Continued).
 1027
 1028

1029		Value
1030		
1031		
1032		
1033	<code>_size</code>	10944
1034		
1035	<code>_lora_rank</code>	512
1036		
1037	<code>_position_embeddings</code>	163840
1038		
1039	<code>_type</code>	MoSTC
1040		
1041	<code>_intermediate_size</code>	1408
1042		
1043	<code>_layer_freq</code>	1
1044		
1045	<code>_group</code>	1
1046		
1047	<code>_routed_experts</code>	64
1048		
1049	<code>_shared_experts</code>	2
1050		
1051	<code>_topk_prob</code>	false
1052		
1053	<code>_attention_heads</code>	16
1054		
1055	<code>_experts_per_tok</code>	6
1056		
1057	<code>_hidden_layers</code>	27
1058		
1059	<code>_key_value_heads</code>	16
1060		
1061	<code>_tp</code>	1
1062		
1063	<code>_lora_rank</code>	null
1064		
1065	<code>_nope_head_dim</code>	128
1066		
1067	<code>_rope_head_dim</code>	64
1068		
1069	<code>_norm_eps</code>	1e-06
1070		
1071	<i>rope_scaling:</i>	
1072	<code>beta_fast</code>	32
1073		
1074	<code>beta_slow</code>	1
1075		
1076	<code>factor</code>	40
1077		
1078	<code>msscale</code>	0.707
1079		
	<code>msscale_all_dim</code>	0.707
	<code>original_max_position_embeddings</code>	4096

1080
 1081
 1082
 1083
 1084
 1085
 1086
 1087
 1088
 1089
 1090
 1091
 1092
 1093
 1094
 1095
 1096
 1097
 1098
 1099
 1100
 1101
 1102
 1103
 1104
 1105
 1106
 1107
 1108
 1109
 1110
 1111
 1112
 1113
 1114
 1115
 1116
 1117
 1118
 1119
 1120
 1121
 1122
 1123
 1124
 1125
 1126
 1127
 1128
 1129
 1130
 1131
 1132
 1133

Table 4: Model configurations from `config_mostc.json` (Continued).

	Value
<code>type</code>	yarn
<code>_theta</code>	10000
<code>_scaling_factor</code>	1.0
<code>_func</code>	softmax
<code>_aux</code>	true
<code>_expert_indices</code>	[0, 1, ..., 30, 31]
<code>_word_embeddings</code>	false
<code>_group</code>	1
<code>_method</code>	greedy
<code>_dtype</code>	bfloat16
<code>_version</code>	4.33.1
<code>_cache</code>	true
<code>_modality_aware_routing</code>	true
<code>_head_dim</code>	128
<code>_size</code>	102400
<code>_continuous_audio</code>	true
<code>_model_path</code>	/.../hubert_MoSTC/mhubert_base_25hz.pt
<code>_model_path</code>	/.../hubert_MoSTC/L11_quantizer_500.pt
<code>_ckpt_type</code>	converted
<code>_hidden_size</code>	768
<code>_audio_token_id</code>	100502
<code>_audio_token_id</code>	100503
<code>_audio_wave_token_id</code>	100504
<code>_audio_wave_token_id</code>	100505

D LIMITATIONS, SOCIAL IMPACTS AND FUTURE WORK

MoST presents both potential societal benefits (e.g., accessibility) and risks (e.g., voice spoofing).

Rationale for Index-Based Partition. In standard sparse MoE models like DeepSeek-V2, experts are randomly initialized and do not exhibit semantic clustering (e.g., specialized "text experts" vs. "math experts") prior to training. Consequently, a random index-based split at initialization is statistically neutral—no particular subset of experts has inherent advantages for any specific modality. Our Stage 1 Cross-Modal Post-Training is explicitly designed to specialize these initially random experts into their assigned modalities through targeted ASR/TTS training.

Role of Shared Experts. Critically, the Shared Experts in MAMoE process every token regardless of modality, providing a mechanism to preserve general capabilities and facilitate cross-modal knowledge transfer. This architectural choice helps mitigate the risk of losing valuable knowledge during the hard partition, as evidenced by MoST’s strong performance on text-based benchmarks (Appendix E).

Future Directions. While our approach demonstrates strong empirical results, we recognize that the index-based partition may not represent the theoretical optimum. More sophisticated initialization strategies represent important directions for future research, including: (1) *Clustering-based initialization*: Analyzing pretrained expert activations on representative data to group experts by their learned specializations before partitioning; (2) *Knowledge-preserving partitioning*: Developing methods to identify and preserve critical expert subsets for each modality; (3) *Adaptive expert allocation*: Dynamically adjusting the ratio of modality-specific experts during training based on task requirements.

Future work presents several exciting avenues. A primary direction is extending the MAMoE concept to incorporate additional modalities, particularly vision, to create a truly comprehensive audio-visual-language model. Further research could also explore scaling MoST to larger parameter counts, investigating different types of expert specializations, and evaluating its capabilities on a broader range of downstream speech and multimodal tasks.

E TEXT-BASED EVALUATIONS

To comprehensively assess MoST’s capabilities beyond speech-specific tasks, we evaluate its performance on standard text-based benchmarks. Table 5 presents results across four diverse text benchmarks: MMLU (general knowledge), TriviaQA (factual question answering), GSM8K (mathematical reasoning), and HumanEval (code generation).

Table 5: Performance on Text-Based Evaluation Benchmarks. MoST demonstrates strong text understanding and generation capabilities, outperforming most speech-native models while maintaining competitive performance with text-focused systems. Higher scores are better. **MoST (Ours)** results are highlighted with green background.

Model	MMLU	TriviaQA	GSM8K	HumanEval
MinMo	58.5	65.8	58.1	42.3
Llama-Omni2	44.7	35.2	21.8	13.2
Moshi	49.8	48.5	40.3	25.4
Qwen2-Audio	45.2	34.1	24.3	11.7
SpiritLM	36.9	42.0	21.5	9.7
MoST (Ours)	55.4	67.1	66.9	58.2

MoST significantly outperforms other open-source speech-native models (SpiritLM, Moshi, Qwen2-Audio) and recent end-to-end models like Llama-Omni2 across text tasks. MoST also maintains strong general reasoning capabilities (GSM8K 66.9) and coding abilities (HumanEval 58.2), confirming that our shared-expert design successfully mitigates catastrophic forgetting. While MinMo achieves a higher MMLU score (58.5 vs 55.4), MoST demonstrates superior performance on TriviaQA, GSM8K, and HumanEval, highlighting the effectiveness of our MAMoE architecture in preserving and enhancing text-based capabilities during multimodal training.

1188 F USE OF LLMs

1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241

As this work is centered on large language models (LLMs), they are inherently part of our study. We clarify the roles of LLMs in four distinct but limited capacities:

- **Base model.** The core of our proposed MoST model is built upon an existing LLM as its backbone, consistent with the research objective of studying and advancing LLM-based methods. And we include it as part of reproducibility as well.
- **LM-guided synthesis.** In data synthesis section, LLMs were used as judges and converters to evaluate candidate outputs and perform format conversion. These roles were auxiliary, and MoST’s effectiveness does not depend on LLM-specific heuristics.
- **Baselines.** We adopted open-source LLMs as baselines to ensure fair and transparent comparisons, with results reproduced from official or widely accepted implementations.
- **Language correction.** LLMs were employed only for grammar correction and minor error detection in the manuscript, without altering the scientific content or analysis.

Overall, LLMs were employed only in auxiliary or benchmarking roles. Importantly, LLMs were never used for idea generation or the formulation of scientific arguments. All core algorithmic design, experimental findings, and analyses remain fully attributable to our proposed methodology.