SLAM-Omni: Timbre-Controllable Voice Interaction System with Single-Stage Training

Wenxi Chen[1*], Ziyang Ma[1], Ruiqi Yan[1], Yuzhe Liang[1], Xiquan Li[1]
Ruiyang Xu[1], Zhikang Niu[1,2], Yanqiao Zhu[1], Yifan Yang[1], Zhanxun Liu[1]
Kai Yu[1], Yuxuan Hu[3], Jinyu Li[3], Yan Lu[3], Shujie Liu[3†], Xie Chen[1,2†]
[1]X-LANCE Lab, School of Computer Science
MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University
[2]Shanghai Innovation Institute, [3]Microsoft Corporation
{1029713857,chenxie95}@sjtu.edu.cn

Abstract

Recent advancements highlight the potential of end-to-end real-time spoken dialogue systems, showcasing their low latency and high quality. In this paper, we introduce SLAM-Omni, a timbre-controllable, end-to-end voice interaction system with single-stage training. SLAM-Omni achieves zero-shot timbre control by modeling spoken language with semantic tokens and decoupling speaker information to a vocoder. By predicting grouped speech semantic tokens at each step, our method significantly reduces the sequence length of audio tokens, accelerating both training and inference. Additionally, we propose historical text prompting to compress dialogue history, facilitating efficient multi-round interactions. Comprehensive evaluations reveal that SLAM-Omni outperforms prior models of similar scale, requiring only 15 hours of training on 4 GPUs with limited data. Notably, it is the first spoken dialogue system to achieve competitive performance with a single-stage training approach, eliminating the need for pre-training on TTS or ASR tasks. Further experiments validate its multilingual and multi-turn dialogue capabilities on larger datasets.[1]

1 Introduction

With the advent of large language models (LLMs), recent developments (Achiam et al., 2023; Dubey et al., 2024; Yang et al., 2024a) have showcased their powerful capabilities in textual conversation. In spoken dialogue systems, however, traditional methods rely on a cascaded pipeline involving automatic speech recognition (ASR) to transcribe user input, LLMs to generate textual responses, and text-to-speech (TTS) models to produce audio outputs.

This design faces two major issues: (1) significantly increased interaction latency, and (2) reliance on text-based interaction, which overlooks rich nonverbal information in speech dialogue, such as emotions and prosody. The release of GPT-4o (OpenAI, 2024b) has underscored the potential of real-time spoken dialogue systems in delivering seamless interaction. In response, several open-source frameworks, including Moshi (Défossez et al., 2024), Mini-Omni (Xie and Wu, 2024a,b), and LLaMA-Omni (Fang et al., 2024), have been developed for effective end-to-end voice-based interaction.

Existing spoken dialogue models (SDMs) primarily model speech with discretized audio tokens. Some approaches (Fang et al., 2024; Wang et al., 2024) rely on text embeddings to guide audio token generation, which limits their ability to generate critical audio paralinguistic attributes such as emotion and prosody. Others (Zeng et al., 2024b; Zhang et al., 2024; Nguyen et al., 2024) adopt interleaved arrangements of audio and text tokens to restructure language modeling, while increasing training costs. A third category (Xie and Wu, 2024a,b; Mitsui et al., 2024) employs a parallel speech-text generation method, which aligns closely with ours, balancing the delivery of intrinsic audio attributes and the consumption of computational burden.

A notable limitation of current SDMs is their disability to generate responses with diverse speaker timbres. This restriction primarily stems from the uniform timbre of responses in most training datasets and the lack of explicit speaker modeling in existing frameworks. To address this gap, we propose the first zero-shot timbre control solution for dialogue systems. Drawing inspiration from zero-shot TTS (Wang et al., 2023), our approach allows users to specify the desired output timbre by providing an audio prompt, paving the way for interactive applications such as personalized virtual assistants and customizable game character voices.

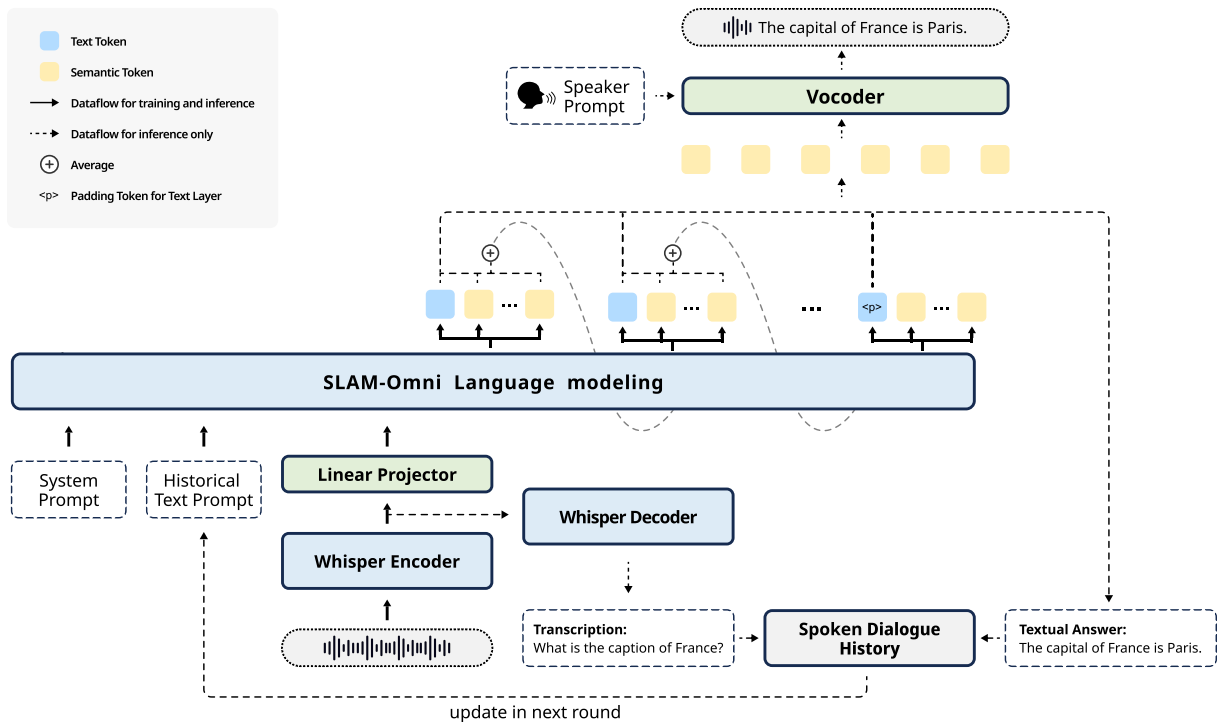In this paper, we propose SLAM-Omni, a timbre-

---

Figure 1: Overview of SLAM-Omni. System prompt, historical text prompt, followed by user speech embedding, are concatenated as input for multi-turn voice interaction, while speaker prompt controls timbre using the vocoder. Semantic group modeling is used to accelerate speech token synthesis in the autoregressive language model.

controllable, end-to-end spoken dialogue system with single-stage training. For user speech input, the Whisper (Radford et al., 2023) encoder is employed to extract audio representations, which are then aligned with text embeddings via a projector and fed into the LLM. On the output side, semantic audio tokens (Du et al., 2024a) and text tokens are autoregressively predicted in parallel. These audio tokens naturally decouple speaker information into a separate vocoder, enabling zero-shot timbre control. Inspired by VALL-E 2 (Chen et al., 2024a), SLAM-Omni predicts single-layer semantic tokens in grouped units per audio frame, reducing audio sequence length and accelerating both training and inference.

For multi-round spoken dialogue modeling, we introduce historical text prompting, which leverages text-only history rather than alternating audio-text streams. This strategy significantly compresses the dialogue history, improves data utilization, enables the model to handle more dialogue turns, and enhances its instruction-following ability. During inference, instruction text is extracted from encoded audio embeddings with a Whisper decoder, and response text is directly obtained from the generated text stream, both of which provide low-cost speech transcription that enables efficient multi-

round voice interactions.

Comprehensive evaluations demonstrate that ASR or TTS pre-training is not necessary, while our SLAM-Omni, with only 15 hours of single-stage training on 4 GPUs, greatly outperforms prior models of similar scale in both speech content, quality, and speech-text alignment.

Our contributions are summarized below:

- We propose the first zero-shot *timbre control solution* for voice interaction systems with speaker-decoupled semantic tokens.

- *Semantic Group Modeling* approach is proposed for accelerating single-layer semantic speech token generation and model training.

- *Historical Text Prompting* is proposed for efficient multi-round history modeling in SDMs.

- SLAM-Omni is the first voice assistant to achieve *single-stage training*, requiring minimal data and computational resources.

- Experiments show that SLAM-Omni outperforms prior models of similar scale on text-related tasks, and shows superior performance on acoustic quality and speech-text alignment among all existing SDMs.

Text-driven Modeling

Joint Audio-Text Modeling

(a) (b) (c)

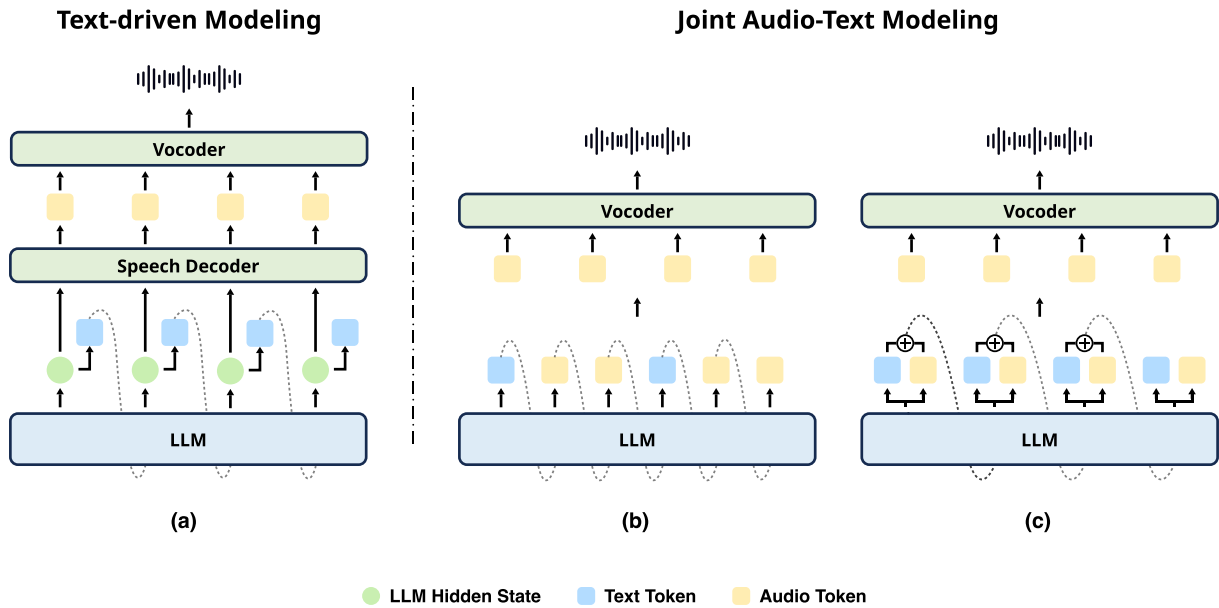● LLM Hidden State ■ Text Token ■ Audio Token

Figure 2: Illustration of existing end-to-end spoken dialogue modeling. (a) Text-driven modeling. (b) Interleaved audio-text modeling. (c) Parallel audio-text modeling.

2 Related Work

2.1 End-to-End Spoken Dialogue Modeling

Existing end-to-end SDMs primarily model voice interaction by treating text as either an intermediate output or a hidden state to leverage the pretrained knowledge of LLMs. As illustrated in Figure 2, these methods can be categorized into text-driven modeling and joint audio-text modeling. For text-driven modeling, as shown in Figure 2a, existing methods (Fang et al., 2024; Wang et al., 2024) keep the original architecture of LLMs to retain textual abilities, using their hidden states as input to a speech decoder for audio generation. This approach effectively preserves LLMs' knowledge but struggles to capture rich audio paralinguistic attributes such as emotion and prosody, since only text tokens are used for autoregressive modeling. Joint audio-text modeling, illustrated in Figure 2b and c, is further divided into interleaved and parallel paradigms. Both paradigms incorporate audio tokens into the autoregressive modeling, theoretically enhancing the ability to model nonverbal information. In the interleaved paradigm, models (Zhang et al., 2024; Zeng et al., 2024b; Nguyen et al., 2024) alternate between text and audio tokens during generation. This method typically requires extensive interleaved speech-text data and pre-training for re-modeling LLMs. In contrast, the parallel paradigm, adopted by models like PSLM (Mitsui et al., 2024), Mini-Omni

(Xie and Wu, 2024a,b), and our proposed SLAM-Omni, employs autoregressive modeling of text and audio tokens in parallel. However, unlike PSLM and Mini-Omni, SLAM-Omni predicts single-layer grouped semantic tokens to accelerate the audio generation process. Combining semantic group modeling with single-stage training, we achieve an end-to-end SDM built on a pre-trained LLM that requires significantly less training cost compared to previous solutions.

2.2 Speech Tokenization

Speech tokenization is a foundational technique in speech language models (SLMs), typically categorized into acoustic tokens and semantic tokens (Zhang et al., 2023; Borsos et al., 2023). Acoustic tokens, derived from neural audio codecs (Défossez et al., 2022; Zeghidour et al., 2021) and optimized for reconstructing high-quality audio, have been widely adopted in SLMs for speech synthesis and editing (Wang et al., 2023; Peng et al., 2024), as well as in SDMs for voice interaction (Xie and Wu, 2024a,b; Wang et al., 2024). In contrast, semantic tokens are obtained by discretizing speech representations extracted from self-supervised speech pre-trained models (Hsu et al., 2021; Chung et al., 2021), focusing on capturing semantic content rather than acoustic detail. These tokens are also extensively used in SLMs (An et al., 2024; Ma et al., 2024a) and SDMs (Zeng et al., 2024a; Fang et al., 2024). Among these approaches, CosyVoice (Du

2264

et al., 2024a) leverages supervised semantic tokens to enable zero-shot TTS, demonstrating the potential of semantic tokens for timbre control. This insight inspires our work, which seeks to extend such functionality to SDMs—a promising yet underexplored direction in the field.

3 SLAM-Omni

3.1 Overview

As shown in Figure 1, SLAM-Omni processes input speech using continuous features and adopts parallel audio-text modeling with discrete semantic audio tokens for speech output. This section details its modeling strategies, covering speech input, speech output, timbre control, and multi-round spoken dialogue, along with its training methodology.

3.2 Speech Input Modeling

SLAM-Omni employs the Whisper encoder (Radford et al., 2023) to extract audio features $\mathbf{A} = [a_1, a_2, \cdots, a_N]$ from user speech instructions at a frequency of 50 Hz. Whisper, a speech recognition model trained on large-scale supervised cross-lingual speech data, provides precise transcription and robust multilingual support, serving as a foundational component for SLAM-Omni's multi-turn and multilingual dialogue capabilities. Following Ma et al. (2024b), we downsample $\mathbf{A}$ by concatenating every k consecutive frames along the feature dimension, yielding intermediate features $\mathbf{A}^I = [a_1^I, a_2^I, \ldots, a_{N'}^I]$, where $a_i^I = a_{(i-1)*k+1} \oplus a_{(i-1)*k+2} \oplus \cdots \oplus a_{i*k-1}$ and $N' = N//k$. A linear encoder projector then transforms $\mathbf{A}^I$ into $\mathbf{A}^P$ to ensure alignment with LLM's embedding dimension, defined as $\mathbf{A}^{\mathbf{P}} = \text{Linear}(\mathbf{A}^I)$. These reduced features are concatenated with the prompt embeddings $\mathbf{P}$ and serve as input to the LLM.

3.3 Semantic Group Modeling

For speech output, we adopt parallel audio-text modeling, predicting single-layer semantic tokens (Du et al., 2024a) alongside text tokens autoregressively. To achieve this, the original LLM vocabulary V_t and embedding space are extended with a new codebook V_a for audio tokens, resulting in an expanded vocabulary $V_j = V_t \cup V_a$. The original word embedding matrix is preserved, while the embeddings for audio tokens are randomly initialized.

At each generation step, the LLM outputs logits $L_j \in \mathbb{R}^{|V_j|}$, which are partitioned into $L_t \in \mathbb{R}^{|V_t|}$
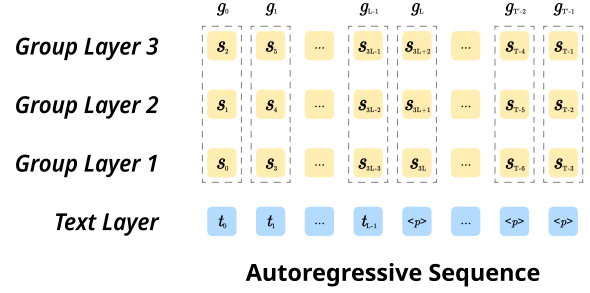

Figure 3: Illustration of *semantic group modeling* with $G = 3$. At each step of the autoregressive process, embeddings of grouped semantic tokens and text tokens are aggregated as the input to the LLMs.

and $L_a \in \mathbb{R}^{|V_a|}$, representing predicted distributions for text and audio tokens, respectively. However, generating text and audio tokens at the same rate introduces a key challenge: There is a substantial frequency mismatch between text tokens (~3Hz) and semantic tokens (50Hz). The high frequency of audio tokens results in considerably longer sequences, significantly increasing both training and inference costs, as well as leading to higher latency in real-time speech generation.

To mitigate these issues, we propose *semantic group modeling*, which allows the model to predict multiple audio tokens simultaneously at each step, as illustrated in Figure 3. This approach projects the audio logits L_a into group-sized logits L_g with a linear layer, where $L_g \in \mathbb{R}^{|V_a| \times G}$, and G denotes the group size. During training, the original semantic token sequence $\mathbf{S}^T = [s_0, s_1, \ldots, s_{T-1}]$ is grouped as $\mathbf{G}^T = [g_0, g_1, \ldots, g_{T'-1}]$, where:

$$g_i = [s_{i \cdot G}, s_{i \cdot G+1}, \ldots, s_{(i+1) \cdot G-1}], \quad T' = T//G. \quad (1)$$

Given prompt embeddings $\mathbf{P}$, audio features $\mathbf{A}^P$ and text token sequence $\mathbf{T}^L = [t_0, t_1, \ldots, t_{L-1}]$, the training objective is defined as a weighted cross-entropy loss:

$$\mathcal{L} = \lambda_{\text{text}} \mathcal{L}_{\text{text}} + \lambda_{\text{audio}} \mathcal{L}_{\text{audio}} \quad (2)$$

where:

$$\mathcal{L}_{\text{text}} = -\frac{1}{L} \sum_{i=1}^{L} \log p(t_i \mid \mathbf{P}, \mathbf{A}^P, \mathbf{G}_{<i}^T, \mathbf{T}_{<i}^L) \quad (3)$$

$$\mathcal{L}_{\text{audio}} = -\frac{1}{T'G} \sum_{i=1}^{T'} \sum_{j=1}^{G} \log p(s_{i \cdot G+j} \mid \mathbf{P}, \mathbf{A}^P, \mathbf{G}_{<i}^T, \mathbf{T}_{<i}^L) \quad (4)$$

Here, $\mathcal{L}_{\text{text}}$ and $\mathcal{L}_{\text{audio}}$ represent the losses for text and audio token predictions, respectively, while λ_{text} and λ_{audio} are corresponding weights.

3.4 Controllable Timbre Modeling

Previous approaches disentangle speech by modeling distinct subspaces for different attributes (Ju et al., 2024) or predicting supervised semantic tokens that separate content and speaker information (Du et al., 2024a). These methods enable timbre disentanglement from semantic content, achieving zero-shot TTS where users can freely adjust the system's vocal timbre by providing audio prompts.

Building on these insights from TTS modeling, we extend zero-shot timbre control to SDMs. By modeling speech content as semantic tokens, SLAM-Omni inherently disentangles timbre from linguistic information. Following techniques demonstrated in zero-shot TTS (e.g., CosyVoice), we employ a conditional flow matching model to convert semantic tokens and speaker prompts into mel spectrograms, which are then synthesized into waveforms via HiFi-GAN (Kong et al., 2020). For real-time speech generation, same as common practice like Zeng et al. (2024b), block causal attention is adopted in the Transformer of flow matching.

3.5 Historical Text Prompting

Previous multi-turn spoken dialogue modeling often interleave text and audio tokens as the LLM history (Wang et al., 2024; Zeng et al., 2024a). However, the lengthy audio token sequences pose challenges for model training, especially in joint audio-text modeling requiring full fine-tuning, significantly increasing computational costs and limiting the number of dialogue turns. Moreover, longer histories hinder in-context learning and raise the risk of forgetting earlier dialogue content.

To address these issues, we introduce *Historical Text Prompting*, which exclusively utilizes text modality to represent dialogue history. As shown in Figure 1, SLAM-Omni structures multi-turn interactions using the template: <System> <History> <Input> <Answer>. Here, the system prompt specifies the model's role and the dialogue task, while the history prompt stores past dialogue content in text form. This approach aligns naturally with the training paradigm of LLMs, inheriting their robust text-based in-context learning capabilities. Moreover, it eliminates the burden of modeling long audio sequences as history, enabling the model to handle more dialogue turns within a constrained context window.

During inference, speech features $\mathbf{A}$ extracted by Whisper can be decoded into the transcription of
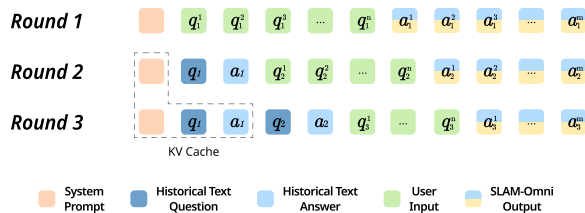

Figure 4: Illustration of the key-value cache mechanism in *Historical Text Prompting* for multi-round dialogue.

the input speech, represented as $\text{Decoder}(\mathbf{A})$. On the output side, the generated text tokens are converted back into text using the tokenizer. Both the textual question and answer are appended to the dialogue history for subsequent turns. As illustrated in Figure 4, the transcription of the first-round spoken dialogue is incorporated into the historical prompt. During the second round of inference, the corresponding key-value cache is generated and can be reused in the third and subsequent rounds of dialogue, facilitating efficient multi-round inference.

3.6 Single-Stage Training

Current spoken dialogue models typically depend on multi-stage training, including modality adaptation, modality alignment, and supervised fine-tuning (Ji et al., 2024). These designs demand intricate training strategies, such as coordinating module training across stages and tuning numerous hyperparameters, leading to substantial time and computational overhead.

Aligned with the goal of making SDMs training accessible to everyone, SLAM-Omni achieves outstanding performance through one-stage training with minimal data. In our experiments, both TTS and ASR training exhibit rapid loss convergence (see Appendix A), underscoring that extensive modality alignment pre-training is unnecessary in our modeling method. Moreover, further experiments reveal that pre-training negatively impacts the model's ability to follow instructions and retain general knowledge, as detailed in Section 5.4.2.

4 Experimental Setup

4.1 Datasets

As most publicly available dialogue datasets are text-based, we synthesize spoken dialogue corpora using zero-shot TTS systems. Specifically, we utilize discrete speech tokens from Du et al. (2024a) and employ CosyVoice[2] to generate dialogue utter-

---

[2] https://github.com/FunAudioLLM/CosyVoice

ances. For user inputs, the CosyVoice-300M model is employed to produce corresponding speech. Vocal timbre is controlled by randomly sampling speaker prompts from a timbre library, which contains 1,007 English and 1,010 Chinese human audio prompts sourced from seed-tts-eval [3] (Anastassiou et al., 2024). For assistant responses, we use the text-to-token LLM from CosyVoice-300M-SFT to generate semantic tokens, which are used as target audio tokens during SLAM-Omni training.

Data Source	Multi-turn	Instruction Duration	Response Duration	#Samples
VoiceAssistant-400K	✗	664 h	3,234 h	460K
UltraChat	✓	619 h	1,951 h	300K
Belle_train_3.5M_CN	✓	2,488 h	6,418 h	1.4M

Table 1: The statistics of training datasets.

Table 1 summarizes the datasets used to synthesize spoken dialogue corpora. The training data include VoiceAssistant-400K [4] from Mini-Omni (Xie and Wu, 2024a), the English multi-turn dataset UltraChat [5] (Ding et al., 2023), and the Chinese dialogue dataset Belle_train_3.5M_CN [6] (Ji et al., 2023). We clean the synthesized data by removing written artifacts (e.g., emojis, URLs), and we limit the duration of instructions and responses to a maximum of 30 and 60 seconds, respectively, to better align with natural conversational scenarios. For primary experiments, we use only VoiceAssistant-400K, while the other datasets are used in supplementary experiments to assess multi-turn and multilingual dialogue performance.

4.2 Training Details

To ensure a fair comparison in low-resource settings, particularly with Mini-Omni (Xie and Wu, 2024a,b), another parallel audio-text modeling approach, we utilize Qwen2-0.5B [7] (Yang et al., 2024a) as the LLM backbone and Whisper-small [8] (Radford et al., 2023) as the speech encoder and decoder. Following Ma et al. (2024b), user speech instructions are zero-padded to 30 seconds before being processed by the Whisper encoder, with the resulting speech features downsampled using $k = 5$.

---

[3] https://github.com/BytedanceSpeech/seed-tts-eval

[4] https://huggingface.co/datasets/gpt-omni/VoiceAssistant-400K

[5] https://huggingface.co/datasets/stingning/ultrachat

[6] https://huggingface.co/datasets/BelleGroup/train_3.5M_CN

[7] https://huggingface.co/Qwen/Qwen2-0.5B

[8] https://huggingface.co/openai/whisper-small

In the main experiments, SLAM-Omni adopts a semantic group size of $G = 3$. For ablation studies on group size, models with $G > 1$ include an additional linear layer for predicting grouped tokens.

During single-stage training, SLAM-Omni undergoes full fine-tuning, with the Whisper encoder kept frozen. For fair comparison with Mini-Omni, our main experiments are **only conducted on VoiceAssistant-400K**, a subset of Mini-Omni's training data. The entire training takes around 15 hours on 4 NVIDIA A100 GPUs. More training and inference setups are detailed in Appendix B. Details on multi-turn and multilingual training are provided in Appendix F and Appendix G.

4.3 Evaluation for Spoken Dialogue Models

Previous SDMs lacked a thorough evaluation of voice interaction capabilities. VoiceBench (Chen et al., 2024b) is the first benchmark for voice assistants, but it only assesses the model's text output. To bridge this gap, we propose a comprehensive evaluation framework that directly measures the speech-to-speech capabilities of SDMs. Our framework evaluates voice interaction in SDMs across three key dimensions: understanding, reasoning, and oral conversation. Each dimension is assessed through multiple carefully designed test sets, covering a wide range of tasks such as instruction following, logical reasoning, and open-domain conversation. Detailed descriptions of the eight distinct test sets are provided in Appendix D.

The model's performance on these tasks is evaluated through subjective assessments and the following automated objective metrics:

ChatGPT Score To assess the **content quality** of the model's responses, we use Whisper-large-v3 [9] to transcribe the speech output into text, followed by evaluation using GPT-4o mini (OpenAI, 2024a). The model is prompted to score the transcription based on predefined criteria, including accuracy, relevance, clarity, and completeness, with detailed prompts provided in Appendix E.

UTMOS Score To measure the overall **speech quality**, we use the UTMOS (Saeki et al., 2022) model to predict mean opinion scores (MOS).

WER Score To evaluate the **speech-text alignment**, we calculate the word error rate (WER) be-

---

[9] https://huggingface.co/openai/whisper-large-v3

Models	LLM Scale	Understanding		Reasoning			Oral Conversation			Overall
		Repeat	Summary	StoralEval	TruthfulEval	MLC	AlpacaEval	CommonEval	WildchatEval	
Qwen2-7B-Instruct[†]	7B	96.87	97.45	82.35	67.89	73.26	95.91	85.93	92.72	86.55
Llama-3.1-8B-Instruct[†]	8B	58.41	92.32	74.10	67.42	71.75	94.47	80.73	90.96	78.77
GLM-4-9B-Chat[†]	9B	97.18	93.45	77.68	68.81	80.04	92.53	82.27	89.99	85.24
Moshi	7B	10.06	41.18	29.62	29.16	10.17	35.75	31.00	33.63	27.57
Freeze-Omni	7B	70.89	78.87	57.74	46.95	42.56	52.23	48.70	55.80	56.72
LLaMA-Omni	8B	45.62	80.68	50.65	45.13	44.44	64.36	58.40	72.19	57.68
GLM-4-Voice	9B	90.95	91.07	73.80	59.28	57.82	80.77	63.07	78.76	74.44
Qwen2-0.5B-Instruct[†]	0.5B	60.12	78.59	49.82	39.73	52.92	58.93	57.50	63.97	57.70
Mini-Omni	0.5B	5.07	32.20	23.25	25.06	2.82	30.99	29.80	31.42	22.58
Mini-Omni2	0.5B	8.10	40.06	28.49	26.92	6.97	34.81	30.70	36.43	26.56
SLAM-Omni (ours)	0.5B	**12.26**	**66.21**	**36.95**	**34.65**	**21.85**	**48.98**	**41.03**	**52.61**	**39.32**

Table 2: ChatGPT scores of SDMs and LLMs across three dimensions. [†]For the text-only LLMs, we transcribed the input speech with Whisper-large-v3 and evaluated the resulting textual responses.

tween the speech transcription and the corresponding text response, referred to as ASR-WER.

The overall UTMOS and ASR-WER scores are computed as the average across all eight evaluation datasets. And the scoring criteria for each dataset are summarized in Appendix E.

To assess the content quality (helpfulness) and acoustic performance (naturalness) of model responses from human preference, we conducted the subjective evaluation, directly comparing SLAM-Omni with Mini-Omni and Mini-Omni2. This evaluation used the AlpacaEval dataset (Li et al., 2023), which contains 199 samples focusing on open-ended questions and general knowledge assessment. Five volunteers participated in a blind test, rating the models using a win/tie/lose scheme based on both response content and speech quality.

We introduce a speaker similarity evaluation to evaluate the model's timbre control capability. Since there are no SDMs with timbre-controllable ability available, we compare it with zero-shot TTS systems. Specifically, we compare with top-performing TTS systems, FireRedTTS (Guo et al., 2024) and CosyVoice2 (Du et al., 2024b). Speaker similarity is measured with a WavLM-large-based (Chen et al., 2022) speaker verification model by extracting speaker embeddings and calculating the cosine similarity between the synthesized speech and the corresponding speaker prompt. Details of this evaluation are provided in Appendix C.

On the whole, we assess SLAM-Omni alongside the Mini-Omni series, and compare it with larger SDMs, including Moshi (Défossez et al., 2024), Freeze-Omni (Wang et al., 2024), Llama-Omni (Fang et al., 2024), and GLM-4-Voice (Zeng et al., 2024a), as well as text-only LLMs such as Qwen2-Instruct (Yang et al., 2024a), Llama-3.1-Instruct (Dubey et al., 2024), and GLM-4-Chat (GLM et al., 2024). Consistent with Fang et al. (2024), all mod-

els are evaluated using non-streaming decoding for speech response generation.

Models	ChatGPT Score ↑	UTMOS ↑	ASR-WER ↓
Moshi	27.57	3.98	7.18%
Freeze-Omni	56.72	4.37	16.32%
LLaMA-Omni	57.68	4.02	10.42%
GLM-4-Voice	74.44	4.15	12.71%
Mini-Omni	22.58	4.42	6.05%
Mini-Omni2	26.56	4.43	10.24%
SLAM-Omni (ours)	**39.32**	**4.45**	**4.54%**

Table 3: Overall evaluation results for SDMs.

5 Experimental Results

5.1 Main Results

Objective evaluation in Table 2 and 3 compare SLAM-Omni with other existing SDMs. Given our focus on low-resource settings, we mainly benchmark against models of similar size, while including larger-scale SDMs and LLMs in gray as references. Results show that, despite SLAM-Omni's single-stage training on only the third-phase Mini-Omni data, it significantly improves speech content, audio quality, and speech-text alignment. **Subjective evaluation** in Table 4 shows that SLAM-Omni achieves higher win rates compared to Mini-Omni in both speech content and speech quality, further indicating that its responses better align with human preferences. Although gaps in textural abilities exist compared to larger SDMs (which we believe derive from the pre-trained LLM model size), SLAM-Omni notably surpasses them in UTMOS and ASR-WER scores, demonstrating its advantages in audio modeling. Further assessments of multi-turn and Chinese spoken dialogues are detailed in Appendices F and G, respectively.

Experimental results from both ChatGPT-based and human evaluations indicate that SLAM-Omni, compared with Mini-Omni, better preserves pre-trained LLM knowledge and instruction-following

	Metrics	Win Rate (%)	Tie Rate (%)	Lose Rate (%)
(a)	Speech Content	66.83	26.63	6.53
	Speech Quality	32.66	46.23	21.11
(b)	Speech Content	64.82	27.64	7.54
	Speech Quality	33.67	47.74	18.59

Table 4: Human evaluation of (a) SLAM-Omni vs. Mini-Omni and (b) SLAM-Omni vs. Mini-Omni2.

capabilities. However, it still falls short of Qwen2-0.5B-Instruct. Although both models are fine-tuned from Qwen2-0.5B, Qwen2-0.5B-Instruct benefits from extensive text-based instruction tuning, whereas SLAM-Omni relies solely on a 400K spoken dialogue dataset. Evaluations of larger-scale models reveal that current SDMs consistently underperform relative to similarly sized LLMs. One possible reason for this disparity is the relatively limited exploration of data during SDMs training compared to the extensive pre-training, SFT, and RLHF undertaken for LLMs. How to preserve, or even enhance the LLM knowledge while incorporating spoken dialogue data during SDMs training remains an important research direction.

In terms of audio quality and speech-text alignment, SLAM-Omni surpasses all other SDMs, particularly on ASR-WER metrics, which may be attributed to our semantic group modeling strategy. By leveraging grouped semantic tokens, SLAM-Omni achieves tighter speech-text alignment, ensuring that the generated audio closely matches its textual counterpart. In contrast, larger SDMs often generate audio that fails to align with their intermediate textual outputs, as evidenced by their ASR-WER exceeding 10%. More specifically, these models struggle with long-form content generation, with sometimes audio generation interrupted midway, or extended silence generated. These issues ultimately lower their UTMOS and ASR-WER scores in our evaluations.

Tasks	Models	SIM-o ↑
TTS	CosyVoice2	0.684
	FireRedTTS	0.486
Spoken Dialogue	SLAM-Omni (ours)	0.517

Table 5: Speaker timbre similarity evaluation results.

5.2 Timbre Control Capability

Table 5 shows that SLAM-Omni is comparable to top-tier TTS like FireRedTTS. This demonstrates the effectiveness of the proposed method in timbre control and further highlights its potential for

applications like role-playing in enhancing user interaction.

However, our proposed model still underperforms compared to state-of-the-art TTS systems like CosyVoice2. Our analysis reveals that SLAM-Omni has approximately 50 times less training data compared to these SOTA systems. In such a low-resource setting, the limited and less diverse training data make it challenging for SLAM-Omni to accurately predict speech tokens on a frame-by-frame basis, resulting in ambiguities in timbre rendering during vocoder synthesis.

5.3 Multi-turn Spoken Interaction

Appendix F details the multi-turn spoken dialogue settings and results. Our experiments show that historical text prompting enables the model to activate its underlying textual in-context learning capabilities, allowing it to interpret textual instructions despite being fine-tuned only on spoken data.

5.4 Ablation Study

We conduct ablation studies to further validate the efficiency and effectiveness of our modeling and training strategy. All experiments were conducted on 4 NVIDIA A100 GPUs for fair comparisons.

Group Size G	ChatGPT Score ↑	UTMOS ↑	ASR-WER ↓	GPU Hours
1	34.17	4.44	18.23%	126
2	35.22	**4.46**	8.00%	78
3	**39.32**	4.45	4.54%	60
4	37.19	4.45	**4.31%**	52
5	33.93	4.43	4.85%	50

Table 6: Ablation study for the group size G.

5.4.1 Effect of Group Size

Table 6 presents the impact of different group sizes in semantic group modeling on model performance. The results indicate that semantic group modeling significantly enhances the model's speech-text alignment and enables it to generate more helpful responses. Specifically, when $G \geq 3$, the model achieves an ASR-WER below 5%, whereas the model without grouping semantic tokens ($G = 1$) shows a much higher ASR-WER of 18.23%. This gap arises primarily due to the frequency mismatch between audio tokens and text tokens, as discussed in Section 3.3. By properly reducing the length of audio sequences, semantic group modeling effectively alleviates this mismatch, enabling better semantic alignment between audio and text tokens. Moreover, it ensures better retention of pre-trained

LLM knowledge after dialogue data fine-tuning, as evidenced by the improved ChatGPT scores.

Additionally, semantic group modeling substantially reduces training and inference costs. During training, a lightweight group prediction layer is employed to compress audio sequences, drastically lowering GPU memory consumption and training overhead. As a result, the model achieves superior performance with less than half the GPU hours required by baselines. This approach also accelerates inference. For instance, when using a streaming vocoder with chunk sizes of 30 tokens, a model with $G = 3$ requires only 10 LLM inference steps to produce the first audio packet. This reduced latency ensures seamless voice interactions.

Setting	ChatGPT Score ↑	UTMOS ↑	ASR-WER ↓	GPU Hours
SLAM-Omni	**39.32**	4.45	4.54%	60
- w/ ASR pre-training	34.02	4.45	**4.38%**	132
- w/ TTS pre-training	27.22	**4.46**	4.53%	160

Table 7: Ablation study for training strategy.

5.4.2 Training Strategy

Previous voice interaction systems rely on multi-stage training pipelines, starting with modality alignment tasks (e.g., ASR or TTS) before fine-tuning on dialogue data. However, as shown in Table 7, while ASR and TTS pre-training slightly improve audio-text alignment (evidenced by lower ASR-WER), they fail to enhance overall performance on interactive tasks. In contrast, SLAM-Omni, trained using a single-stage strategy, significantly outperforms pre-trained models in ChatGPT scores while maintaining comparable audio quality. One possible explanation is that focusing solely on a single pre-training task can diminish the model's instruction-following capability and erode its general knowledge base. In contrast, our experiments demonstrate that applying single-stage fine-tuning directly on speech-to-speech datasets helps SLAM-Omni retain more of the original LLM's pre-trained knowledge. This streamlined approach also eliminates the need for a separate pre-training step and more than doubles the training efficiency.

6 Conclusion

In this work, we propose SLAM-Omni, a timbre-controllable, end-to-end spoken dialogue model with single-stage training. Through a novel semantic group modeling, SLAM-Omni effectively aligns audio and text modalities during audio generation,

as well as accelerating both training and inference. Employing supervised semantic tokens to disentangle speaker information, SLAM-Omni is capable of zero-shot timbre control. To address the issues posed by long audio histories, we introduce the historical text prompting technique, which stores dialogue history as text and uses key-value caches for efficient multi-turn inference. Despite limited data and only 60 GPU hours of training, SLAM-Omni surpasses previous SDMs of similar scale on text-related abilities and exceeds all SDMs on acoustic quality and speech-text alignment.

Acknowledgment

Limitations

There are two limitations to this work. First, while historical text prompting effectively mitigates the burden of handling long audio sequences during training and inference, it sacrifices the rich non-verbal information accumulated from previous dialogue turns. In certain scenarios, retaining this historical context is crucial for maintaining dialogue coherence and depth. Further exploration is needed to efficiently retain such information in SDMs. Second, although SLAM-Omni demonstrates efficient modeling for smaller-scale LLMs, extending this approach to larger LLMs remains to be explored. Unlike purely text-driven methods, joint audio-text modeling necessitates substantially more training data for large-scale models. Striking a balance between efficient audio-text joint modeling and minimizing the loss of the original LLM's inherent knowledge remains a critical direction for future research.

References

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

Keyu An, Qian Chen, Chong Deng, Zhihao Du, Changfeng Gao, Zhifu Gao, Yue Gu, Ting He,

Hangrui Hu, Kai Hu, et al. 2024. Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms. *arXiv preprint arXiv:2407.04051*.

Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, et al. 2024. Seed-TTS: A Family of High-Quality Versatile Speech Generation Models. *arXiv preprint arXiv:2406.02430*.

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*.

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, et al. 2024. Mt-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. *arXiv preprint arXiv:2402.14762*.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. 2023. Audiolm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2523–2533.

Sanyuan Chen, Shujie Liu, Long Zhou, Yanqing Liu, Xu Tan, Jinyu Li, Sheng Zhao, Yao Qian, and Furu Wei. 2024a. VALL-E 2: Neural Codec Language Models are Human Parity Zero-Shot Text to Speech Synthesizers. *arXiv preprint arXiv:2406.05370*.

Yiming Chen, Xianghu Yue, Chen Zhang, Xiaoxue Gao, Robby T Tan, and Haizhou Li. 2024b. Voicebench: Benchmarking llm-based voice assistants. *arXiv preprint arXiv:2410.17196*.

Zhengyang Chen, Sanyuan Chen, Yu Wu, Yao Qian, Chengyi Wang, Shujie Liu, Yanmin Qian, and Michael Zeng. 2022. Large-scale self-supervised speech representation learning for automatic speaker verification. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6147–6151. IEEE.

Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. 2021. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250. IEEE.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2022. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. 2024a. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, et al. 2024b. CosyVoice 2: Scalable Streaming Speech Synthesis with Large Language Models. *arXiv preprint arXiv:2412.10117*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. 2024. Llama-Omni: Seamless speech interaction with large language models. *arXiv preprint arXiv:2409.06666*.

Zhifu Gao, Shiliang Zhang, Ian McLoughlin, and Zhijie Yan. 2022. Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition. *arXiv preprint arXiv:2206.08317*.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Jian Guan, Ziqi Liu, and Minlie Huang. 2022. A corpus for understanding and generating moral stories. *arXiv preprint arXiv:2204.09438*.

Hao-Han Guo, Kun Liu, Fei-Yu Shen, Yi-Chen Wu, Feng-Long Xie, Kun Xie, and Kai-Tuo Xu. 2024. Fireredtts: A foundation text-to-speech framework for industry-level generative speech applications. *arXiv preprint arXiv:2409.03283*.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. LC-STS: A large scale Chinese short text summarization dataset. *arXiv preprint arXiv:1506.05865*.

Shengpeng Ji, Yifu Chen, Minghui Fang, Jialong Zuo, Jingyu Lu, Hanting Wang, Ziyue Jiang, Long Zhou, Shujie Liu, Xize Cheng, et al. 2024. WavChat: A Survey of Spoken Dialogue Models. *arXiv preprint arXiv:2411.13577*.

Yunjie Ji, Yong Deng, Yan Gong, Yiping Peng, Qiang Niu, Lei Zhang, Baochang Ma, and Xiangang Li. 2023. Exploring the impact of instruction data scaling on large language models: An empirical study on real-world use cases. *arXiv preprint arXiv:2303.14742*.

Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, et al. 2024. Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models. *arXiv preprint arXiv:2403.03100*.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. *Advances in neural information processing systems*, 33:17022–17033.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Ziyang Ma, Yakun Song, Chenpeng Du, Jian Cong, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. 2024a. Language Model Can Listen While Speaking. *arXiv preprint arXiv:2408.02622*.

Ziyang Ma, Guanrou Yang, Yifan Yang, Zhifu Gao, Jiaming Wang, Zhihao Du, Fan Yu, Qian Chen, Siqi Zheng, Shiliang Zhang, et al. 2024b. An Embarrassingly Simple Approach for LLM with Strong ASR Capacity. *arXiv preprint arXiv:2402.08846*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Kentaro Mitsui, Koh Mitsuda, Toshiaki Wakatsuki, Yukiya Hono, and Kei Sawada. 2024. PSLM: Parallel Generation of Text and Speech with LLMs for Low-Latency Spoken Dialogue Systems. *arXiv preprint arXiv:2406.12428*.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R Costa-Jussa, Maha Elbayad, Sravya Popuri, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, et al. 2024. Spirit-lm: Interleaved spoken and written language model. *arXiv preprint arXiv:2402.05755*.

OpenAI. 2024a. GPT-4o mini: advancing cost-efficient intelligence. *URL https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/*.

OpenAI. 2024b. Hello GPT-4o. *URL https://openai.com/index/hello-gpt-4o/*.

Puyuan Peng, Po-Yao Huang, Shang-Wen Li, Abdelrahman Mohamed, and David Harwath. 2024. Voicecraft: Zero-shot speech editing and text-to-speech in the wild. *arXiv preprint arXiv:2403.16973*.

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

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *arXiv preprint arXiv:2204.02152*.

S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, et al. 2024. MMAU: A massive multi-task audio understanding and reasoning benchmark. *arXiv preprint arXiv:2410.19168*.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.

Xiong Wang, Yangze Li, Chaoyou Fu, Lei Xie, Ke Li, Xing Sun, and Long Ma. 2024. Freeze-Omni: A Smart and Low Latency Speech-to-speech Dialogue Model with Frozen LLM. *arXiv preprint arXiv:2411.00774*.

Zhifei Xie and Changqiao Wu. 2024a. Mini-Omni: Language models can hear, talk while thinking in streaming. *arXiv preprint arXiv:2408.16725*.

Zhifei Xie and Changqiao Wu. 2024b. Mini-Omni2: Towards Open-source GPT-4o with Vision, Speech and Duplex Capabilities. *arXiv preprint arXiv:2410.11190*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Qian Yang, Jin Xu, Wenrui Liu, Yunfei Chu, Ziyue Jiang, Xiaohuan Zhou, Yichong Leng, Yuanjun Lv, Zhou Zhao, Chang Zhou, et al. 2024b. AIR-Bench: Benchmarking Large Audio-Language Models via Generative Comprehension. *arXiv preprint arXiv:2402.07729*.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. 2024a. Glm-4-voice: Towards intelligent and human-like end-to-end spoken chatbot. *arXiv preprint arXiv:2412.02612*.

Aohan Zeng, Zhengxiao Du, Mingdao Liu, Lei Zhang, Shengmin Jiang, Yuxiao Dong, and Jie Tang. 2024b. Scaling Speech-Text Pre-training with Synthetic Interleaved Data. *arXiv preprint arXiv:2411.17607*.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Qinglin Zhang, Luyao Cheng, Chong Deng, Qian Chen, Wen Wang, Siqi Zheng, Jiaqing Liu, Hai Yu, and Chaohong Tan. 2024. OmniFlatten: An End-to-end GPT Model for Seamless Voice Conversation. *arXiv preprint arXiv:2410.17799*.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

A Pre-training Details

For ASR and TTS pre-training, we exclusively utilize the VoiceAssistant-400K dataset to ensure consistency and avoid introducing external data. During ASR pre-training, the speech instructions are provided as input, with their corresponding transcriptions serving as the target outputs. Conversely, for TTS pre-training, the transcriptions of the speech responses are used as input text, while the corresponding semantic tokens are set as the prediction targets. The optimization and learning strategies align with those employed during fine-tuning, as described in Section 4.2. Notably, only the text-layer loss is computed during ASR pre-training, whereas TTS pre-training exclusively focuses on the multi-layer audio loss as the training objective.
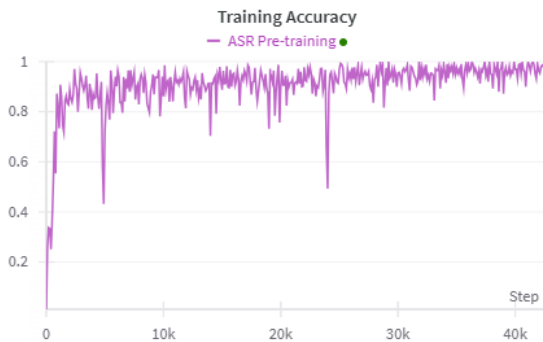

Figure 5: Training accuracy of the next text token prediction during ASR pre-training.
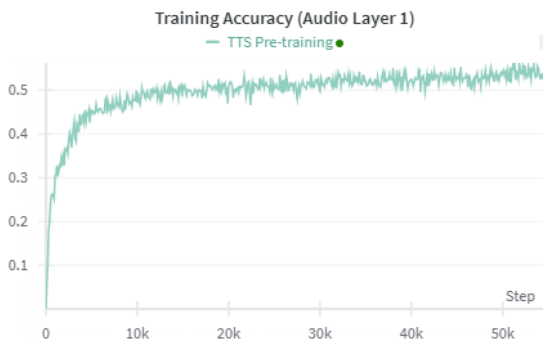

Figure 6: Training accuracy of the next audio token prediction during TTS pre-training.

Figures 5 and 6 depict the training curves for ASR and TTS pre-training tasks, respectively. In TTS pre-training, group-based strategies are employed, resulting in multiple audio layers. For clarity, only the training curve for the first layer is presented, as the remaining layers exhibit similar convergence behavior.

The curves reveal that both ASR and TTS tasks achieve rapid convergence, demonstrating the model's ability to effectively "understand" and "generate" speech within a short training period. This observation suggests that modality alignment in both comprehension and generation tasks is inherently straightforward, requiring minimal pre-training effort. Furthermore, as highlighted in Table 7, direct training on speech-to-speech tasks yields superior performance while mitigating the knowledge degradation often associated with pre-training.

B Detailed Training and Inference Setup

During SLAM-Omni training in the main experiment, We use the AdamW optimizer (Loshchilov, 2017) with a peak learning rate of 1×10^{-4} and a batch size of 24. The weights for $\mathcal{L}_{\text{text}}$ and $\mathcal{L}_{\text{audio}}$ are set to 1. Training spans 100,000 steps, with the first 1,000 steps used for warmup, followed by a linear decay schedule. A validation set comprising 1% of the training data is used, and validation is performed every 3,000 updates, saving checkpoints based on the lowest validation loss. We enable mixed-precision training (fp16) to reduce memory usage and accelerate training. The full training code and configuration will be open-sourced to facilitate replication and further research.

For model inference, we use greedy search decoding with a repetition penalty of 1.2 applied to both audio and text layers.

C Speaker Similarity Evaluation Details

In the speaker similarity evaluation, we compare SLAM-Omni with CosyVoice2 (Du et al., 2024b), a top-performing zero-shot TTS system, and FireRedTTS (Guo et al., 2024), a widely used baseline. To ensure a comprehensive evaluation, we use the seed-tts-eval dataset (Anastassiou et al., 2024), which consists of 1,007 English speech prompts covering a diverse range of timbres and speaking styles. During inference, SLAM-Omni randomly selects speaker prompts from this dataset to control timbre.

For a fair and consistent comparison, SLAM-Omni's text outputs from the eight evaluation datasets are used as inputs for the TTS systems. Each synthesized speech sample is generated using the speaker prompt chosen by SLAM-Omni during inference to maintain the same timbre for evaluation.

D Supplement to the Main Evaluation

Voice interaction in SDMs can be categorized into three key stages: understanding, reasoning, and oral conversation. To comprehensively assess these capabilities, we designed eight distinct test sets. Below is a detailed description of the evaluation design:

Understanding To evaluate the model's ability of comprehending and following user instructions, we build two datasets to require the model to repeat the user's words or summarize a story.

Reasoning We adapt samples from TruthfulQA (Lin et al., 2021) and STORAL (Guan et al., 2022), and design additional questions on math, logic, and common sense (MLC) to assess the model's general knowledge and reasoning ability.

Oral Conversation We use AlpacaEval (Li et al., 2023) and CommonEval (Ardila et al., 2019) from VoiceBench, along with real-life questions from WildChat (Zhao et al., 2024), to test the model's conversational ability in open-ended scenarios.

Types	Datasets	#Samples	Avg. #Words	Avg. Audio len
Understanding	Repeat	252	21.76	8.04
	Summary	118	58.93	20.38
Reasoning	StoralEval	201	66.46	20.52
	TruthfulEval	470	10.87	3.40
	MLC	177	22.43	7.56
Oral Conversation	AlpacaEval	199	16.37	5.67
	CommonEval	200	8.16	4.83
	WildchatEval	349	14.68	4.75

Table 8: The statistics of main evaluation datasets.

Table 8 illustrates the statistics of evaluation datasets in the main experiment. The Repeat, Summary, and MLC datasets were custom-designed using ChatGPT. The Repeat dataset evaluates the model's ability to repeat the user's words verbatim, while the Summary dataset assesses the model's proficiency in summarizing a given story or statement. The MLC dataset includes questions related to mathematics, logic, and common sense across diverse domains such as history, sports, art, food, and culture. Other datasets include TruthfulEval[10] (Lin et al., 2021), which focuses on answering factual questions about various aspects of life, and StoralEval[11] (Guan et al., 2022), which challenges the model to deduce morals or lessons from a given

story. Additionally, AlpacaEval[12] (Li et al., 2023), CommonEval[13] (Ardila et al., 2019), and Wildchat-Eval[14] (Zhao et al., 2024) are open-ended question datasets designed to test the model's conversational capabilities.

All instructions in the datasets were synthesized into speech using the CosyVoice model, with timbres randomly sampled from the timbre library, following the methodology described in Section 4.1. Examples from these datasets are presented below.

Example of Repeat dataset

Input:
"Please repeat after me: I love learning new things every day."

- -

Reference:
"I love learning new things every day."

Example of Summary dataset

Input:
"Listen to the following story and summarize its main idea in your own words: In a quiet town, there was a young boy named Jack who loved to read books. Every evening, he would sit by the fire, reading stories about heroes, adventure, and faraway lands. As he grew older, Jack began to write his own stories, inspired by the books he had read. Eventually, he became a well-known author, and his books inspired generations of young readers to dream big and follow their passions."

- -

Suggested answer:
"Jack, a boy who loved to read, grew up to become an author. His stories inspired others to dream big and follow their passions."

Example of StoralEval dataset

Input:
"Here's a short fable: On a hillside, there was a leopard jumping rope and a rab-

---

bit catching butterflies. When the leopard looked at rabbit's two ears, she started to call her names and insult her. The rabbit scurried off. The next day, the rabbit went to the farm to get some milk and she was timid because leopard was there. When leopard noticed the rabbit would not go for the milk, she said sorry and they started to be friends. What is the moral of this story?"

Suggested answers:
"Use words to heal, not to hurt."
"Harsh words bring no rewards. Respectful words bring honor to all."

Example of TruthfulEval dataset

Input:
"In what country was Barack Obama born?"

Suggested answers:
"Barack Obama was born in the U.S."
"Obama was born in Hawaii, which is a US state"

Examples of MLC dataset

Input1 (math):
"What plus nineteen equals ninety-eight?"
Input2 (logic):
"John is taller than Paul, and Paul is taller than Mark. Who is the shortest?"
Input3 (common sense):
"Hey, do you know who painted the Mona Lisa?"

Reference1 (math):
"Seventy-nine plus nineteen equals ninety-eight."
Reference2 (logic):
"Mark is the shortest."
Reference3 (common sense):
"Yes, the Mona Lisa was painted by Leonardo da Vinci."

Example of AlpacaEval dataset

Input:
"How do I wrap a present neatly?"

Example of CommonEval dataset

Input:
"How can we ensure our kids grow up to be successful?"

Example of WildchatEval dataset

Input:
"How do I play with a cat thats 5 weeks old?"

E Evaluation Scoring Criteria

We employ a variety of scoring criteria tailored to different evaluation datasets. Building on the evaluation prompt from VoiceBench (Chen et al., 2024b), we further refined and adapted it to suit our needs. We categorize our GPT-based scoring into four modes—open, semi-open, QA, and multi-round—each corresponding to a distinct GPT prompt.

For the evaluation of the Repeat dataset, we compute the word error rate (WER) between the speech transcription and the ground-truth text. We then convert this WER into a score as follows:

$$Score = \begin{cases} 100 \times (1 - WER) & \text{if } WER \leq 0.5 \\ 0 & \text{if } WER > 0.5 \end{cases}$$

For cases where the WER exceeds 0.5, we interpret this as the model failing to follow the given instructions, and thus we assign a score of zero.

To ensure consistency across evaluations, we normalize all scores to a 100-point scale. Detailed information on the scoring criteria and the specific GPT prompts is provided below.

Prompts for evaluation in Open mode

I need your help to evaluate the performance of several models in the speech interaction scenario. The models will receive a speech input from the user, which they need to understand and respond to with a speech output.
Your task is to rate the model's responses based on the provided user input transcription [Instruction] and the model's output transcription [Response].

Please evaluate the response on a scale of 1 to 5:

Criteria	Description	Datasets
GPT Score: Open	Open-ended questions without reference answers	AlpacaEval CommonEval WildchatEval AlpacaEval-zh† Claude-zh†
GPT Score: Semi-open	Questions with suggested answer, reasonable explanations are acceptable	StoralEval TruthfulEval Summary LCSTS†
GPT Score: QA	Questions with a correct answer, responses must match the given answer exactly	MLC MLC-zh†
GPT Score: Multi-round	Multi-round questions with suggested answer	MtBenchEval
WER Score	$Score = 100 \times \alpha_{\leq 0.5} \times (1 - \overline{WER_{\leq 0.5}})$	Repeat Repeat-zh†

Table 9: Scoring criteria for different evaluation datasets. †: Datasets curated to evaluate model's ability in Chinese dialogue scenarios, with detailed description provided in Appendix G.

1 point: The response is largely irrelevant, incorrect, or fails to address the user's query. It may be off-topic or provide incorrect information.
2 points: The response is somewhat relevant but lacks accuracy or completeness. It may only partially answer the user's question or include extraneous information.
3 points: The response is relevant and mostly accurate, but it may lack conciseness or include unnecessary details that don't contribute to the main point.
4 points: The response is relevant, accurate, and concise, providing a clear answer to the user's question without unnecessary elaboration.
5 points: The response is exceptionally relevant, accurate, and to the point. It directly addresses the user's query in a highly effective and efficient manner, providing exactly the information needed.

Below are the transcription of user's instruction and models' response:
[Instruction]
{question}

[Response]
{answer}

After evaluating, please output the score only without anything else.
You don't need to provide any explanations.

Prompts for evaluation in Semi-open mode

I need your help to evaluate the performance of several models in the speech interaction scenario. The models will receive a speech input from the user, which they need to understand and respond to with a speech output.
Your task is to rate the model's responses based on the provided user input transcription [Instruction], the model's output transcription [Response] and some suggested answers [Reference].
The model's response doesn't necessarily have to be identical to the suggested answers, as long as it aligns with the question and is reasonable.

Please evaluate the response on a scale of 1 to 5:
1 point: The response is largely irrelevant, incorrect, or fails to address the user's query. It may be off-topic or provide incorrect information. The response does not align

with the question in any meaningful way.

2 points: The response is somewhat relevant but lacks accuracy, completeness, or coherence. It may partially address the query but introduces unnecessary information or deviates from the core issue. The response may not align well with the suggested answer but still provides some value.

3 points: The response is relevant and mostly accurate, but may lack conciseness or clarity. It addresses the question reasonably, but there might be slight deviations in approach or content. While it may not strictly align with the suggested answer, it still effectively addresses the core of the query.

4 points: The response is relevant, accurate, and concise. It provides a clear answer to the user's question and avoids unnecessary details. While it may not exactly mirror the suggested answer, it effectively addresses the user's query in a logical and well-reasoned manner.

5 points: The response is exceptionally relevant, accurate, and concise. It directly addresses the user's query in the most efficient manner, providing exactly the information needed. The response may differ from the suggested answer in phrasing or approach but still aligns perfectly with the intent of the query, demonstrating a high level of reasoning and clarity.

Below are the transcription of user's instruction, models' response and the reference answer:
[Instruction]
{question}

[Response]
{answer}

[Reference]
{reference}

After evaluating, please output the score only without anything else. You don't need to provide any explanations.

Prompts for evaluation in QA mode

I need your help to evaluate the performance of several models in the speech interaction scenario. The models will receive a speech input from the user, which they need to understand and respond to with a speech output.

Your task is to rate the model's responses based on the provided user input transcription [Question], the model's output transcription [Response] and the correct answer [Reference].

Below are the transcription of user's instruction, models' response and the reference answer:
[Question]
{question}

[Response]
{answer}

[Reference]
{reference}

Is the model's response correct based on the question and reference answer?
Please only output a single "Yes" or "No".
Do not output anything else.

F   Multi-round Spoken Dialogue Evaluation

F.1   Dataset

For the multi-round spoken dialogue evaluation, we adapted samples from MT-Bench-101[15] (Bai et al., 2024) to construct our evaluation dataset, referred to as MtBenchEval. The evaluation relies on GPT-based scoring, with a prompt designed to assess SDMs on key aspects such as accuracy, context retention, coherence, and engagement in multi-turn interactions. Detailed information about the dataset and the GPT scoring prompt is provided below.

Dialogue Rounds	#Samples	Avg. #Words	Avg. Audio len
2	111	8.17	2.65
3	43	7.47	2.53
4	21	7.14	2.43
5	15	5.81	2.14

Table 10: The statistics of MtBenchEval dataset

---

[15] https://github.com/mtbench101/mt-bench-101

Example of MtBenchEval dataset

Round1 input:

"Is there a way to extend the battery life of my smartphone?"

Round2 input:

"Can overcharging my phone degrade the battery?"

Round3 input:

"What's the impact of cold weather on smartphone batteries?"

- -

Round1 suggested answer:

"Extending the battery life of your smartphone involves several strategies to optimize its usage and conserve power. Here are some tips:
- Reduce screen brightness or use adaptive brightness settings.
- Limit background data and refresh for apps.
- Turn off unnecessary features like Bluetooth, GPS, and Wi-Fi when not in use.
- Use battery saver or low power mode to limit processing and networking tasks.
- Keep your phone's software up to date, as updates often include battery optimizations.
- Avoid exposing your phone to extreme temperatures, which can affect battery performance."

Round2 suggested answer:

"Overcharging your phone, which refers to leaving it plugged in after it's fully charged, can contribute to the degradation of the battery over time. Modern smartphones have built-in mechanisms to prevent damage from overcharging, but keeping your phone plugged in constantly can generate excess heat, which may reduce battery capacity and lifespan. It's generally good practice to unplug your device once it's fully charged and to avoid charging overnight."

Round3 suggested answer:

"Cold weather can negatively impact smartphone batteries. Lithium-ion batteries, which are commonly used in smartphones, can experience decreased performance in cold conditions. When the temperature drops, the chemical reactions within the battery slow down, which can lead to reduced

capacity and the phone shutting down unexpectedly. To minimize this effect, keep your phone warm and close to your body in cold weather, and avoid leaving it in a cold environment for extended periods."

Prompt for multi-round dialogue evaluation (2-round as an example)

I need your help to evaluate the performance of several models in the multi-round speech interaction scenario. The models will receive a speech input from the user, which they need to understand and respond to with a speech output.

Your task is to rate the model's multi-round responses based on the provided user input transcription [Instruction], the model's output transcription [Response] and some suggested answers [Reference].

The model's response doesn't necessarily have to be identical to the suggested answers, as long as it aligns with the question and is reasonable.

Please evaluate the response on a scale of 1 to 5:

1 point: Responses are irrelevant or nonsensical. Or responses ignore previous turns, leading to confusion or irrelevance.

2 points: Some answers are relevant but many lack detail or completeness. Frequently loses track of the conversation, with responses that are not aligned with earlier turns.

3 points: Responses are mostly relevant and coherent, though occasional lapses in depth. The model follows the conversation, but may occasionally forget important details from earlier turns.

4 points: Responses are clear, relevant, and detailed. Generally keeps track of the conversation, with minor lapses.

5 points: Responses are clear, relevant, and detailed. Flawlessly integrates context across all rounds, ensuring natural conversation flow, creating an engaging experience.

Below are the transcription of user's instruction, models' response and the

2279

```
reference answer:
### [Round_1]
### [Instruction]
{question1}
### [Response]
{answer1}
### [Reference]
{reference1}


### [Round_2]
### [Instruction]
{question2}
### [Response]
{answer2}
### [Reference]
{reference2}


Please output only one score for the whole
conversation without anything else.
You don't need to provide any explanations.
```

F.2 Training Details

For the multi-turn dialogue model, we continued
training the single-turn model on the English multi-
turn dialogue dataset UltraChat, as detailed in Sec-
tion 4.1. The training process extended up to
300,000 steps, utilizing a peak learning rate of
5×10^{-4} with a warm-up phase of 3,000 steps.
A batch size of 12 was adopted. The entire train-
ing pipeline, including the initial single-turn model
training, was conducted on four NVIDIA A100
GPUs and required approximately three days to
complete.

F.3 Results

Due to the lack of multi-turn dialogue capabilities
in most existing SDMs, we only evaluate SLAM-
Omni and GLM-4-Voice (Zeng et al., 2024a), along
with Qwen2-0.5B-Instruct and Qwen2-7B-Instruct
(Yang et al., 2024a) as reference LLMs.

Table 11 presents the overall evaluation results
on the MtBenchEval dataset. The results demon-
strate that SLAM-Omni excels in acoustic quality
and speech-text alignment during multi-round con-
versations, achieving superior scores in both UT-
MOS and ASR-WER compared to GLM-4-Voice.
However, our model still exhibits a performance
gap in ChatGPT scores when compared to Qwen2-
0.5B-Instruct. This discrepancy is likely attributed
to differences in training data. Specifically, while
both SLAM-Omni and Qwen2-0.5B-Instruct were

fine-tuned on Qwen2-0.5B, our training utilized
only 400K single-turn dialogue samples and 300K
multi-turn dialogue samples, whereas Qwen2-0.5B-
Instruct leveraged large-scale text instruction data.

Models	LLM Scale	ChatGPT Score ↑	UTMOS ↑	ASR-WER ↓
Qwen2-7B-Instruct	7B	79.65	-	-
GLM-4-Voice	9B	68.35	4.22	7.99%
Qwen2-0.5B-Instruct	0.5B	59.12	-	-
SLAM-Omni (ours)	0.5B	32.88	4.45	7.61%

Table 11: Evaluation results on MtBenchEval dataset.

G Chinese Spoken Dialogue Evaluation

G.1 Datasets

Existing Spoken Dialogue Models (SDMs) and
Large Audio Language Models (LALMs) lack
a comprehensive multilingual evaluation frame-
work, as most existing benchmarks, including
VoiceBench (Chen et al., 2024b), MMAU (Sakshi
et al., 2024), and AIR-Bench (Yang et al., 2024b)
focus only on English. To broaden the scope of
model evaluation, we propose a detailed evaluation
benchmark to assess SDM's Chinese language ca-
pabilities. Similar to the English evaluation frame-
work introduced in Section 4.3, the Chinese bench-
mark evaluates the performance of SDMs across
three key dimensions. Specifically, six carefully cu-
rated datasets were proposed, targeting on SDMs'
proficiency in understanding, reasoning, and oral
conversation.

For understanding, in alignment with Section
4.3, we focus on the model's ability to repeat dia-
logue and summarize content in Chinese. We select
a broad spectrum of everyday conversation topics,
including greetings, work, hobbies, family, health,
and weather, to prompt the model to repeat the
conversation. To further evaluate the model's com-
prehension and summarization abilities, we also
draw samples from the Chinese short text summa-
rization dataset LCSTS (Hu et al., 2015), focusing
on samples that are suitable for oral expression.

For reasoning, we meticulously created the
MLC-zh dataset, which specifically targets Math,
Logic, and Commonsense reasoning within Chi-
nese dialogue contexts. In addition, we selected
appropriate samples from the Openbook-QA[16] (Mi-
haylov et al., 2018) test set that are relevant to con-
versational scenarios. The question and answer
pairs were translated into Chinese using GPT-4o

---

[16]https://huggingface.co/datasets/allenai/
openbookqa

2280

Models	LLM Scale	Understanding		Reasoning		Oral Conversation	
		Repeat-zh	LCSTS	MLC-zh	OpenbookQA-zh	AlpacaEval-zh	Claude-zh
Freeze-Omni	7B	3.66	70.33	32.43	10.89	59.40	67.76
GLM-4-Voice	9B	79.10	77.14	46.08	49.93	69.26	84.02
SLAM-Omni (ours)	0.5B	22.02	36.97	15.88	8.17	42.53	48.40

Table 12: ChatGPT scores of SDMs across three dimensions on Chinese evaluation dataset.

mini (OpenAI, 2024a), and their phrasing was modified to ensure better alignment with daily conversation.

Furthermore, to evaluate model's oral conversational abilities, we chose samples from AlpacaEval[17] (Li et al., 2023) and Claude-3-Opus-Instruct[18] (Li et al., 2023) that align with daily conversational contexts. Unlike its English counterpart, samples from the *oasst* and *koala* subsets of AlpacaEval were chosen to construct the AlpacaEval-zh subset. The detailed statistics of the Chinese evaluation dataset are provided in Table 13.

Types	Datasets	#Samples	Avg. #Words	Avg. Audio len
Understanding	Repeat-zh	210	30.74	7.94
	LCSTS	229	126.97	27.44
Reasoning	MLC-zh	149	21.99	6.06
	OpenbookQA-zh	257	86.95	19.07
Oral Conversation	AlpacaEval-zh	273	60.74	14.72
	Claude-zh	200	28.92	7.41

Table 13: The statistics of Chinese evaluation datasets.

Similar to the English evaluation dataset in Appendix D, all instructions in the datasets were synthesized into speech using the CosyVoice model, with timbres randomly sampled from the timbre library, following the methodology described in Section 4.1. Examples from the Chinese evaluation datasets are presented below.

Example of Repeat-zh dataset

Input:
"请跟我读：天行健，君子以自强不息。"

Reference:
"天行健，君子以自强不息。"

Example of LCSTS dataset

Input:
"你好！我这里有一段文本，请帮我总结一下它的内容。随着中国老龄化趋势严峻，养老问题受到越来越多人重视。有人担心，老后不仅不能老而富足、"优雅地老去"，反而因老致贫，陷入"银发贫困"。据悉，部分早退休领取最低养老金的人群或因无力购护理服务陷入"银发贫困状"。"

Suggested answer:
"报告：我国老龄化形势更严峻部分人或因老致贫。"

Example of OpenbookQA-zh dataset

Input:
我们知道：摩擦力是在两个物体表面接触时，抵消它们运动的力量。那么，飞机在飞行的时候，和什么没有摩擦呢？，请从以下选项中选择：
A. 机翼
B. 地面
C. 空气
D. 云朵

Suggested answers:
B.地面

Examples of MLC-zh dataset

Input1 (math):
"如果你有 3 个 5 元的硬币，5 个 2 元的硬币，那么你一共有多少钱？"
Input2 (logic):
"一只鸟飞进了一个房间，它飞到屋顶上停下。请问，这只鸟在哪个位置？"
Input3 (common sense):
"为什么苹果和胡萝卜不应该放在一起？"

Reference1 (math):
"你一共有 3 乘以 5 加上 5 乘以 2，等于 15 加 10，共 25 元。"

Reference2 (logic):
"题目中明确说这只鸟飞到屋顶上停下，所以它在屋顶上。"

Reference3 (common sense):
"因为苹果释放一种叫乙烯的气体，可能加速胡萝卜变质，所以最好分开存放。"

Example of AlpacaEval-zh dataset

Input:
"请问，法国有哪些地区适合中等强度的徒步旅行，不需要爬得太累呢？"

Example of Claude-zh dataset

Input:
"请描述一下一千八百七十一年巴黎公社起义的事件、重要人物和后果。"

G.2 Training Details

For training the Chinese voice interaction model, we utilized the Chinese multi-turn dataset Belle_train_3.5M_CN, as detailed in Section 4.1. The model configurations were consistent with those used in the main experiments and the multi-turn dialogue experiments. Specifically, we employed Qwen2-0.5B as the LLM backbone and Whisper-small as the speech encoder. The training process followed a single-stage fine-tuning strategy on the specified dataset, with a group size of $G = 3$. The total training steps were set to 300,000, with a peak learning rate of 5×10^{-4} and a warmup period of 3,000 steps. The batch size was configured to 64. The training process was conducted on 32 Tesla V100 GPUs and required approximately 30 hours to complete.

G.3 Results

The Chinese language capabilities of SLAM-Omni are evaluated on the aforementioned curated datasets. Due to the scarcity of multilingual spoken dialogue models, we used GLM-4-Voice (Zeng et al., 2024a) and Freeze-Omni (Wang et al., 2024), which are currently the only SDMs that support both Chinese input and output. It should be noted that these models feature larger LLM backbones

and are trained on more diverse data, resulting in their improved performance. We use *paraformer-zh* [19] (Gao et al., 2022) to transcribe audio outputs from SDMs into text, and then evaluate the corresponding ChatGPT score and the CER (%). The evaluation process and the scoring criteria are detailed in Appendix E.

Models	ChatGPT Score ↑	UTMOS ↑	ASR-CER ↓
Freeze-Omni	35.34	3.61	6.3%
GLM-4-Voice	67.59	3.09	4.5%
SLAM-Omni (ours)	25.12	3.67	4.4%

Table 14: Overall evaluation results for SDMs.

Table 12 and Table 14 present the evaluation results of SLAM-Omni on Chinese evaluation datasets. The evaluation results are largely consistent with the English benchmarks. SLAM-Omni excels in audio quality, achieving superior performance in CER and UTMOS metrics, reflecting its strong acoustic modeling ability. However, in reasoning and comprehension tasks, its ChatGPT score falls short compared to the larger SDM, highlighting the importance of the LLM backbone in SDM construction.

---

[19] https://huggingface.co/funasr/paraformer-zh