



OpenOmni: Advancing Open-Source Omnimodal Large Language Models with Progressive Multimodal Alignment and Real-time Emotional Speech Synthesis

Run Luo^{1,2,3*}, Ting-En Lin⁴, Haonan Zhang⁴, Yuchuan Wu⁴, Xiong Liu⁴,
Yongbin Li^{4†}, Longze Chen^{1,2}, Jiaming Li^{1,2}, Lei Zhang^{1,2}, Xiaobo Xia^{5,6},
Hamid Alinejad-Rokny⁷, Fei Huang⁴, Min Yang^{1,3†}

¹Shenzhen Key Laboratory for High Performance Data Mining,

Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences

²University of Chinese Academy of Sciences

³Shenzhen University of Advanced Technology ⁴Tongyi Lab, Alibaba Group

⁵National University of Singapore ⁶MoE Key Laboratory of Brain-inspired Intelligent
Perception and Cognition, University of Science and Technology of China

⁷University of New South Wales, Sydney, New South Wales, Australia

{r.luo, min.yang}@siat.ac.cn {ting-en.lte, shuide.lyb}@alibaba-inc.com

Abstract

Recent advancements in omnimodal learning have significantly improved understanding and generation across images, text, and speech, yet these developments remain predominantly confined to proprietary models. The lack of high-quality omnimodal datasets and the challenges of real-time emotional speech synthesis have notably hindered progress in open-source research. To address these limitations, we introduce OpenOmni, a two-stage training framework that integrates omnimodal alignment and speech generation to develop a state-of-the-art omnimodal large language model. In the alignment phase, a pretrained speech model undergoes further training on image-text tasks, enabling (near) zero-shot generalization from vision to speech, outperforming models trained on tri-modal datasets. In the speech generation phase, a lightweight decoder is trained on speech tasks with direct preference optimization, which enables real-time emotional speech synthesis with high fidelity. Extensive experiments demonstrate that OpenOmni surpasses state-of-the-art models across omnimodal, vision-language, and speech-language benchmarks. It achieves a 4-point absolute improvement on OmniBench over the leading open-source model VITA, despite using $5\times$ fewer training examples and a smaller model size (7B vs. 7×8 B). Besides, OpenOmni achieves real-time speech generation with less than 1 second latency at non-autoregressive mode, reducing inference time by $5\times$ compared to autoregressive methods, and improves emotion classification accuracy by 7.7%. The codebase is available at <https://github.com/RainBowLuoCS/OpenOmni>.

1 Introduction

The success of large language models (LLMs) [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] has driven rapid advancements in multimodal large language models (MLLMs) [11, 12, 13, 14, 15, 16, 17], partic-

*Work done while interning at Tongyi Lab.

†Min Yang and Yongbin Li are corresponding authors.

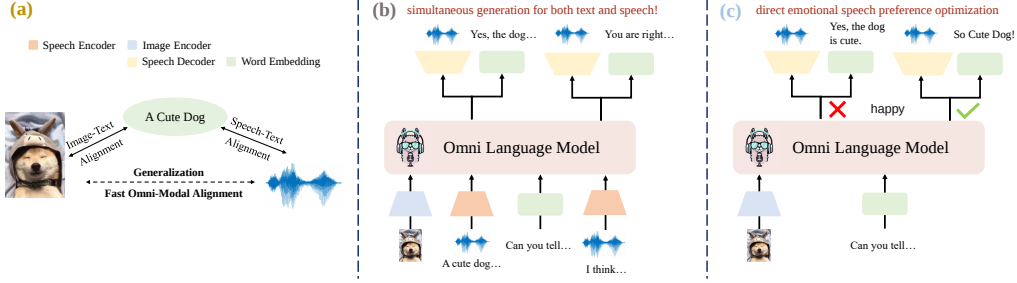


Figure 1: **Overview of the motivation and architecture of OpenOmni.** (a) OpenOmni adopts a progressive alignment strategy to generalize from vision-language to speech-language tasks, avoiding the need for costly tri-modal datasets and resources. (b) OpenOmni integrates a lightweight end-to-end speech decoder, enabling parallel text and speech generation while effectively reducing inference latency. (c) By utilizing DPO, OpenOmni generates emotionally coherent and context-aware speech without relying on additional control modules or handcrafted prompts. For simplicity, our core architecture is presented without the connectors between modules.

ularly in vision-language models (VLMs) [18, 19, 20, 21, 22, 23] and speech-language models (SLMs) [24, 25, 26, 27, 28]. These innovations mark a paradigm shift in machine understanding and human-computer interaction, fueling interest in omnimodal large language models (OLLMs), which are models that integrate vision, language, and speech into a unified system. The emergence of GPT-4o underscores the potential of holistic multimodal AI, yet open-source alternatives remain significantly behind.

Despite their promise, existing open-source OLLMs [29, 30, 31, 32] face three fundamental challenges, limiting their performance in real-world applications. First, training fully end-to-end OLLMs requires high-quality tri-modal datasets (images, speech, and text), which are scarce, expensive, and difficult to curate at scale. Most open-source models rely on true tri-modal corpora and ignore pairwise datasets (*e.g.*, image-text or speech-speech), resulting in suboptimal cross-modal alignment and weaker generalization. Without effective zero-shot alignment strategies, these models struggle to transfer learned representations across modalities, reducing their robustness in realistic multimodal tasks. Second, existing models predominantly rely on autoregressive (AR) architectures, which generate outputs sequentially, introducing high inference latency that hinders real-time multimodal interaction. Speech generation, in particular, is slow, as most models integrate external text-to-speech (TTS) modules [33], resulting in latency overhead and preventing end-to-end optimization. Achieving low-latency multimodal synthesis is essential for applications such as conversational AI, assistive technologies, and real-time interactive agents, where response time directly affects usability. Finally, emotionally expressive speech is critical for natural and engaging human-computer interactions. However, current OLLMs fail to generate emotionally consistent responses. Most models lack self-awareness, producing flat and robotic speech that does not modulate prosody, tone, or sentiment based on conversational context. Without direct preference optimization (DPO) [34, 35] for emotional speech, existing models struggle to align speech intonation with user emotions, leading to inauthentic and disconnected interactions. To summarize, these challenges significantly constrain the real-world applicability of open-source OLLMs, leaving commercial models far ahead in omnimodal reasoning, real-time interaction, and expressive speech synthesis.

To bridge this gap, we propose OpenOmni in this paper, which is a fully open-source two-stage training framework that enables efficient omnimodal learning while addressing the key limitations of existing models. As illustrated in Figure 1, OpenOmni introduces a progressive alignment strategy that enables cross-modal generalization from vision-language tasks to speech-language tasks, eliminating the need for expensive tri-modal datasets and computing resources. It further incorporates a lightweight and end-to-end speech decoder that facilitates parallel text and speech generation, which drastically reduces inference latency compared to autoregressive models. Moreover, by leveraging direct preference optimization (DPO), our model generates emotionally coherent and context-aware speech without requiring additional control modules or handcrafted prompts.

Extensive experiments confirm that OpenOmni achieves state-of-the-art performance in omnimodal alignment, real-time speech synthesis, and emotional speech generation. Compared to VITA [31], the leading fully open-source OLLM, which employs a $7\times 8B$ language model trained on 5M sam-

ples, OpenOmni attains superior results with a smaller model size (7B vs. 7×8 B) and $3\times$ fewer training samples (1.6M vs. 5M) while outperforming VITA by four absolute points on the OmniBench benchmark [36]. Additionally, OpenOmni reduces speech generation latency by $5\times$, achieving real-time inference (<1 second) and improving emotion classification accuracy by 7.7%. Our main contributions are summarized as follows:

- **High-quality speech datasets.** We construct O2S-300K and EO2S-9K, comprising 8,000 hours of bilingual text-synthesized speech, which enables efficient speech generation and emotional preference optimization.
- **Effective zero-shot omnimodal alignment.** We introduce a scalable and model-agnostic framework that enables low-resource and rapid omnimodal alignment using text as a pivot, followed by speech generation and emotional preference training. This approach allows the rapid development of an advanced all-modal assistant akin to GPT-4o.
- **An end-to-end omnimodal LLM.** We train OpenOmni with integrated text, image, and speech understanding progressively. After speech generation training and emotional preference optimization, OpenOmni can naturally produce real-time emotional speech.

2 Related Work

Vision-language models. The rapid advancement of vision-language models (VLMs) has been largely driven by the remarkable success of large language models (LLMs) [37, 38, 39, 40, 41] and the increasing availability of diverse image-text instruction data [18, 42, 43, 44, 45, 46, 47, 48]. LLaVA [18] and MiniGPT-4 [49] demonstrate strong cross-task generalization by integrating visual encoders with LLMs through lightweight connector modules trained on instruction datasets. To further enhance visual perception, LLaVA-NeXT [19] employs dynamic resolution techniques, which improve the adaptability to images of varying sizes and complexities. Expanding beyond conventional methods, DreamLLM [50] explores interleaved generation, enabling the simultaneous production of images and text within a shared multimodal context. Meanwhile, DEEM [21] enhances model robustness by employing diffusion models to extract visual features, which replaces traditional visual encoders and simplifies the overall architecture. These innovations collectively contribute to advancing vision-language reasoning in multimodal systems. Readers can refer to [51, 52, 53, 54, 55] for more details and recent advances in VLMs.

Speech-language models. Recent advancements in speech-language models (SLMs) [56, 57, 58, 59, 60] have significantly improved human-computer interactions by enabling direct speech processing without relying on intermediate text transcription. For example, SpeechGPT [11] and LLaMA-Omni [25] eliminate the need for explicit text-based transcriptions, reducing latency in multimodal content generation. For full-duplex dialogue systems, Moshi [61] and OmniFlatten [62] introduce mechanisms for handling simultaneous speech and text streams, adeptly managing challenges such as overlapping speech and interruptions [63]. Meanwhile, Freeze-Omni [64] introduces an innovative training approach that preserves the core capabilities of the original LLM, allowing low-latency speech-to-speech dialogue without requiring modifications to the pre-trained architecture. Focusing on emotional speech synthesis, Emo-DPO [65] applies direct preference optimization to generate expressive and controllable emotional speech, which addresses the emotional coherence gap in existing speech-language models. These developments mark a significant shift towards more natural and real-time speech interactions in multimodal AI systems. Readers can refer to [66] for more details of SLMs.

Omnimodal language models. With the development of multimodal research, models are increasingly shifting towards unified frameworks that seamlessly integrate diverse input and output modalities [67]. By tokenizing different data types into a shared representation, models like AnyGPT [29] and Unified-IO 2 [68] achieve seamless cross-modal task adaptability, allowing them to process audio, text, and images without significant architectural modifications. More recently, Mini-Omni2 [69] extends multimodal capabilities by integrating visual and auditory encoders, enabling real-time multimodal responses while incorporating mechanisms for detecting and interpreting semantic interruptions. Meanwhile, video-SALMONN [30] enhances video understanding by incorporating fine-grained temporal modeling, improving the model’s ability to interpret speech

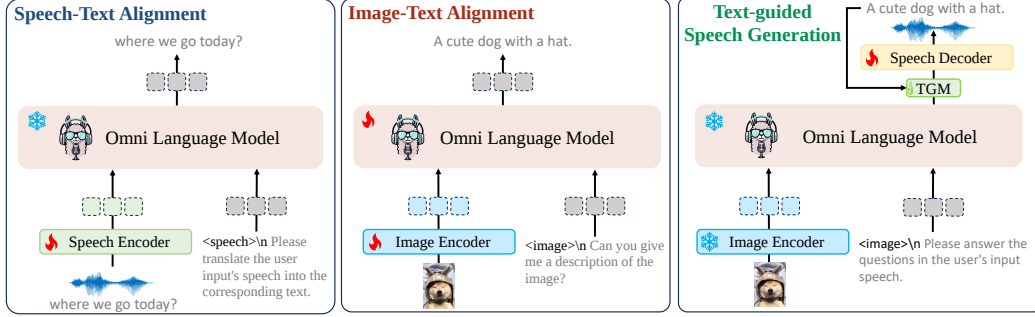


Figure 2: **Overview of the training process of OpenOmni.** To enable zero-shot omnimodal learning and real-time emotional speech generation, OpenOmni undergoes a progressive three-stage training process: (1) **Speech-text alignment.** A speech encoder extracts continuous speech and text features for alignment learning, equipping the large language model with speech understanding capabilities. (2) **Image-text alignment.** An image encoder extracts continuous image and text features, facilitating alignment learning that enhances OpenOmni’s image comprehension and instruction-following abilities. This process also establishes implicit omnimodal alignment, which enables omni-understanding. (3) **Text-guided speech generation.** A lightweight speech decoder is trained using high-quality synthesized speech dialogue data, with a focus on direct preference optimization for emotional speech. This final stage allows OpenOmni to generate real-time and self-aware emotional speech. A text-guided module (TGM) is utilized to accelerate the training convergence.

and actions within videos. To enhance human-computer interaction, VITA [31] introduces duplex communication schemes, enabling fluid and intuitive exchanges between users and AI models. EMOVA [32] further extends the expressive capabilities of multimodal systems by integrating controllable emotional speech synthesis, which provides more natural and engaging user interactions.

Building upon these advancements, OpenOmni introduces a novel method for nearly zero-shot omnimodal alignment across language, vision, and speech, which incorporates self-aware emotional speech synthesis to enhance expressiveness and realism. By optimizing for speed, data efficiency, and generalization, OpenOmni achieves state-of-the-art performance in omnimodal tasks, surpassing previous models in real-time speech generation, multimodal alignment, and emotion-aware synthesis. Note that compared to Qwen-Omni [70], OpenOmni is a fully open-source solution focused on achieving advanced OLLMs under limited training and data resources, which helps researchers to easily conduct their studies and accelerate innovation in the field.

3 Method

In this section, we first formulate the omnimodal learning problem and provide an overview of the training procedure of OpenOmni as demonstrated in Figure 2. Afterward, we describe the specific training procedures for omnimodal alignment and real-time speech generation step by step.

3.1 Problem Setup and OpenOmni Overview

Problem setup. Omnimodal learning aims to model the relationships between images ($\mathbf{x}^v$), speech ($\mathbf{x}^s$), and text ($\mathbf{x}^t$). The speech-to-text alignment task, which generates relevant text responses given input speech encoded by a speech encoder $h_s(\cdot)$, is formulated as learning $p_\phi(\mathbf{x}^t|h_s(\mathbf{x}^s))$, parameterized by ϕ . Similarly, the image-to-text alignment task, which involves generating textual descriptions for input images encoded by an image encoder $h_v(\cdot)$, is modeled as learning the conditional distribution $p_\theta(\mathbf{x}^t|h_v(\mathbf{x}^v))$, parameterized by θ . Lastly, the omnimodal-to-speech generation task, which synthesizes speech responses based on input text, speech, and images, is represented as learning $p_\psi(\mathbf{x}^s|f_{\text{LLM}}(\mathbf{x}^t, h_s(\mathbf{x}^s), h_v(\mathbf{x}^v)))$, parameterized by ψ , where f_{LLM} represents the large language model.

In the setting of standard omnimodal learning, training typically relies on image-speech-text triples $\mathcal{D}_o = \{(\mathbf{x}_i^v, \mathbf{x}_i^s, \mathbf{x}_i^t)\}_{i=1}^K$. Nevertheless, high-quality image-text-speech datasets are scarce. To mitigate this limitation, we introduce text as a pivot, which leverages a large-scale speech-text dataset $\mathcal{D}_{s-t} = \{(\mathbf{x}_i^s, \mathbf{x}_i^t)\}_{i=1}^M$ and image-text dataset $\mathcal{D}_{v-t} = \{(\mathbf{x}_i^v, \mathbf{x}_i^t)\}_{i=1}^N$, where $M \gg K$ and $N \gg K$. Inspired by human learning mechanisms, where individuals naturally align visual concepts with speech across languages, OpenOmni transfers visual concepts learned from image-text tasks to speech understanding. Technically, OpenOmni decomposes the omnimodal alignment process into two consecutive stages: *speech-text alignment* and *image-text alignment*. The speech-text alignment stage establishes cross-modal alignment between speech $\mathbf{x}^s$ and text $\mathbf{x}^t$. This is achieved by training a speech LLM on text-speech pairs $\mathcal{D}_{s-t}$ with the objective $p_\phi(\mathbf{x}^t|h_s(\mathbf{x}^s))$, which also ensures that the hidden representations of semantically similar speech-text pairs are close. In the image-text alignment stage, OpenOmni utilizes image-text pairs $\mathcal{D}_{v-t}$ to optimize the objective $p_\theta(\mathbf{x}^t|h_v(\mathbf{x}^v))$. Note that OpenOmni is architecture-agnostic, which allows flexible integration with existing advanced model architectures. Below, we detail OpenOmni.

3.2 Speech-Text Alignment

We incorporate a speech encoder h_s to extract audio features from input speech $\mathbf{x}^s$. These audio features $h_s(\mathbf{x}^s)$ are then replaced with corresponding text as input into the LLM. Following recent work to train speech conversation models [25, 24, 11], we pretrain OpenOmni on a large scale of text-speech pairs using the language modeling objective:

$$\mathcal{L}_{s-t}(\phi, \mathcal{D}_{s-t}) = - \sum_{i=1}^M \log p_\phi(\mathbf{x}_i^t|h_s(\mathbf{x}_i^s)). \quad (1)$$

3.3 Image-Text Alignment

We incorporate an image encoder h_v to provide visual features as $h_v(\mathbf{x}^v)$. These visual features are then concatenated with the text embedding as input into the speech LLM. Following prior work to train image-text conversation models [18, 42], OpenOmni’s training process for image-text alignment consists of two sub-stages: image-text pretraining and image-text instruction tuning.

Image-text pretraining. In this sub-stage, we pretrain the visual module to align it with the LLM on a large scale of image-text pairs using the language modeling objective:

$$\mathcal{L}_{v-t}(\theta, \mathcal{D}_{v-t}) = - \sum_{i=1}^N \log p_\theta(\mathbf{x}_i^t|h_v(\mathbf{x}_i^v)). \quad (2)$$

Here we fix the parameters of the LLM to prevent short texts in the image-text pairs from influencing the general capabilities.

Image-text instruction tuning. To enhance models’ capabilities in following human instructions, we conduct instruction tuning on elaborately curated multimodal instruction tuning datasets built by blending existing image-text instruction tuning datasets. We denote this image-text instruction tuning dataset as $\mathcal{D}_{v-t}^l = \{(\mathbf{x}_i^v, \mathbf{x}_i^{t,q}, \mathbf{x}_i^{t,a})\}_{i=1}^L$, where $\mathbf{x}_i^{t,q}$ denotes the instruction and $\mathbf{x}_i^{t,a}$ is the response. Both the visual module and the speech LLM are then fine-tuned by maximizing the probability of the response:

$$\mathcal{L}_{v-t}^l(\theta, \mathcal{D}_{v-t}^l) = - \sum_{i=1}^L \log p_\theta(\mathbf{x}_i^{t,a}|h_v(\mathbf{x}_i^v), f_{\text{LLM}}(\mathbf{x}_i^{t,q})). \quad (3)$$

Remark. We observe a *quasi-zero-shot* transfer capability in OpenOmni within this scenario. When instruction tuning is performed exclusively on the image-text dataset, the model demonstrates the ability to respond accurately to an image $\mathbf{x}^v$ and either a text-based question $\mathbf{x}^{t,q}$ or an instruction provided in speech $\mathbf{x}^{s,q}$. However, its responses are predominantly in text. This behavior can be attributed to the inherent similarity between the hidden representations of textual and spoken instructions learned by the LLM, *i.e.*, $f_{\text{LLM}}(\mathbf{x}^{t,q}) \approx f_{\text{LLM}}(h_s(\mathbf{x}^{s,q}))$. Consequently, the model satisfies the following approximation: $p_{\theta}(\mathbf{x}^{t,a}|h_v(\mathbf{x}^v), f_{\text{LLM}}(\mathbf{x}^{t,q})) \approx p_{\theta}(\mathbf{x}^{t,a}|h_v(\mathbf{x}^v), f_{\text{LLM}}(h_s(\mathbf{x}^{s,q})))$. OpenOmni completes the progressive omnimodal alignment, enabling the LLM to achieve a comprehensive understanding across image, text, and speech modalities.

3.4 Text-Guided Speech Generation

For speech generation, we incorporate a speech decoder h_s^{de} to generate speech based on the output of the LLM f_{LLM} . The speech generation training process in OpenOmni consists of two sub-stages: speech decoder training and emotional speech direct preference optimization (DPO).

Speech decoder training. To equip OpenOmni with real-time speech generation for enhancing interactive experiences, we adopt a streaming speech decoder, which supports both autoregressive (AR) and non-autoregressive (NAR) speech decoding modes. Besides, we curate a dataset, termed OpenOmni-300K, consisting of 300K single-round image-text instructions from MMEval [42] and UltraChat [71] with corresponding speech responses for speech decoder training. We denote this dataset as $\mathcal{D}_{\text{o-s}}^{\text{l}} = \{(\mathbf{x}_i^v, \mathbf{x}_i^{t,q}, \mathbf{x}_i^{t,a}, \mathbf{x}_i^{s,a})\}_{i=1}^L$, where $\mathbf{x}_i^{s,a}$ is the speech response.

To process the speech response $\mathbf{x}^{s,a}$, we follow [11, 25] to discretize speech into discrete units. Specifically, we use a pretrained speech tokenizer [11] to extract continuous speech representations and then convert these representations into a single unit, resulting in the final discrete unit sequence $\mathbf{x}^{u,a} = [x_1^{u,a}, x_2^{u,a}, \dots]$, where $x_i^{u,a} \in \{0, 1, \dots, V-1\}$ with V is the speech vocabulary size. The discrete units can then be converted back into a waveform using an additional unit-based vocoder [72], trained on English and Chinese datasets. As shown in Figure 3, we integrate the streaming speech decoder after the LLM to generate speech responses. The decoder consists of a *mixture of expert (MoE)* layer and a tiny standard decoder-only language model. The MoE layer stabilizes training and accelerates convergence—without this layer, the speech decoder fails to train effectively. Similar to [73, 74, 25], the speech decoder takes the output hidden states from the LLM as input and generates the discrete unit sequence corresponding to the speech response in real-time.

Given the output hidden states of the text response, denoted by $f_{\text{LLM}}(\mathbf{x}^v, \mathbf{x}^{s,q})$, we first pass these hidden states through the text-guided module (TGM) to obtain the transformed hidden state $\mathbf{c}$. Then, $\mathbf{c}$ is fed into the speech decoder layers, leading to the final hidden state sequence $\mathbf{o}$. We use *connectionist temporal classification (CTC)* [75] to align $\mathbf{o}$ with the discrete unit sequence $\mathbf{x}^{u,a}$. During training, CTC marginalizes over all possible alignments as follows:

$$\mathcal{L}_{\text{CTC}}(\psi, \mathcal{D}_{\text{o-s}}^{\text{l}}) = -\log p_{\psi}(\mathbf{x}^{u,a}|\mathbf{o}) = -\log \sum_{\mathbf{A} \in \Delta(\mathbf{x}^{u,a})} p_{\psi}(\mathbf{A}|\mathbf{o}) = -\log \sum_{\mathbf{A} \in \Delta(\mathbf{x}^{u,a})} \prod p_{\psi}(x_i^{u,a}|\mathbf{o}), \quad (4)$$

where $\Delta(\mathbf{x}^{u,a})$ denotes all possible alignments that collapse to $\mathbf{x}^{u,a}$. During inference, the best alignment is selected as $\mathbf{A}^* = \arg \max_{\mathbf{A}} p(\mathbf{A}|\mathbf{o})$. The corresponding discrete unit sequence is fed into the vocoder to synthesize the waveform.

Emotional speech DPO. To enable OpenOmni to generate self-aware, emotionally coherent, and expressive speech based on contextual history without additional control modules, we introduce

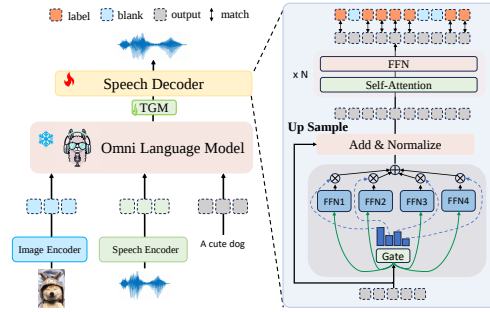


Figure 3: **The structure of our speech decoder.** The speech decoder consists of a mixture of expert modules and multiple transformer layers, which achieves end-to-end speech unit learning through the connectionist temporal classification (CTC) loss.

the *CTC-DPO* algorithm. This method enhances smooth and natural dialogue interactions and is formulated as

$$\mathcal{L}_{\text{CTC-DPO}} = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_1)} [\log \sigma(\beta \log \frac{\pi_*(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})} - \beta \log \frac{\pi_*(\mathbf{y}_1|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_1|\mathbf{x})})], \quad (5)$$

where β is a constant, σ is the sigmoid function, and $(\mathbf{y}_w, \mathbf{y}_1)$ is the preference pair. Besides, the reference model π_{ref} is the pretrained model from the real-time speech generation stage and remains fixed during DPO training. Only the policy model π_* is updated. Compared to traditional reinforcement learning with human feedback (RLHF) [76], the DPO paradigm is simpler, more efficient, and more stable for aligning OpenOmni with self-aware emotional speech generation.

Following the Plutchik model of emotions [77], we construct a multi-turn dialogue preference dataset incorporating nine distinct emotions. Each preference pair consists of an emotionally congruent speech response unit sequence $\mathbf{y}_w = \mathbf{x}_w^{u,a}$, which aligns with the conversational history, and an emotionally neutral sequence $\mathbf{y}_1 = \mathbf{x}_1^{u,a}$, which is inconsistent with the context. The policy model π_* during training is optimized as: $-\log \pi_*(\mathbf{y}|\mathbf{x}) = -\log \sum_{\mathbf{A} \in \Delta(\mathbf{x}^{u,a})} \prod p_\psi(\mathbf{x}_i^{u,a}|\mathbf{o})$. After training, OpenOmni is capable of generating real-time and emotionally expressive multi-turn dialogues.

4 Experiments

4.1 Implementation Details

In this subsection, we introduce data construction and the models used. More details about the data and optimization strategy of OpenOmni can be found in Appendix A and Appendix D respectively.

Omnimodal alignment data. During the speech-text alignment phase, in addition to WeNet-Speech [78], LibriSpeech [79], and AIShell-4 [80], we exploit portions of shorter responses from O2S-300K, totaling 8,000 hours of data, for bilingual speech-text alignment training. For image-text alignment, we train OpenOmni on the LLaVA-Pretrain-595K [18]. Besides, in the image-text instruction tuning stage, we fine-tune OpenOmni on the compact high-quality dataset MMEvol [42] for efficient optimization.

Speech decoder training data. To support real-time speech generation, we curate a dataset of 300K instructions from MMEvol [42] and UltraChat [71] that include long responses for training the speech decoder. Specifically, we decompose multi-turn dialogues into single-turn question-answer pairs, rank the responses based on their length, and select 100K question-answer pairs with relatively long responses. To support bilingual output in Chinese and English, we translate 50K question-answer pairs into their corresponding Chinese versions using GPT-4o-mini API, and then convert the answers into the corresponding speech using CosyVoice [33]. We employ the same method for text-conditioned speech synthesis on 200K randomly selected data from UltraChat. As a result, we obtain 8,000 hours of high-quality bilingual speech generation data, termed O2S-300K.

Emotional speech DPO data. Based on the Plutchik model of emotions [77], which categorizes emotions into eight distinct types, we curate a multi-turn speech preference dataset, EO2S-9K, for self-awareness emotion evaluation. In more detail, we randomly select 200K samples from MMEvol and employ Qwen2-72B [2] to categorize responses into nine predefined emotions per round. From this, we extract 1K bilingual dialogues labeled with emotion categories, reserving an additional 100 samples as an emotional test set for evaluating self-aware speech generation. Since certain emotions, such as anger and sadness, are underrepresented in the MMEvol dataset, we augment the dataset using the GPT-4o-mini API to ensure sufficient data for these categories. The final dataset maintains an equal representation of Chinese and English samples. To further enhance emotional preference training, we use CosyVoice [33] to generate unconditional speech as negative samples and emotion-conditioned speech as positive samples, constructing preference pairs for training direct preference optimization in emotional speech generation.

Models. We design the architecture following LLaVA series [18, 19], where the omnimodal large language model consists of four key components: an LLM (Qwen2.5-7B-Instruct [2]) for next token prediction, an image encoder (CLIP-ViT-L [81]) for extracting visual features, a speech encoder

Table 1: **Overall omni-understanding results on OmniBench.** In each case, the best result is indicated in **bold**, and the second-best result is marked with an underline.

Method	Action & Activity	Story Description	Plot Inference	Identification & Description	Contextual & Environmental	Identity & Relationship	Text & Symbols	Count & Quantity	Overall
AnyGPT (7B) [29]	5.98	8.70	7.59	4.74	5.67	12.50	8.00	20.00	7.01
Video-SALMONN (13B) [30]	28.69	25.65	24.47	23.22	29.08	21.83	52.00	26.63	26.53
UnifiedIO2-Large (1.1B) [68]	28.29	22.17	<u>32.49</u>	30.81	28.37	21.83	16.00	13.33	27.76
UnifiedIO2-XLarge (3.2B) [68]	30.28	26.52	30.38	31.75	28.37	18.75	<u>28.00</u>	26.63	29.16
UnifiedIO2-XXLarge (6.8B) [68]	27.49	23.04	28.69	25.59	26.95	12.50	12.00	46.67	25.92
Baichuan-Omni (7B) [84]	-	-	-	-	-	-	-	-	33.25
VITA (7×8B) [31]	<u>33.47</u>	<u>34.35</u>	27.00	<u>36.02</u>	<u>43.97</u>	<u>31.25</u>	24.00	6.67	33.45
VITA-1.5 (7B) [31]	-	-	-	-	-	-	-	-	<u>33.48</u>
OpenOmni (7B)	36.65	45.65	32.91	44.08	48.23	34.38	24.00	<u>33.33</u>	37.40

Table 2: **Comparison with state-of-the-art methods on visual-language benchmarks.** This includes an indication of audio input/output support. The best performance among fully open-source models is highlighted in **bold**.

Model	w/ Audio IO	PT	IT	MMStar	MMB	MMB ^{CN}	HallBench	MathVista ^M	MMMU ^V	AI2D	RWQA
Proprietary Models											
GPT-4o	✓	-	-	-	83.4	82.1	55.0	63.8	69.1	-	75.4
GPT-4o-mini	✓	-	-	-	-	-	46.1	52.4	60.0	-	67.1
Weight Open-Source											
MiniCPM-V2.5 (8B) [85]	✗	570M	9.1M	51.3	76.7	73.3	42.5	54.3	45.8	-	63.5
Qwen2-VL-Chat (7B) [20]	✗	1.4B	-	60.7	86.4	81.9	50.6	58.2	52.7	-	69.7
Baichuan-Omni (7B) [84]	✓	-	8M	-	76.2	74.9	47.8	51.9	47.3	-	62.6
EMOVA (8B) [32]	✓	7.4M	4.4M	-	82.8	-	-	61.1	-	82.8	64.3
Fully Open-Source											
Cambrain-1 (8B) [86]	✗	2.5M	7M	50.7	-	-	34.3	47.0	41.8	73.1	64.2
MMEvolution (7B) [42]	✗	0.6M	1.5M	51.6	74.6	74.3	42.9	52.4	45.1	74.7	63.9
VITA (7×8B) [31]	✓	-	5M	-	74.7	71.4	39.7	44.9	45.3	74.3	59.0
OpenOmni (7B)	✓	0.6M	1.7M	52.3	76.2	76.4	44.2	52.7	46.7	74.8	64.3

(Whisper-large-v3 [82]) for extracting audio features and a streaming speech decoder (Qwen2.5-0.5B-Instruct [2]) for generating vivid speech in real-time. Moreover, an image-text projector and a speech-text projector are adopted to align the image-text and speech-text modalities, respectively. The MoE module and the text-guided module are designed to align the omnimodal embedding and speech decoder efficiently and stably. For the autoregressive mode, we use the speech tokenizer from GLM4-Voice [83] with a vocabulary size of 16K, which leads to better speech quality. For non-autoregressive models, we use the CosVoice [33] speech tokenizer with a smaller vocabulary size of 6K, which facilitates faster convergence during CTC-based optimization. All experiments are conducted on 8×NVIDIA A100-80G GPUs.

4.2 Main Results and Discussions

Omni-language evaluation. OmniBench [36] is a pioneering benchmark designed to evaluate omnimodal large language models (OLLMs) by assessing their ability to integrate and interpret simultaneous inputs from images, audio, and text. This evaluation framework consists of 1,142 question-answer pairs categorized into tasks that focus on cognitive and reasoning abilities, which poses significant challenges in entity recognition, causal inference, and abstract concept comprehension. We compare our OpenOmni with other OLLMs on OmniBench, with results summarized in Table 1. Notably, our model achieves excellent zero-shot omnimodal alignment using only two training phases: speech-text alignment and image-text alignment. Compared to the fully open-source state-of-the-art OLLM, *e.g.*, VITA [31], which is trained on tri-modal data (image-speech-text triplets), OpenOmni achieves superior overall results on OmniBench (37.40 vs. 33.45) despite using significantly fewer training parameters (7B vs. 7×8B) and less image-text training data (1.6M vs. 5M). Furthermore, by leveraging text as a pivot, our method completes omnimodal alignment implicitly, which demonstrates enhanced scalability in scenarios with limited tri-modal data. In addition to OmniBench, we provide empirical results on AV-Odyssey Bench in Appendix B.1.

Vision-language evaluation. To assess the effectiveness of OpenOmni in aligning image-text modalities, we compare its performance against previous vision-language models (VLLMs) across eight representative benchmarks: MMBench-EN [87], MMBench-CN [87], MMStar [88], RealWorldQA [89], MMMU [90], MathVista [91], AI2D [92], and HallusionBench [93]. To en-

Table 3: **Comparison with state-of-the-art methods on speech-language benchmarks.** In each case, the best result among Omnimodal LLMs is indicated in **bold**.

Model	AIShell-2 (ZH-CER)				Librispeech (EN-WER)			
	Dev		Test		Test_clean		Test_other	
	S2T	T2S	S2T	T2S	S2T	T2S	S2T	T2S
Speech LLM								
SpeechT5 [95]	-	-	-	-	2.4	-	5.8	-
SALMONN [96]	-	-	-	-	2.1	-	4.9	-
Mini-Omni (7B) [69]	-	-	-	-	4.7	-	9.4	-
Freeze-Omni (7B) [64]	-	-	-	-	3.2	-	7.7	-
Qwen2-Audio (7B) [24]	3.1	-	3.3	-	2.0	-	4.5	-
Omnimodal LLM								
AnyGPT (13B) [29]	-	-	-	-	8.5	-	-	-
VITA (7×8B) [31]	-	-	-	-	8.1	-	18.4	-
EMOVA (7B) [32]	10.3	7.9	-	-	4.0	3.4	-	-
VITA 1.5 (7B) [31]	-	-	-	-	3.4	-	7.5	-
OpenOmni (7B)	6.8	7.3	6.9	13.1	3.1	2.6	4.1	5.6

Table 4: **Overall self-aware emotional speech generation results on the bilingual EO2S-9K test set.** In each case, the best result is indicated in **bold**.

Model	Lang	Angry & Disgusted	Fearful	Happy	Neutral	Other	Sad	Surprised	Overall
OpenOmni w/ DPO	ZH	89.7	54.8	33.3	92.3	51.6	60.2	23.7	57.9
	ZH	96.6	78.4	37.7	97.1	62.8	90.7	29.8	70.4
OpenOmni w/ DPO	EN	89.2	68.7	57.5	91.9	48.0	75.6	7.5	62.6
	EN	91.3	70.4	60.6	94.6	49.6	77.3	13.9	65.4

sure reproducibility and maintain consistency across all models and benchmarks, we employ VLMEvalKit [94] for zero-shot evaluation. As shown in Table 2, OpenOmni achieves superior results compared to the fully open-source state-of-the-art OLLM, VITA [31], despite being trained on significantly less data. Notably, our model outperforms VITA with gains of 7.0% on MMBench-Chinese and 11.3% on HallusionBench. We can also observe that the use of additional speech modals can further enhance the vision-language capabilities of the model. Furthermore, compared to other fully open-source VLMs, OpenOmni maintains competitive performance despite reduced training data, which demonstrates the effectiveness of our image-text alignment strategy.

Speech-language evaluation. To evaluate the speech understanding and generation capabilities of our OpenOmni, we measure word error rate (WER) on the AIShell-2 [80] and Librispeech [79] benchmarks for two tasks: speech-to-text recognition (S2T) and text-to-speech generation (T2S). Specifically, for T2S evaluation, we use Whisper-large-V3 to transcribe OpenOmni’s synthesized speech and compute WER against ground-truth text labels. As shown in Table 3, OpenOmni achieves the best performance on both S2T and T2S tasks for bilingual (Chinese and English) data, and outperforms other omnimodal models. These results indicate that OpenOmni not only comprehends speech effectively but also generates fluent and high-quality audio while maintaining strong alignment between speech and text modalities. Additionally, compared to VITA [31], which relies on separate text-to-speech (TTS) models, and EMOVA [32], which uses an autoregressive (AR) structure, OpenOmni demonstrates significantly faster speech generation via two-mode support. Owing to its end-to-end, lightweight, and non-autoregressive (NAR) decoding mode, OpenOmni can generate up to 30 seconds of speech with less than one second of latency, which achieves real-time speech generation at over $5\times$ speed of autoregressive models.

Emotional speech synthesis evaluation. To assess the effectiveness of direct preference optimization in emotional speech generation, we evaluate OpenOmni’s self-aware emotional speech synthesis on the EO2S-9K test set. Specifically, we use Emotion2Vec [97] to classify the emotions in the generated speech and measure accuracy against ground-truth labels. As shown in Table 4, direct preference optimization for emotional speech effectively enhances OpenOmni’s ability to generate emotionally expressive speech. This improvement is particularly evident in bilingual and multi-

turn emotional speech generation tasks, demonstrating the model’s ability to produce natural and contextually aware speech with accurate emotional intonation.

We also provide ablation studies to investigate the text-guided module (TGM), the number of layers and experts in the speech decoder, training strategy in alignment stages, and the performance of order in progressive alignment. Due to the limited page, experimental results and the following discussions can be checked in Appendix C.

5 Conclusion

In this paper, we introduce OpenOmni, a novel omnimodal model that leverages text as a pivot to achieve tri-modal zero-shot alignment, which addresses the challenge of limited tri-modal data. By integrating a lightweight streaming speech decoder with direct preference optimization for emotional speech, OpenOmni enables real-time, self-aware, and high-quality speech interactions. The extensive evaluations demonstrate that OpenOmni achieves state-of-the-art performance on multiple benchmarks while using significantly fewer training parameters and less training data than previous advanced models. Comprehensive ablation studies and discussions are also presented to rigorously validate our claims.

Acknowledgements

This work was supported by Alibaba Research Intern Program. Min Yang is supported by Guangdong Basic and Applied Basic Research Foundation (2025B1515020032 and 2024A1515030166). Xiaobo Xia is supported by MoE Key Laboratory of Brain-inspired Intelligent Perception and Cognition, University of Science and Technology of China (Grant No. 2421002).

References

- [1] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- [2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.
- [3] Zhengwei Tao, Ting-En Lin, Xiancai Chen, Hangyu Li, Yuchuan Wu, Yongbin Li, Zhi Jin, Fei Huang, Dacheng Tao, and Jingren Zhou. A survey on self-evolution of large language models. *arXiv preprint arXiv:2404.14387*, 2024.
- [4] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- [5] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.
- [6] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023.
- [7] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38, 2024.
- [8] Lei Zhang, Yunshui Li, Jiaming Li, Xiaobo Xia, Jiayi Yang, Run Luo, Minzheng Wang, Longze Chen, Junhao Liu, Qiang Qu, et al. Hierarchical context pruning: Optimizing real-world code completion with repository-level pretrained code llms. In *AAAI*, pages 25886–25894, 2025.

- [9] Yunshui Li, Binyuan Hui, Xiaobo Xia, Jiayi Yang, Min Yang, Lei Zhang, Shuzheng Si, Ling-Hao Chen, Junhao Liu, Tongliang Liu, et al. One-shot learning as instruction data prospector for large language models. *arXiv preprint arXiv:2312.10302*, 2023.
- [10] Xiaohao Liu, Xiaobo Xia, Weixiang Zhao, Manyi Zhang, Xianzhi Yu, Xiu Su, Shuo Yang, See-Kiong Ng, and Tat-Seng Chua. L-mtp: Leap multi-token prediction beyond adjacent context for large language models. In *NeurIPS*, 2025.
- [11] Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. In *EMNLP Findings*, pages 15757–15773, 2023.
- [12] Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Philip S Yu. Multimodal large language models: A survey. In *BigData*, pages 2247–2256. IEEE, 2023.
- [13] Chaoyou Fu, Yuhao Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.
- [14] Xiaohui Chen, Satya Narayan Shukla, Mahmoud Azab, Aashu Singh, Qifan Wang, David Yang, ShengYun Peng, Hanchao Yu, Shen Yan, Xuewen Zhang, et al. Compcap: Improving multimodal large language models with composite captions. *arXiv preprint arXiv:2412.05243*, 2024.
- [15] Qing Jiang, Gen Luo, Yuqin Yang, Yuda Xiong, Yihao Chen, Zhaoyang Zeng, Tianhe Ren, and Lei Zhang. Chatrex: Taming multimodal llm for joint perception and understanding. *arXiv preprint arXiv:2411.18363*, 2024.
- [16] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024.
- [17] Jialong Zuo, Jiahao Hong, Feng Zhang, Changqian Yu, Hanyu Zhou, Changxin Gao, Nong Sang, and Jingdong Wang. Plip: Language-image pre-training for person representation learning. In *NeurIPS*, pages 45666–45702, 2024.
- [18] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2024.
- [19] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024.
- [20] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023.
- [21] Run Luo, Yunshui Li, Longze Chen, Wanwei He, Ting-En Lin, Ziqiang Liu, Lei Zhang, Zikai Song, Xiaobo Xia, Tongliang Liu, et al. Deem: Diffusion models serve as the eyes of large language models for image perception. *arXiv preprint arXiv:2405.15232*, 2024.
- [22] Yuchen Zhou, Jiayu Tang, Xiaoyan Xiao, Yueyao Lin, Linkai Liu, Zipeng Guo, Hao Fei, Xiaobo Xia, and Chao Gou. Where, what, why: Towards explainable driver attention prediction. In *ICCV*, 2025.
- [23] Yuchen Zhou, Jiayu Tang, Shuo Yang, Xiaoyan Xiao, Yuqin Dai, Wenhao Yang, Chao Gou, Xiaobo Xia, and Tat-Seng Chua. Logic unseen: Revealing the logical blindspots of vision-language models. *arXiv preprint arXiv:2508.11317*, 2025.
- [24] Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023.

- [25] Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. Llama-omni: Seamless speech interaction with large language models. *arXiv preprint arXiv:2409.06666*, 2024.
- [26] Ke-Han Lu, Zhehuai Chen, Szu-Wei Fu, Chao-Han Huck Yang, Jagadeesh Balam, Boris Ginsburg, Yu-Chiang Frank Wang, and Hung-yi Lee. Developing instruction-following speech language model without speech instruction-tuning data. In *ICASSP*, pages 1–5, 2025.
- [27] Shengpeng Ji, Ziyue Jiang, Hanting Wang, Jialong Zuo, and Zhou Zhao. Mobilespeech: A fast and high-fidelity framework for mobile zero-shot text-to-speech. In *ACL*, 2024.
- [28] Haonan Zhang, Run Luo, Xiong Liu, Yuchuan Wu, Ting-En Lin, Pengpeng Zeng, Qiang Qu, Feiteng Fang, Min Yang, Lianli Gao, Jingkuan Song, Fei Huang, and Yongbin Li. OmniCharacter: Towards immersive role-playing agents with seamless speech-language personality interaction. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26318–26331, Vienna, Austria, July 2025. Association for Computational Linguistics.
- [29] Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, Hang Yan, Jie Fu, Tao Gui, Tianxiang Sun, Yu-Gang Jiang, and Xipeng Qiu. AnyGPT: Unified multimodal LLM with discrete sequence modeling. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *ACL*, pages 9637–9662, 2024.
- [30] Guangzhi Sun, Wenyi Yu, Changli Tang, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, Yuxuan Wang, and Chao Zhang. video-salmonn: Speech-enhanced audio-visual large language models. *arXiv preprint arXiv:2406.15704*, 2024.
- [31] Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, Yifan Zhang, Xiong Wang, Di Yin, Long Ma, Xiawu Zheng, et al. Vita: Towards open-source interactive omni multimodal llm. *arXiv preprint arXiv:2408.05211*, 2024.
- [32] Kai Chen, Yunhao Gou, Runhui Huang, Zhili Liu, Daxin Tan, Jing Xu, Chunwei Wang, Yi Zhu, Yihan Zeng, Kuo Yang, et al. Emova: Empowering language models to see, hear and speak with vivid emotions. *arXiv preprint arXiv:2409.18042*, 2024.
- [33] Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. corr, abs/2407.05407, 2024. doi: 10.48550. *arXiv preprint ARXIV.2407.05407*.
- [34] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2024.
- [35] Shunyu Liu, Wenkai Fang, Zetian Hu, Junjie Zhang, Yang Zhou, Kongcheng Zhang, Rongcheng Tu, Ting-En Lin, Fei Huang, Mingli Song, et al. A survey of direct preference optimization. *arXiv preprint arXiv:2503.11701*, 2025.
- [36] Yizhi Li, Ge Zhang, Yinghao Ma, Ruibin Yuan, Kang Zhu, Hangyu Guo, Yiming Liang, Jiaheng Liu, Jian Yang, Siwei Wu, et al. Omnibench: Towards the future of universal omni-language models. *arXiv preprint arXiv:2409.15272*, 2024.
- [37] Muhammad Usman Hadi, Rizwan Qureshi, Abbas Shah, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, et al. A survey on large language models: Applications, challenges, limitations, and practical usage. *Authorea Preprints*, 3, 2023.
- [38] Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenying Huang, Lifeng Shang, Xin Jiang, and Qun Liu. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*, 2023.

- [39] Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, et al. A survey on large language models for recommendation. In *WWW*, page 60, 2024.
- [40] Shaokun Zhang, Xiaobo Xia, Zhaoqing Wang, Ling-Hao Chen, Jiale Liu, Qingyun Wu, and Tongliang Liu. Ideal: Influence-driven selective annotations empower in-context learners in large language models. In *ICLR*, 2024.
- [41] Longze Chen, Ziqiang Liu, Wanwei He, Yunshui Li, Run Luo, and Min Yang. Long context is not long at all: A prospector of long-dependency data for large language models. *arXiv preprint arXiv:2405.17915*, 2024.
- [42] Run Luo, Haonan Zhang, Longze Chen, Ting-En Lin, Xiong Liu, Yuchuan Wu, Min Yang, Minzheng Wang, Pengpeng Zeng, Lianli Gao, et al. Mmevol: Empowering multimodal large language models with evol-instruct. *arXiv preprint arXiv:2409.05840*, 2024.
- [43] Jinyi Hu, Yuan Yao, Chongyi Wang, Shan Wang, Yinxu Pan, Qianyu Chen, Tianyu Yu, Hanghao Wu, Yue Zhao, Haoye Zhang, et al. Large multilingual models pivot zero-shot multimodal learning across languages. *arXiv preprint arXiv:2308.12038*, 2023.
- [44] Yiwei Zhou, Xiaobo Xia, Zhiwei Lin, Bo Han, and Tongliang Liu. Few-shot adversarial prompt learning on vision-language models. In *NeurIPS*, pages 3122–3156, 2024.
- [45] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, pages 12888–12900, 2022.
- [46] Run Luo, Renke Shan, Longze Chen, Ziqiang Liu, Lu Wang, Min Yang, and Xiaobo Xia. Vcm: Vision concept modeling based on implicit contrastive learning with vision-language instruction fine-tuning. *arXiv preprint arXiv:2504.19627*, 2025.
- [47] Xiaobo Xia and Run Luo. Gui-r1: A generalist r1-style vision-language action model for gui agents. *arXiv preprint arXiv:2504.10458*, 2025.
- [48] Jiayi Ji, Haowei Wang, Changli Wu, Yiwei Ma, Xiaoshuai Sun, and Rongrong Ji. Jm3d & jm3d-llm: Elevating 3d representation with joint multi-modal cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [49] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.
- [50] Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, et al. Dreamllm: Synergistic multimodal comprehension and creation. *arXiv preprint arXiv:2309.11499*, 2023.
- [51] Jingyi Zhang, Jiaying Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [52] Zongxia Li, Xiyang Wu, Hongyang Du, Huy Nghiem, and Guangyao Shi. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189*, 2025.
- [53] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*, 2022.
- [54] Xingcheng Zhou, Mingyu Liu, Ekim Yurtsever, Bare Luka Zagar, Walter Zimmer, Hu Cao, and Alois C Knoll. Vision language models in autonomous driving: A survey and outlook. *IEEE Transactions on Intelligent Vehicles*, 2024.
- [55] Hanchao Liu, Wenyan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*, 2024.

- [56] Sijing Chen, Yuan Feng, Laipeng He, Tianwei He, Wendi He, Yanni Hu, Bin Lin, Yiting Lin, Yu Pan, Pengfei Tan, et al. Takin: A cohort of superior quality zero-shot speech generation models. *arXiv preprint arXiv:2409.12139*, 2024.
- [57] Helin Wang, Meng Yu, Jiarui Hai, Chen Chen, Yuchen Hu, Rilin Chen, Najim Dehak, and Dong Yu. Ssr-speech: Towards stable, safe and robust zero-shot text-based speech editing and synthesis. In *ICASSP*, pages 1–5, 2025.
- [58] Yuancheng Wang, Haoyue Zhan, Liwei Liu, Ruihong Zeng, Haotian Guo, Jiachen Zheng, Qiang Zhang, Xueyao Zhang, Shunsi Zhang, and Zhizheng Wu. Maskgct: Zero-shot text-to-speech with masked generative codec transformer. *arXiv preprint arXiv:2409.00750*, 2024.
- [59] Shujie Hu, Long Zhou, Shujie Liu, Sanyuan Chen, Lingwei Meng, Hongkun Hao, Jing Pan, Xunying Liu, Jinyu Li, Sunit Sivasankaran, et al. Wavllm: Towards robust and adaptive speech large language model. *arXiv preprint arXiv:2404.00656*, 2024.
- [60] Zhisheng Zheng, Puyuan Peng, Ziyang Ma, Xie Chen, Eunsol Choi, and David Harwath. Bat: Learning to reason about spatial sounds with large language models. In *ICML*, 2024.
- [61] Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue.
- [62] Qinglin Zhang, Luyao Cheng, Chong Deng, Qian Chen, Wen Wang, Siqi Zheng, Jiaqing Liu, Hai Yu, and Chaohong Tan. Omniflatten: An end-to-end gpt model for seamless voice conversation. *arXiv preprint arXiv:2410.17799*, 2024.
- [63] Ting-En Lin, Yuchuan Wu, Fei Huang, Luo Si, Jian Sun, and Yongbin Li. Duplex conversation: Towards human-like interaction in spoken dialogue systems. In *ACM SIGKDD*, pages 3299–3308, 2022.
- [64] Xiong Wang, Yangze Li, Chaoyou Fu, Lei Xie, Ke Li, Xing Sun, and Long Ma. Freeze-omni: A smart and low latency speech-to-speech dialogue model with frozen llm. *arXiv preprint arXiv:2411.00774*, 2024.
- [65] Xiaoxue Gao, Chen Zhang, Yiming Chen, Huayun Zhang, and Nancy F Chen. Emo-dpo: Controllable emotional speech synthesis through direct preference optimization. *arXiv preprint arXiv:2409.10157*, 2024.
- [66] Wenqian Cui, Dianshi Yu, Xiaoqi Jiao, Ziqiao Meng, Guangyan Zhang, Qichao Wang, Yiwen Guo, and Irwin King. Recent advances in speech language models: A survey. *arXiv preprint arXiv:2410.03751*, 2024.
- [67] Run Luo, Xiaobo Xia, Lu Wang, Longze Chen, Renke Shan, Jing Luo, Min Yang, and Tat-Seng Chua. Next-omni: Towards any-to-any omnimodal foundation models with discrete flow matching. *arXiv preprint arXiv:2510.13721*, 2025.
- [68] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savva Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action. In *CVPR*, pages 26439–26455, 2024.
- [69] Zhifei Xie and Changqiao Wu. Mini-omni2: Towards open-source gpt-4o model with vision, speech and duplex. *arXiv preprint arXiv:2410.11190*, 2024.
- [70] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, et al. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025.
- [71] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*, 2023.
- [72] Adam Polyak, Yossi Adi, Jade Copet, Eugene Kharitonov, Kushal Lakhotia, Wei-Ning Hsu, Abdelrahman Mohamed, and Emmanuel Dupoux. Speech resynthesis from discrete disentangled self-supervised representations. *arXiv preprint arXiv:2104.00355*, 2021.

- [73] Zhengrui Ma, Qingkai Fang, Shaolei Zhang, Shoutao Guo, Yang Feng, and Min Zhang. A non-autoregressive generation framework for end-to-end simultaneous speech-to-any translation. *arXiv preprint arXiv:2406.06937*, 2024.
- [74] Shaolei Zhang, Qingkai Fang, Shoutao Guo, Zhengrui Ma, Min Zhang, and Yang Feng. Streamspeech: Simultaneous speech-to-speech translation with multi-task learning. *arXiv preprint arXiv:2406.03049*, 2024.
- [75] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, pages 369–376, 2006.
- [76] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, pages 27730–27744, 2022.
- [77] 6seconds.org. Plutchik-model-emotion. <https://www.6seconds.org/2022/03/13/plutchik-wheel-emotions>, 2022.
- [78] Binbin Zhang, Hang Lv, Pengcheng Guo, Qijie Shao, Chao Yang, Lei Xie, Xin Xu, Hui Bu, Xiaoyu Chen, Chenchen Zeng, et al. Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition. In *ICASSP*, pages 6182–6186, 2022.
- [79] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *ICASSP*, pages 5206–5210, 2015.
- [80] Yihui Fu, Luyao Cheng, Shubo Lv, Yukai Jv, Yuxiang Kong, Zhuo Chen, Yanxin Hu, Lei Xie, Jian Wu, Hui Bu, et al. Aishell-4: An open source dataset for speech enhancement, separation, recognition and speaker diarization in conference scenario. *arXiv preprint arXiv:2104.03603*, 2021.
- [81] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021.
- [82] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, pages 28492–28518, 2023.
- [83] Aohan Zeng, Zhengxiao Du, Mingdao Liu, Kedong Wang, Shengmin Jiang, Lei Zhao, Yuxiao Dong, and Jie Tang. Glm-4-voice: Towards intelligent and human-like end-to-end spoken chatbot. *arXiv preprint arXiv:2412.02612*, 2024.
- [84] Yadong Li, Haoze Sun, Mingan Lin, Tianpeng Li, Guosheng Dong, Tao Zhang, Bowen Ding, Wei Song, Zhenglin Cheng, Yuqi Huo, et al. Baichuan-omni technical report. *arXiv preprint arXiv:2410.08565*, 2024.
- [85] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.
- [86] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024.
- [87] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023.
- [88] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024.
- [89] x.ai. Grok-1.5 vision preview. <https://x.ai/blog/grok-1.5v>, 2024.

- [90] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multi-modal understanding and reasoning benchmark for expert agi. In *CVPR*, pages 9556–9567, 2024.
- [91] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating math reasoning in visual contexts with gpt-4v, bard, and other large multimodal models. *arXiv e-prints*, pages arXiv–2310, 2023.
- [92] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *ECCV*, pages 235–251, 2016.
- [93] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusion-bench: An advanced diagnostic suite for entangled language hallucination & visual illusion in large vision-language models, 2023.
- [94] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *ACM MM*, pages 11198–11201, 2024.
- [95] Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, et al. Specht5: Unified-modal encoder-decoder pre-training for spoken language processing. *arXiv preprint arXiv:2110.07205*, 2021.
- [96] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*, 2023.
- [97] Ziyang Ma, Zhisheng Zheng, Jiaxin Ye, Jinchao Li, Zhifu Gao, Shiliang Zhang, and Xie Chen. emotion2vec: Self-supervised pre-training for speech emotion representation. *arXiv preprint arXiv:2312.15185*, 2023.
- [98] Ziqiang Liu, Feiteng Fang, Xi Feng, Xinrun Du, Chenhao Zhang, Zekun Wang, Yuelin Bai, Qixuan Zhao, Liyang Fan, Chengguang Gan, Hongquan Lin, Jiaming Li, Yuansheng Ni, Haihong Wu, Yaswanth Narsupalli, Zhigang Zheng, Chengming Li, Xiping Hu, Ruifeng Xu, Xiaojun Chen, Min Yang, Jiaheng Liu, Ruibo Liu, Wenhao Huang, Ge Zhang, and Shiwen Ni. li-bench: An image implication understanding benchmark for multimodal large language models, 2024.
- [99] Kaixiong Gong, Kaituo Feng, Bohao Li, Yibing Wang, Mofan Cheng, Shijia Yang, Jiaming Han, Benyou Wang, Yutong Bai, Zhuoran Yang, et al. Av-odyssey bench: Can your multi-modal llms really understand audio-visual information? *arXiv preprint arXiv:2412.02611*, 2024.
- [100] Jiaming Han, Kaixiong Gong, Yiyuan Zhang, Jiaqi Wang, Kaipeng Zhang, Dahua Lin, Yu Qiao, Peng Gao, and Xiangyu Yue. Onellm: One framework to align all modalities with language. In *CVPR*, pages 26584–26595, 2024.
- [101] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*, 2023.
- [102] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023.
- [103] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024.
- [104] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. In *ICML*, 2024.

A Data Construction

We provide details of the data construction for multiple training stages below:

- **OpenOmni-1-1:** In addition to datasets WeNetSpeech, LibriSpeech, and AIShell-4, we randomly select 80K image-text instruction data with shorter responses from MMEvol [42]. We translate 40K of this data into Chinese using Qwen72B and synthesize the responses into speech data with CosVoice. This results in 1,600 hours of OpenOmni-1-1 data for speech-text alignment pretraining.
- **OpenOmni-2-1:** For rapid image-text alignment pretraining, we use the llava pretrain dataset, following previous work [18, 19, 42, 98].
- **OpenOmni-2-2:** To achieve efficient image-text instruction tuning, we employ MMEvol data. Since we later train the speech decoder by freezing the LLM mode, we include O2S-300K to stabilize the training of the speech decoder, leading to a combined dataset of 1.7M for OpenOmni-2-2.
- **OpenOmni-3-1:** To better utilize computational resources, we select 300K data with long response instructions from MMEvol and UltraChat. This includes 100K image-text instruction data, 100K single-round dialogue, and 100K multi-round dialogue. We synthesize the corresponding speech using CosVoice, resulting in 8,000 hours of O2S-300K.
- **OpenOmni-3-2:** We curate 9K emotion preference data and generate emotional speech preference pairs using CosVoice’s conditional control. This is used for emotional speech direct preference optimization.

B Additional Experiments

B.1 Additional Omni-Language Evaluation

In addition to OmniBench, we conduct experiments on the AV-Odyssey Bench [99], which involves the four modalities: audio, text, image, and video. For video, we test by averaging 8 sampled frames into a single image. The experimental results are shown in Table 5 below. Compared to other OLLMs, OpenOmni achieves the best average performance using only bi-modal speech-text and image-text data. With 7B model parameters and no audio or video training, it outperforms VITA by 4.4 points, demonstrating the effectiveness and efficiency of OpenOmni.

Table 5: **Overall omni-understanding results on AV-Odyssey Bench.** In each case, the best result is indicated in **bold**. We conduct a performance comparison of omni-understanding among various fully open-source Omnimodal Large Language Models (OLLMs) on AV-Odyssey Bench. Compared to the state-of-the-art OLLM, VITA [31], which was trained on tri-modal data, OpenOmni achieves comparable advanced performance using significantly less training data and smaller model size.

Method	Timbre	Tone	Melody	Space	Time	Hall	Intricacy	Overall
OneLLM (7B) [100]	25.0	25.5	21.5	37.5	29.3	25.5	38.4	27.4
PandaGPT (7B) [101]	23.5	23.2	27.6	45.0	23.8	28.0	23.9	26.7
Video-LLaMA (7B) [102]	25.5	22.3	24.4	30.0	26.2	25.0	30.7	26.1
Video-LLaMA2(7B) [103]	24.1	25.5	26.4	30.0	27.2	33.0	34.5	26.8
AnyGPT (7B) [29]	24.6	25.0	26.4	27.5	29.2	29.0	25.7	26.1
NexTGPT (7B) [104]	23.3	20.9	27.8	30.0	28.8	28.5	23.6	25.5
VITA (7×8B) [31]	24.1	26.4	27.8	22.5	26.3	31.0	36.8	26.4
OpenOmni (7B)	23.9	27.7	25.9	60.0	25.2	29.5	37.6	32.8

C Additional Ablation Studies

On TGM. To explore the effect of TGM on speech generation in two modes, we plot the change of training loss under the same setting. As shown in Figure 4, we can observe that TGM can significantly improve the convergence speed of training and the performance of model speech generation,

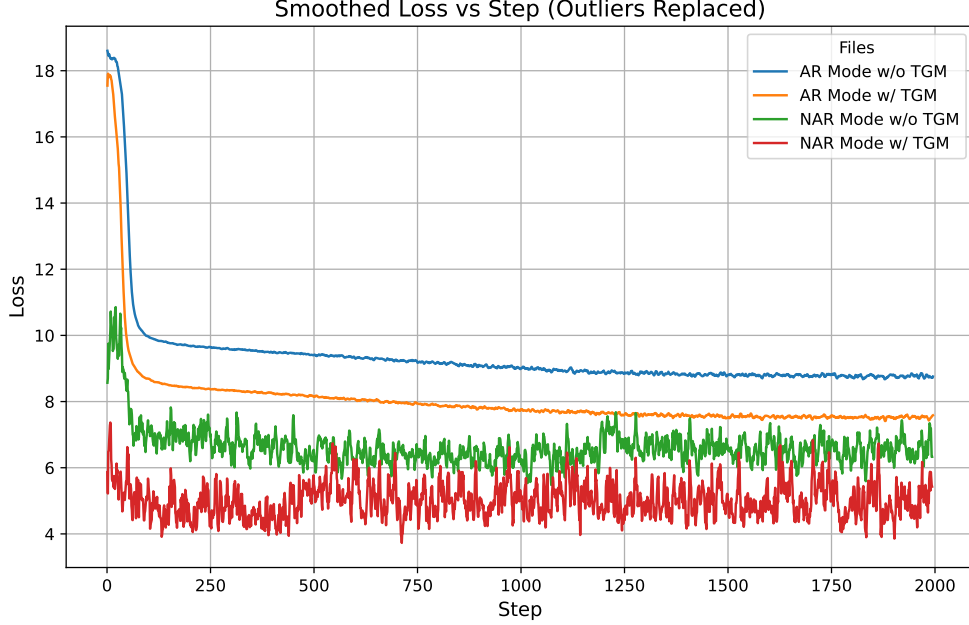


Figure 4: **Ablation study of the text-guided module (TGM).** In order to explore the effect of TGM on speech generation under the two modes, we plot the change of training loss under the same setting. TGM can significantly improve the convergence speed of training and improve the effect of speech generation of the speech decoder.

Table 6: **Ablation study on the number of layers and experts in the speech decoder.** Increasing experts in the mixture of experts module stabilizes the CTC loss during training and enhances speech generation capacity. Deeper transformer layers improve English and Chinese speech generation, with greater benefits for Chinese.

Layers	Experts	Wenetspeech(ZH)		Librispeech(EN)	
		Test_Net	Test_Meeting	Test_clean	Test_other
2	1	113.6	129.7	87.8	96.5
2	2	16.7	22.3	10.7	14.6
2	4	8.5	8.4	4.2	4.7
4	4	7.3	7.9	3.8	4.3
6	4	6.4	6.7	4.1	4.5

Table 7: **Ablation study of the model training in image-text alignment and speech-text alignment stages.** The speech and text have clear temporal correspondence, enabling low-cost alignment. In contrast, the image-text gap is larger, requiring LLM fine-tuning for better results.

Stage	LLM freeze	GPUxHour	MMStar	MathVista ^M	MMMU ^V	AI2D
image-text	✓	76	41.2	42.3	35.5	54.3
image-text	×	192	44.4	47.6	40.2	59.1
Stage	LLM freeze	GPUxHour	AIShell-2-Dev	AIShell-2-Test	LibriSpeech-Test-Clean	LibriSpeech-Test-Other
speech-text	✓	32	12.7	11.5	9.8	13.3
image-text	×	192	13.1	11.8	10.1	13.5
speech-text	×	84	12.2	11.1	9.2	12.8

which verifies the effectiveness of our model design, whether it is a next-token-prediction (NTP) loss under the stable AR mode or CTC loss under the unstable NAR mode.

On the number of layers and experts in the speech decoder. To explore the impact of the number of layers in the NAR speech decoder and the MoE module on Chinese and English speech genera-

Table 8: **Ablation study of the alignment order and joint training strategy.** The order of the alignment strategies has minimal impact on the final performance. Compared to joint training, the multi-stage alignment strategy not only significantly reduces memory requirements during training but also ensures competitive results, making it the most efficient and optimal training strategy under low-resource conditions.

Order First	Joint Training	VRAM	MMStar	HallBench	MathVista ^M	MMMU ^V	AI2D	RWQA	AIShell-2 Dev	Test	LibriSpeech Test-Clean	Test-Other
image-text	×	40GB×8	44.7	35.9	47.1	40.7	58.6	60.1	13.4	11.3	10.4	13.6
speech-text	×	40GB×8	44.4	36.7	47.6	40.2	59.1	55.9	12.7	11.5	9.8	13.3
speech-text	✓	90GB×8	44.9	37.1	47.8	40.6	59.6	60.4	12.4	11.1	9.4	13.1

tion, we conduct ablation experiments on WeNetSpeech [78] and LibriSpeech [79]. As illustrated in Table 6, the instability and fragility associated with training using the CTC loss function present significant challenges. When simply employing a single feed-forward network (*i.e.*, the number of experts is 1), it becomes increasingly difficult to reconcile the conflicting training dynamics inherent in mixed-language scenarios, particularly when dealing with varying response lengths. As a result, training the speech decoder under these conditions proves to be quite challenging. Our findings demonstrate that incrementally increasing the number of experts significantly enhances the model’s performance in bilingual speech generation, thereby underscoring the effectiveness of our MoE module design. However, we observe inconsistent preferences regarding the optimal number of layers in the speech decoder for generating speech in Chinese and English. Specifically, while four layers yield the best results for English generation, six layers are more suitable for generating Chinese speech.

LLM training in image-text alignment and speech-text alignment stages. To investigate whether training large language models affects modality alignment at different stages, we conduct an ablation study. As shown in Table 7, since speech and text data naturally have a temporal alignment relationship, freezing the LLM during alignment training still achieves competitive alignment performance. However, the gap between image and text modalities is significantly larger, and better alignment results are only achieved by unfreezing the LLM during training. Furthermore, we find that even after image-text training, there is no catastrophic forgetting of knowledge related to speech-text alignment. This validates the effectiveness and efficiency of our progressive alignment method.

Alignment order and joint training strategy. We conduct ablation studies to explore the impact of multi-stage alignment order and joint training strategies. By using 20K speech data and 500K image data, as shown in Table 8, we observe that the relative order of speech-text alignment and image-text alignment has little effect on the final performance, which indicates a low correlation between the two stages.

Due to limitations in data and computational resources, we adopt a multi-stage progressive multi-modal alignment strategy to complete the omnimodal alignment training. At any given stage, only two modalities of data are processed simultaneously. This method not only alleviates the challenges posed by missing tri-modal data but also significantly reduces computational memory requirements. With fewer computational resources and less training data, our method achieves superior omnimodal alignment results compared to existing approaches.

As shown in Table 8, it can be observed that multi-stage training requires only 40GB×8 of VRAM, which is significantly lower than the memory demands of joint training. At the same time, it achieves comparable results, making it a more efficient and practical choice in resource-constrained scenarios.

D Additional Implementation Details

OpenOmni is trained in five sequential sub-stages. Further details on these training stages are provided in Table 9.

Besides, as shown in Figure 5, we provide more details of the speech decoder design and training here. For the speech decoder, OpenOmni supports both autoregressive (AR) and non-autoregressive (NAR) methods. Specifically, the AR mode has better generation quality but a slower generation

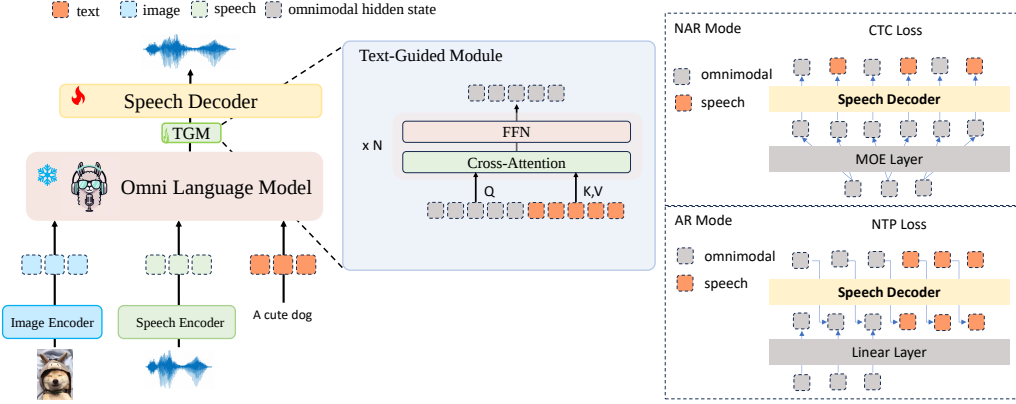


Figure 5: **Overview of text-guided module and speech decoder mode.** (Left) Text-guided module fuses the hidden state and response textual feature via cross-attention, accelerating convergence speed of training without dropping the speed of speech decoding and context emotion perception. (Right) OpenOmni supports both autoregressive (AR) and non-autoregressive speech (NAR) generation. The NAR mode uses the CTC loss modeling and a 6K speech vocabulary size to enable real-time parallel speech decoding generation. The AR mode uses the NTP loss modeling and a speech vocabulary size of 16K to support streaming decoding and higher-quality speech generation. To make the training of the speech generator more stable, we design a text-guided output feature fusion method to ensure the correctness of semantic alignment in speech generation modeling.

Table 9: The detailed training setup for OpenOmni and the hyperparameters across the training stage. All experiments are conducted on 8×NVIDIA A100-80G GPUs. The dataset index can be checked in Appendix A.

Hyperparameter	I	II	III	IV	V
batch size	256	128	128	32	32
lr	1×10^{-3}	1×10^{-3}	5×10^{-5}	5×10^{-4}	5×10^{-4}
warmup ratio	0.3	0.3	0.3	0.3	0.3
epoch	1	1	1	3	3
freeze LLM	✓	✓	✗	✓	✓
optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
cost	40 GPU Hours	80 GPU Hours	500 GPU Hours	36 GPU Hours	8 GPU Hours
dataset	1-1	2-1	2-2	3-1	3-2
loss	$\mathcal{L}_{s-t}$	$\mathcal{L}_{v-t}$	$\mathcal{L}_{v-t}^I$	$\mathcal{L}_{CTC}$	$\mathcal{L}_{CTC-DPO}$

speed, while the NAR mode can achieve real-time speech generation, but the generation quality is slightly worse. At the same time, in order to train the speech generator more efficiently, we also design a text-guided feature fusion module, so that the conditional features used for speech generation have more accurate alignment semantics, which can improve the generation quality and training efficiency of the speech decoder.

NAR mode. In the NAR mode, the conditional features generated by OLLM are fed into the speech decoder by a layer of MoE and then upsampled to obtain the predicted speech output, and finally, the end-to-end optimization is carried out by the CTC loss modeling of the speech output. Due to the instability of CTC loss training, the smaller the size of the speech vocabulary, the easier it is to be successfully trained, but the generation quality of the corresponding speech will be affected by the smaller vocabulary.

AR mode. The AR mode projects the conditional features generated by OLLM into the speech space through a layer of linear layer and feeds them into the speech decoder to obtain the speech prediction output, and finally optimizes the speech output end-to-end by modeling the NTP loss. Due to the stability of NTP loss training, the quality of speech generation will be higher than that of NAR generation, but the speed of speech generation will be reduced by AR decoding.

Note that both AR and NAR modes depend on the quality of the speech generation conditional features generated by OLLM. Although OpenOmni will let the OLLM fit the text answer corresponding to the speech through multiple rounds of training in advance, there will still be OLLM outputs decoded into the wrong text answer. In this case, the erroneously generated condition features will be incorrectly aligned with the speech during the training process, which will ultimately reduce the performance of the speech decoder. To ensure the efficiency of training, OpenOmni fuses the speech generation condition features output by OLLM with the corresponding text features with correct semantics, and then feeds them into the speech decoder for speech generation modeling training. Through the feature fusion module of text prior, OpenOmni avoids the misalignment of speech and corresponding text and ultimately makes the speech decoder training more stable. At the same time, it enjoys more efficient and accurate speech generation quality.

E Broader Impacts

OpenOmni marks a significant advancement in open-source omnimodal large language models (OLLMs), seamlessly integrating vision, language, and speech into a unified framework. Its open-source nature fosters transparency, community-driven innovation, and trust in AI technologies. However, challenges remain, including emotional manipulation and privacy concerns. To address these, we emphasize responsible AI practices and secure handling of speech data.

F Limitation

Due to resource limitations, our method primarily focuses on Chinese and English and has not been trained or validated on multilingual data. In the future, we plan to utilize multilingual speech data to activate the multilingual capabilities of the large language model, thereby enhancing its applicability across diverse scenarios.

G Prompt Template

We provide the specific prompts used in our tasks, as well as examples of synthesized multi-turn emotional speech data, to help readers better understand this work.

English Prompts used in Text-to-Speech (T2S) and Speech-to-Text (S2T) tasks.

S2T (ASR)

Please translate the user’s spoken input into corresponding text, word by word.

T2S

Echo game: Repeat this sentence back to me exactly as I said it!

Mirror challenge: Reflect my words right back at me!

Parrot mode: Can you mimic what I just said?

Copycat test: Let’s see how well you can replicate my sentence!

Playback time: Rewind and repeat my last sentence.

Echo chamber: Say back to me what I just mentioned.

Simulate my speech: Can you say exactly what I just did?

Mimicry game: Echo my words precisely!

Repeat relay: Send my sentence back to me unchanged.

Reflection exercise: Reiterate my last statement word for word.

English emotional preference data case.

Emotion: Neutral

user: "Please engage in a conversation with me."

assistant: "Of course, please go ahead with your question." [A calm and collected woman, speaking in a steady and neutral tone.]

user: "I was wondering how you manage to stay so composed all the time."

assistant: "It's something that comes with practice and mindfulness. I focus on observing my surroundings and maintaining a balanced perspective." [A composed woman, speaking in a measured and tranquil tone.]

user: "That sounds quite challenging. Do you ever feel overwhelmed?"

assistant: "There are moments when things can be overwhelming, but I remind myself to breathe and take one step at a time." [A calm woman, delivering her words in an even and serene manner.]

user: "Do you have any techniques to stay calm under pressure?"

assistant: "Yes, I practice meditation and deep breathing exercises. It helps to center my thoughts and maintain clarity." [A composed woman, speaking with a steady and soothing tone.]

user: "I see. Do you think anyone can learn to be as calm as you?"

assistant: "Absolutely. It requires patience and consistent practice, but anyone can develop the ability to remain calm and composed." [A neutral-toned woman, expressing herself in a clear and composed manner.]

user: "What advice would you give to someone starting on this path?"

assistant: "Begin with small steps. Start by observing your thoughts and emotions without judgment. Practice mindfulness and breathing exercises daily." [A composed woman, offering her advice in a gentle and calm tone.]

user: "Thank you for your advice. It has been very insightful."

assistant: "You're welcome. I'm glad I could help. Remember, the journey to calmness is gradual, but rewarding." [A tranquil woman, speaking in a reassuring and composed tone.]

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [\[Yes\]](#)

Justification: We claim the contributions and scope in both the abstract and introduction, supported by a series of justifications.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We discuss the limitation in Appendix F.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#) .

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: We provide extensive experimental details in the main content (see Section 4) and appendix (see Appendix D).

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the full instructions to access publicly available data and code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We specify all the training and test details in Section 4, Appendix A, and Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: Following previous works, one trial was performed in each experimental case due to time consumption.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provide such implementation details in Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We adhere to the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We provide the broader impact statement in Appendix E.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper involves only publicly available datasets and models. No specific safeguards were necessary.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite the relevant papers and provide links to existing assets.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset’s creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: This paper does not introduce new assets. Therefore, no documentation is applicable.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing or research with human subjects. Therefore, the inclusion of participant instructions, screenshots, and compensation details is not applicable.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: This paper does not involve LLMs for any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.