# LongCat-Flash-Omni Technical Report

**Meituan LongCat Team**
longcat-team@meituan.com

## ABSTRACT

We introduce LongCat-Flash-Omni, a state-of-the-art open-source omni-modal model with 560 billion parameters, excelling at real-time audio-visual interaction. By adopting a curriculum-inspired progressive training strategy that transitions from simpler to increasingly complex modality sequence modeling tasks, LongCat-Flash-Omni attains comprehensive multimodal capabilities while maintaining strong unimodal capability. Building upon LongCat-Flash, which adopts a high-performance Shortcut-connected Mixture-of-Experts (MoE) architecture with zero-computation experts, LongCat-Flash-Omni integrates efficient multimodal perception and speech reconstruction modules. Despite its immense size of 560B parameters (with 27B activated), LongCat-Flash-Omni achieves low-latency real-time audio-visual interaction. For training infrastructure, we developed a modality-decoupled parallelism scheme specifically designed to manage the data and model heterogeneity inherent in large-scale multimodal training. This innovative approach demonstrates exceptional efficiency by sustaining over 90% of the throughput achieved by text-only training. Extensive evaluations show that LongCat-Flash-Omni achieves state-of-the-art performance on omni-modal benchmarks among open-source models. Furthermore, it delivers highly competitive results across a wide range of modality-specific tasks, including text, image, and video understanding, as well as audio understanding and generation. We provide a comprehensive overview of the model architecture design, training procedures, and data strategies, and open-source the model to foster future research and development in the community.

**LongCat Chat**: https://longcat.ai
**Hugging Face**: https://huggingface.co/meituan-longcat/LongCat-Flash-Omni
**GitHub**: https://github.com/meituan-longcat/LongCat-Flash-Omni
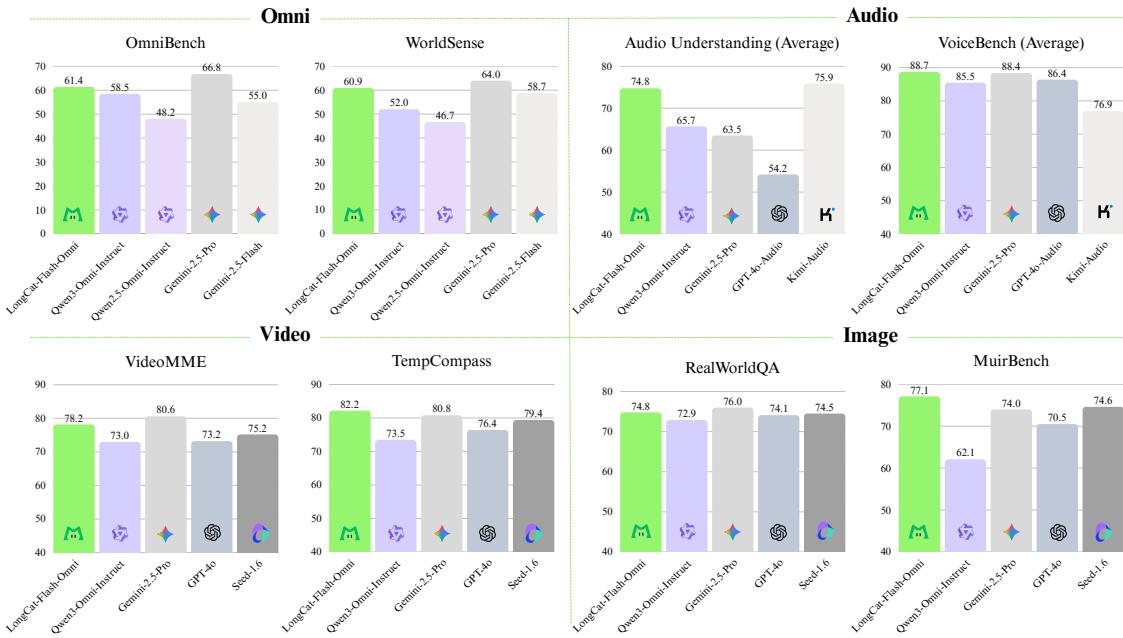


Figure 1: Benchmark performance of LongCat-Flash-Omni.

# Contents

# 1 Introduction

Humans are inherently omni-modal beings, capable of efficiently perceiving and integrating diverse forms of information, including visual and auditory inputs, to accomplish a wide range of challenging tasks. The seamless combination and transmission of multiple modalities significantly enhance the effectiveness and efficiency of human communication and interaction. In pursuit of Artificial General Intelligence (AGI), the field of large language models (LLMs) is now rapidly evolving toward the integration of richer multimodal capabilities and more efficient human-AI interaction. Recent pioneers such as Gemini-2.5 [Comanici et al., 2025] and GPT-4o [OpenAI, 2024] have integrated text, audio, image, and video processing within a single model, enabling efficient audio-visual interaction. Following these advances, research on omni-modal models has attracted broad attention, with many subsequent efforts proposed in the community [Xu et al., 2025, Guo et al., 2025, Li et al., 2025, Liu et al., 2025, Fu et al., 2025].

Training an omni-modal model that possesses both strong offline multimodal understanding and real-time audio-visual interaction capabilities is highly challenging. The difficulties mainly arise in the following aspects: (1) *Cross-modal heterogeneity*: The substantial differences among modalities require exploring effective unified representations and fusion strategies to enable synergy across modalities, ensuring that the performance of any single modality does not degrade compared to their unimodality counterparts of similar scale. (2) *Unified offline and streaming capabilities*: Integrating offline multimodal understanding with streaming audio-visual interaction presents a significant challenge. Streaming interaction scenarios require distinct capabilities not typically found in offline processing, such as the perception of relative time, precise synchronization of audio-visual information, and efficient management of multi-turn interaction contexts. (3) *Real-time interaction*: Achieving real-time audio-visual interaction presents inherent difficulties, including the necessity of supporting both streaming audio and video input, as well as streaming speech output. The stringent low-latency requirement further imposes strict constraints on computational efficiency, thus placing high demands on both model architecture design and deployment infrastructure. (4) *Training Efficiency*: The heterogeneity within model and data poses great challenges for the design of distributed strategies.

In this report, we attempt to address the aforementioned challenges. To overcome the first challenge, we carefully design a multi-stage large-scale pretraining pipeline. Based on an early-stage text pretraining foundation model, we progressively incorporate audio and visual data into the large-scale pretraining process. By adopting a balanced multimodal data mixture and effective ealy-fusion strategy, the model achieves deeply integrated comprehension across modalities while maintaining strong unimodal performance.

To address the second challenge of balancing offline multimodal understanding with real-time audio-visual interaction, we introduce a human-in-the-loop strategy to construct high-quality interaction data, with careful consideration for long-term memory and multi-turn dialogue handling. In addition, we derive vision-speech question-answering data from existing vision-text corpora, enabling natural speech output and facilitating the transfer of strong offline multimodal understanding capabilities into interactive scenarios.

To address the third challenge of achieving low-latency audio-visual interaction in a large-scale model, we dedicate substantial effort to the design of all modules in LongCat-Flash-Omni. We adopt the ScMoE architecture with zero-computation experts from LongCat-Flash [Meituan, 2025a,b] as the LLM backbone. To handle streaming input, we employ efficient audio and video encoders for feature extraction and introduce a synchronized chunk-wise interleaving strategy for real-time processing. For efficient audio reconstruction, we adopted a multi-codebook audio detokenization scheme with a coarser temporal resolution. This approach significantly improves decoding efficiency while preserving reconstruction quality. Furthermore, we designed an efficient streaming pipeline for model serving to minimize end-to-end server-side latency. As a result, despite having up to 560B parameters (with 27B activated), LongCat-Flash-Omni achieves millisecond-level response latency in real-time interactive scenarios.

To address the fourth challenge of training efficiency, we devote substantial effort to large-scale omni-modal distributed training. We propose a modality-decoupled parallelism (MDP) strategy. This approach enables independent optimization of both performance and memory usage for the LLM, vision encoder, and audio encoder. For the LLM component, we systematically optimize the distributed training configuration for computational efficiency and apply multiple memory-reduction techniques to ensure robust system stability throughout training. Experimental results demonstrate the effectiveness of this strategy, our system sustains more than 90% of the throughput achieved during pure text training.

Comprehensive evaluations demonstrate that our model delivers strong and consistent performance across both omni-modal benchmarks and real-time interactive tasks. LongCat-Flash-Omni achieves state-of-the-art results on omni-modal benchmarks such as Omni-Bench and WorldSense, while also exhibiting highly competitive performance on a wide range of unimodal tasks, including text, image, and video understanding, as well as speech comprehension and generation, establishing itself as the most powerful omni-modal model in the open-source community. Furthermore,

extensive subjective evaluations confirm that LongCat-Flash-Omni supports high-quality audio-visual interaction with low latency.

The key features of LongCat-Flash-Omni are summarized as follows:

- **SOTA and Unified Omni-Modal Model:** LongCat-Flash-Omni achieves state-of-the-art cross-modal comprehension performance among open-source models. It seamlessly integrates powerful offline multimodal understanding with real-time audio-visual interaction within a single all-in-one framework.

- **Large-Scale with Real-time Audio-Visual Interaction:** By leveraging an efficient LLM backbone, carefully designed lightweight modality encoders and decoder, and a chunk-wise audio-visual feature interleaving mechanism, LongCat-Flash-Omni achieves low-latency, high-quality audio-visual processing and streaming speech generation. It supports a context window of up to 128K tokens, enabling advanced capabilities in long-term memory, multi-turn dialogue, and temporal reasoning across multiple modalities.

- **Effective Early-Fusion Training:** The model adopts an innovative multi-stage pretraining pipeline that progressively incorporates text, audio, and visual modalities under a balanced data strategy and early-fusion training paradigm, ensuring strong omni-modal performance without degradation in any single modality.

- **Efficient Training Infrastructure:** Inspired by the concept of modality decoupling, we propose a modality-decoupled parallelism training scheme that significantly enhances the efficiency of large-scale and highly challenging multimodal training.

- **Open-Source Contribution:** We provide a comprehensive overview of the training methodology and data strategies behind LongCat-Flash-Omni, and release the model to accelerate future research and innovation in omni-modal intelligence.

The remainder of this paper is organized as follows. Section 2 presents the model architecture of LongCat-Flash-Omni. Sections 3 and 4 describe the pretraining and post-training datasets and pipelines, respectively. Sections 5 and 6 introduce training infrastructures and inference deployment. Section 7 reports the experimental results. Section 8 draws a conclusion of this report.

## 2   Architecture

As illustrated in Figure 2, LongCat-Flash-Omni is a fully end-to-end omni-modal model. It can receive various modalities as input, including text, audio, image, video, and their any arbitrary combination, and is able to directly generates speech tokens from the LLM backbone. LongCat-Flash-Omni employs a vision encoder and an audio encoder as multimodal perceivers. An LLM processes the multimodal inputs and generates text and audio tokens. An audio decoder reconstructs the waveform from the speech tokens generated by the LLM, enabling natural speech interaction. All modules are carefully designed to support efficient streaming inference. The audio encoder, vision encoder, and audio decoder are lightweight components, each with approximately 600M parameters. The large-scale LLM backbone uses the novel and efficient architectural design proposed in the LongCat LLM family [Meituan, 2025a,b].

In the reminder of this section, we first introduce each structural component of LongCat-Flash-Omni, including the vision encoder that supports inputs with arbitrary aspect ratios and native-resolution feature encoding, the audio encoder, decoder, and tokenizer, and the LLM backbone. Then we elaborate on the video processing strategy and the architectural components that enable low-latency, real-time audio-visual interaction.

### 2.1   Vision Encoder

The vision encoder serves as a critical component in multimodal language models. To effectively encode visual inputs such as images and videos, LongCat-Flash-Omni incorporates a well-designed Vision Transformer (ViT), referred to as LongCat-ViT [Qiao et al., 2025]. LongCat-ViT achieves high performance across multimodal tasks, natively supports inputs with various resolutions and aspect ratios, while providing unified encoding capabilities for both image and video data.

**Architecture Design** LongCat-ViT is a Transformer-based encoder that retains the core structure of the conventional Vision Transformer while integrating several key enhancements: a unified patchification module for native image and video inputs, 2D rotary position embeddings (2D-RoPE) [Su et al., 2024], SwiGLU activation function, RMSNorm layer, a LayerScale module, and Query-Key normalization. Together, these refinements yield a more robust and efficient architecture compared to conventional designs. To improve computational efficiency in video frame encoding during real-time interaction while maintaining model performance, we adopt a relatively lightweight model configuration. The detailed hyper-parameters of the model are provided in Table 1. In line with common practice, a two-layer multilayer
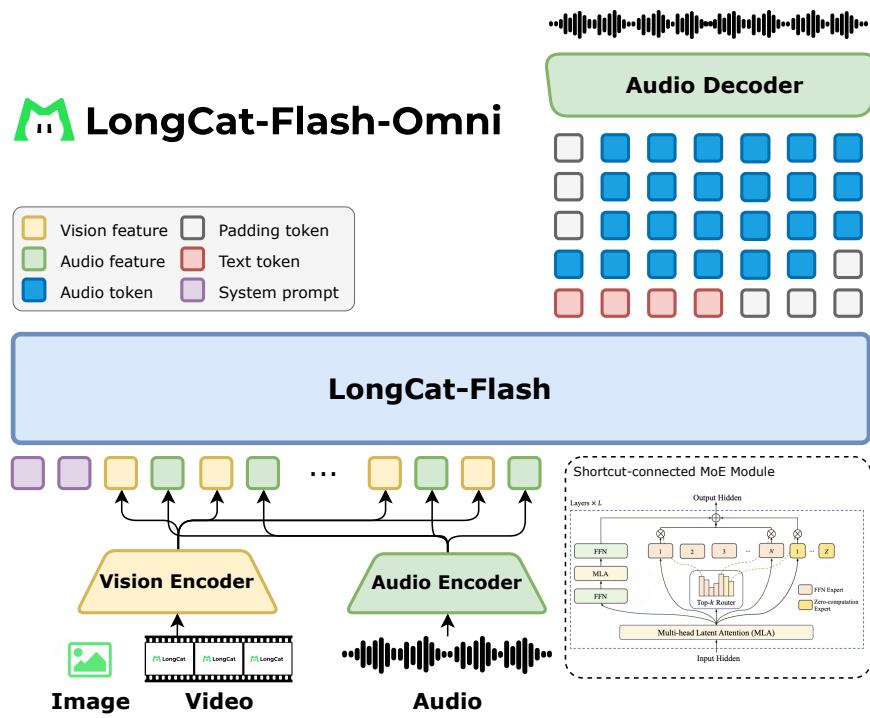
Figure 2: An overview of the LongCat-Flash-Omni model architecture. The model is fully end-to-end and unifies multimodal understanding and generation across text, image, video, and audio within a single large language model framework. An vision encoder and an audio encoder are used to obtain vision features and audio features, respectively, which are then projected into a shared latent token space and fed into the LongCat-Flash LLM backbone. The LLM decoder directly generates multi-codebook speech tokens, parallel to generated text tokens, which are then converted to audio waveforms by an audio decoder. Shortcut-connected MoE (ScMoE) with zero-computation experts module proposed in LongCat-Flash is employed to achieve efficient multimodal fusion. Vision and audio features are chunk-wisely interleaved to support streaming audio-visual input.

perceptron (MLP) with pre-normalization is employed as the vision-language projector to align visual and textual representations. Moreover, a $2\times$ pixel-unshuffle operation is applied along the spatial dimension to mitigate the quadratic computational complexity associated with high-resolution inputs.

Table 1: Detailed architectural configuration for LongCat-ViT.

| Patch Size | Pos Embed | Hidden Size | Intermediate Size | Num Layers | Num Heads | Parameters (M) |
|---|---|---|---|---|---|---|
| 14 | 2D-RoPE | 1280 | 5184 | 32 | 16 | 637 |

**Native Resolution Encoding** Conventional ViT models widely adopted in the community (e.g., CLIP [Radford et al., 2021], SigLIP [Zhai et al., 2023]) typically resize input images to a fixed resolution, which can result in substantial information loss, particularly for images with extreme aspect ratios or high native resolutions. To mitigate this limitation, LongCat-ViT encodes visual inputs at their native resolutions, circumventing the limitations of fixed-resolution ViT models. This preserves the spatial and contextual information inherent in visual data, thereby enhancing the model's capability to comprehend and reason over complex visual inputs. For each image or video frame, if the number of patches falls within a predefined range (576-5832 during training), only minimal resizing is applied to ensure both dimensions are divisible by 112. Otherwise, the image is rescaled to fit within this range while preserving its aspect ratio.

**Contrastive Vision-Language Pretraining** LongCat-ViT adopts a progressive training scheme that integrates two complementary adaptation strategies: (1) progressive resolution adaptation, which leverages curriculum learning by transitioning from fixed low-resolution (e.g., 224) pretraining to native-resolution fine-tuning; and (2) progressive visual modality adaptation, which postpones the incorporation of video data until the final training stage to reduce
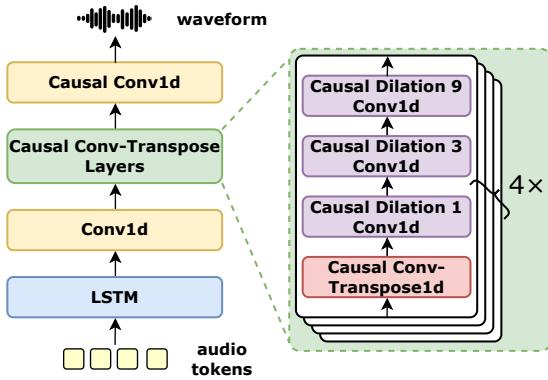
Figure 3: Architecture of the audio decoder.
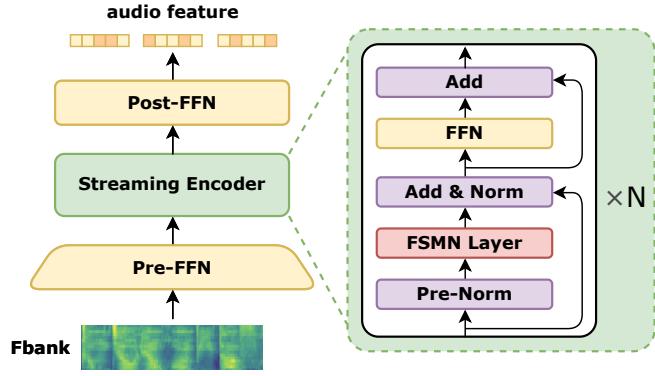


Figure 4: Architecture of the audio encoder.

computational overhead. To further facilitate convergence in the early phases, feature distillation from a frozen pretrained vision model is introduced as an auxiliary objective, and the weight of this objective gradually reduces in later stages. The model is trained from scratch on a total of 14.6 billion samples during the contrastive pretraining phase.

## 2.2 Audio Tokenizer, Encoder, and Decoder

We input audio in different formats to the LLM backbone of LongCat-Flash-Omni across training phases. Specifically, during pre-training stages 1-4, an *audio tokenizer* is employed to convert raw speech into four-codebook discrete tokens, enabling consistent next-token prediction and improved training efficiency. However, we observe that such discretization hinders the model's ability to capture fine-grained acoustic details. Therefore, from pre-training stage 5 (Section 3.2.6) we incorporate an *audio encoder* to convert raw speech into continuous audio features before input to the LLM. For speech generation, to align with the inherent next token prediction paradigm, the LLM consistently outputs the four-codebook discrete tokens, which are subsequently converted back into a waveform using an *audio decoder*.

**Audio Tokenizer and Decoder** We adopt LongCat-Audio-Codec [Zhao et al., 2025a] as our audio *tokenizer* and *decoder*, owing to its robust semantic modeling, flexible acoustic feature extraction, and low-latency streaming synthesis capabilities. The tokenizer discretizes audio waveforms into four codebooks at a frame rate of 16.67 Hz, with one codebook representing semantic information and the other three capturing acoustic details. To achieve low-latency inference in real-time interaction scenarios, unlike conventional waveform reconstruction methods that rely on diffusion- or flow-matching-based code2mel models followed by vocoders, we directly employ the decoder from LongCat-Audio-Codec as the audio decoder to reconstruct waveform from tokens. It supports streaming audio decoding with a look-ahead of only three frames. As illustrated in Figure 3, the audio decoder consists of LSTM layers, convolutional blocks, and causal transposed convolution layers, and is trained under a generative adversarial network (GAN) framework.

**Audio Encoder** To optimize response latency and accommodate speech inputs of arbitrary duration, the audio encoder is designed using a streaming architecture. As illustrated in Figure 4, the audio encoder takes 80-dimensional Fbank features as input. The architecture incorporates a Pre-FFN module that reduces the audio sequence length by a factor of eight through frame splicing downsampling technique, with each frame representing an 80ms time window. The core process is performed by a streaming encoder that maintains a Transformer-like structure while incorporating several key modifications: (1) a Pre-Norm configuration for enhanced training stability, and (2) replacing standard self-attention modules with FSMN layers [Zhang et al., 2018] to enable efficient feature processing within constrained context windows. To balance latency and performance, we implement a hybrid approach where only the final six layers incorporate a one-frame look-ahead mechanism, while maintaining strict causality in the preceding layers. The architecture concludes with a post-FFN module for additional feature refinement. The audio encoder is trained under supervised learning using speech recognition data using the CTC loss [Graves et al., 2006].

## 2.3 LLM Backbone

LongCat-Flash-Omni is built upon LongCat-Flash [Meituan, 2025a], a 560-billion-parameter Mixture-of-Experts (MoE) language model. LongCat-Flash adopts Multi-head Latent Attention (MLA) [Liu et al., 2024a], shortcut-connected MoE [Cai et al., 2024] and zero-computation experts, performing variable computation per token by activating 18.6B-

31.3B parameters (27B on average), thereby unifying efficiency, performance and sparsity. These characteristics are retained and extended to multimodal understanding and audio-visual interaction by LongCat-Flash-Omni.

## 2.4    Video Strategy and Streaming Audio-Visual Interaction

LongCat-Flash-Omni is designed to seamlessly integrate robust offline multimodal understanding with low-latency audio-visual interaction. The audio and visual streams are independently processed by an audio encoder and a vision encoder, respectively. Their extracted features are subsequently time-aligned and chunked into synchronized segments, which are interleaved and fed into the LLM decoder for multimodal understanding. Here we elaborate the video strategy adopted by LongCat-Flash-Omni and how audio-visual input is processed to support streaming interaction.

### 2.4.1    Video Strategy

Efficient video processing remains a significant challenge due to the substantial variability in video properties, including durations ranging from a few seconds to several hours and resolutions spanning a wide spectrum. To address these challenges, we adopt a series of strategies to effectively balance between model performance and computational efficiency.

**Dynamic Video Frame Sampling** We adopt a default sampling rate of 2 frames per second (2 FPS), while also enable dynamic adjustments according to video duration. During training, shorter videos are sampled at a higher frame rate to capture denser temporal information, ensuring at least 16 frames for effective information utilization. In contrast, for excessively long videos, frames will be sampled uniformly based on a maximum frame number constraint. This frame cap further facilitates training by regulating memory consumption and preserving computational efficiency.

**Textual Timestamps** LongCat-Flash-Omni introduces timestamps preceding each input video frame to strengthen the model's temporal awareness and improve its ability to recognize specific time points. The timestamps are input as pure text that are naturally aligned with the textual space. Similar to [Chen et al., 2024a], when sampling a video frame at second $i$, we prepend the text "$Second\{i\}$" to its corresponding visual tokens before inputting them into the LLM. The resulting input sequence is structured as "$Second\{i\}||\mathbf{V}_i||Second\{j\}||\mathbf{V}_j||\ldots$", where $Second\{i\}$ is the textual timestamp, $\mathbf{V}_i$ represents the visual tokens of the video frame at second $i$, and $||$ indicates concatenation.

**Hierarchical Token Compress in Video Inputs** We compress the video inputs with three successive steps: First, we rescale the each frame according to a predefined upper limit on the number of patches as described in Section 2.1. Then we apply a 3D convolution with temporal stride of 2 before feeding the video into the visual encoder, compressing the $N$ input frames to $N/2$ in temporal size. Finally, after the video is processed by the visual projector into visual tokens, if the number of visual tokens exceeds a predefined limit, we further perform an interpolative downsample on the visual tokens.

### 2.4.2    Streaming Audio-Visual Interaction

The streaming audio-visual interaction mechanism is a core component of LongCat-Flash-Omni, enabling real-time integration of video and speech signals to support interactive communication. The proposed audio-visual interaction framework is characterized by two key aspects:

**Streaming Audio-Visual Feature Interleaving** Unlike offline audio-visual understanding tasks, where audio and visual features can be concatenated at the sequence level, real-time audio-visual interaction requires features from the audio and video streams to be prefilled into the LLM backbone as early as possible to minimize response latency once the user query is received. To achieve this, we design a temporally-synchronized, chunk-wise audio-visual feature interleaving mechanism. Audio-visual feature chunks are structured as "<|timestamp|>:<|video-tokens|><|audio-start-token|><|audio-tokens|><|timestamp|>:<|video-tokens|><|audio-tokens|>...<|audio-end-token|>", where the timestamp is represented in textual form, as described in Section 2.4.1.

**Sparse-Dense Sampling Strategy** We design a sparse-dense sampling strategy to optimally balance computational cost and information loss during turn-taking interactions between the user and the model. Specifically, we employ a chunk size of 1 second during the information input period to preserve as much audio-visual information as possible, using a denser video sampling rate of 2 FPS. During the model response period, video frames are buffered with a sparser sampling rate (i.e., chunk size of 2 seconds, 0.5 FPS) and prepended to the next user turn. This design effectively balances visual information retention during the model response period and computational overhead, enabling high-quality audio-visual interaction, a core capability that distinguishes it from other omni-modal models in the community.

# 3 Pre-Training

This section describes the data curation process (Section 3.1) and training strategy (Section 3.2) of LongCat-Flash-Omni.

## 3.1 Data Curation

We collect a large-scale and diverse multimodal corpus with over 2.5 trillion tokens for pre-training. This pre-training corpus consists of the following components: audio data (Section 3.1.1), generic image-text data (Section 3.1.2), OCR, grounding and GUI data (Section 3.1.3), STEM data (Section 3.1.4), multi-image data (Section 3.1.5), video data (Section 3.1.6), and long-context multimodal data (Section 3.1.7).

### 3.1.1 Audio Data

**Speech-Text Interleaved Data** Our data consists of tens of millions of hours of audio with rich diversity. We employ the following pipeline to extract high-quality speech audios with consistent topics: First, we use a VAD system to split a long audio into speaking segments and remove non-speaking regions. Next, we use two proprietary ASR models for cross-validation, filtering out segments with significant discrepancies on transcription results. A multilingual speech aligner [Pratap et al., 2024] is then applied for forced alignment, yielding precise transcription timestamps. We further refine the dataset by computing the ratio between speech duration and text length for each segment, and discard the segments whose ratios fall outside the $0.5$-th to $99.5$-th percentile range. Finally, we merge adjacent speaking segments separated by silences shorter than 10 seconds to form the training samples.

To build speech-text interleaved training data, we first split each training sample into fragments using punctuation marks as delimiters: $(A_1, T_1), (A_2, T_2), \ldots, (A_n, T_n)$, where $A_i$ and $T_i$ represent the $i$-th audio fragment and its transcribed text, respectively. We then randomly mask either audio or text component of some fragments, resulting in the final training input such as $(T_1, A_2 + T_2, T_3, A_4 + T_4, \ldots)$ or $(A_1, A_2 + T_2, A_3, A_4 + T_4, \ldots)$.

**Audio Understanding Data** We curate an audio understanding dataset that encompasses a wide range of tasks, including audio captioning, semantic audio understanding, paralinguistic analysis, acoustic scene and event detection, audio question answering, and music understanding. The dataset comprises a combination of open-source datasets and in-house proprietary datasets.

We apply text translation as data augmentation for speech recognition datasets, various speech models for pseudo-label generation and label quality filtering, particularly for paralinguistic understanding and audio captioning datasets, and a diverse set of prompts for each task to enhance instruction variability.

### 3.1.2 Generic Image-Text Data

**Image Caption Data** High-quality image captions are crucial for aligning visual representations with the language model's knowledge space. To this end, we build a large-scale image-caption dataset by first applying a multi-stage cleaning pipeline across text, image, and image-text pair levels. Low-quality samples, such as those with extremely short text, abnormal resolutions, or poor image quality, are removed using elaborately-refined heuristic rules. At the pair level, we further filter samples using a SigLIP similarity threshold, discarding those below a strict minimum to preserve downstream diversity. We then improve the dataset through re-captioning to generate dense, fine-grained captions with multiple open-source vision-language models while incorporating world knowledge from original annotations. Additional filtering, including mixed-language detection, repetition removal, and truncation handling is applied to help mitigate hallucinations and ensure the final image-text pairs are accurate, informative, and contextually rich.

To broaden semantic coverage and avoid overfitting to dominant patterns, we significantly enhance data diversity and balance. We cluster image-text pairs based on joint image and text embeddings and resample from each cluster to ensure representation of long-tail content, while simultaneously removing low-quality samples frequently concentrated in specific clusters. Experimental results confirm that preserving diversity at this stage is more beneficial than overly strict quality filtering. Furthermore, to address the pronounced long-tail distribution inherent in multimodal datasets, we adopt a concept-based resampling strategy inspired by MetaCLIP [Xu et al., 2023]. By expanding the vocabulary with a 200K-scale Chinese lexicon and resampling for broader concept coverage, we achieve a more balanced distribution across semantic categories.

**Interleaved Image-Text Data** Interleaved image-text data provides broader visual-textual coverage and improves few-shot performance, but it is often noisy and has uneven quality. To address this, we construct a high-quality dataset through a two-stage pipeline of filtering and diversity sampling from open-source data. In the filtering stage, we remove samples with noisy tokens, sensitive content, and overly complex samples, and discard corrupted or low-resolution images. We then improve image-text alignment by pairing each image with its most relevant text segment by SigLIP

similarity scores, and discard pairs with low semantic alignment. To sample a diverse and evenly distributed subset, we apply density-based pruning and semantic clustering similar to [Abbas et al., 2024]. Samples are scored by knowledge content, quality, and similarity, then evenly sampled based on the cluster they belong to and their distances to the cluster centroid. This process reduces the raw dataset by approximately 74%, while maintaining diversity and quality for multimodal pretraining.

To enhance domain knowledge and reasoning capabilities, we further curate an in-house, high-quality image-text interleaved dataset derived from video content containing educational materials. We use an automated pipeline to select only instructional video segments and remove other parts. Textual information is extracted using ASR and OCR, then refined by an LLM to improve accuracy and consistency. Videos are segmented into meaningful scenes, and key frames are extracted and aligned with the refined text based on temporal and semantic correspondence. The resulting dataset comprises structured, context-rich sequences of synchronized visual and textual information, significantly strengthening the model's capacity for academic reasoning and multimodal understanding.

### 3.1.3 OCR, Grounding and GUI Data

**OCR Data** We curate richly annotated training samples encompassing various content types, including PDFs, papers and web pages. These efforts further enrich our document parsing dataset, enabling the model to learn both content extraction and structural understanding, and resulting in more robust and accurate document parsing. We further include diverse OCR datasets covering scene text, structured documents, handwriting, and mathematical expressions. We also synthesize multi-page and region-level OCR samples from scene and document OCR data, with each sample comprising 2 to 6 pages. For OCR-related VQA, we incorporate a diverse range of datasets for visual question answering, covering domains of text-centric VQA, document VQA, table VQA, and chart VQA.

**Grounding Data** To enhance the model's grounding capabilities, we adopt several widely used open-source object detection datasets [Krishna et al., 2017, Mao et al., 2016, Lin et al., 2014, Shao et al., 2019]. We perform data quality validation to filter out incorrect and redundant samples. Using these high-quality open-source datasets, we construct two types of question-answering data: localization data and region-captioning data. We uniformly adopt relative coordinates, which are normalized to the range of 0-1000, and process the data into JSON format. We also incorporate PixMo dataset [Deitke et al., 2024] to enhance the model's counting ability.

**GUI Data** Graphical User Interface (GUI) data contains rich information on visual understanding and task planning, enabling automated interactions on both mobile and desktop platforms. To this end, we utilize diverse types of GUI-related data to enhance the model's perception, grounding, and planning capabilities [Zeng et al., 2025]. For GUI perception, we utilize a large number of image-text pairs from screenshots of various PC and mobile applications, covering tasks such as OCR, VQA, and captioning with GUI images. For GUI grounding, we construct instruction-answer pairs based on visual screenshots from diverse platforms and devices. For each screenshot we create multiple pairs, accounting for different UI elements and interaction possibilities. For GUI planning, we collect a diverse set of navigational paths with rich context from both mobile and desktop. The planning data comprises key components such as screenshot observations, summarizations, reasoning traces, and corresponding actions. To enhance the model's reasoning capability, we integrate multiple inference settings that involve different combinations of action, summarization, and reasoning outputs. Furthermore, to capture the underlying logic of GUI dynamics, we incorporate contextual scenarios that enable the model to predict changes across different screenshots.

### 3.1.4 STEM Data

To strengthen the model's foundational capabilities in scientific reasoning and problem-solving, we construct a large-scale, multimodal STEM (Science, Technology, Engineering, and Mathematics) dataset. The collected data are meticulously processed and structured into both multiple-choice and open-ended generative question-answer formats. We implement a rigorous multi-stage filtering pipeline to ensure factual accuracy, eliminate ambiguity, and standardize formatting. This effort culminates in a high-fidelity pre-training dataset comprising 15 million image-text pairs with substantial diversity in both subject matter and academic level, and covering a wide range of disciplines from K12 education to advanced university studies. This foundational dataset is essential for enabling the model to achieve deep conceptual understanding and robust reasoning capabilities across a broad spectrum of scientific and technical domains.

### 3.1.5 Multi-Image Data

To enhance the model's fine-grained image understanding, we constructed a taxonomy with various coarse-grained and fine-grained capabilities, such as "emotion recognition", "vehicle identification", and "time calculation with clocks". Based on this taxonomy, we employed diverse strategies to collect multi-image question answering data, including open-source datasets with careful data selection and augmentation, and synthesizing images using dedicated tools.
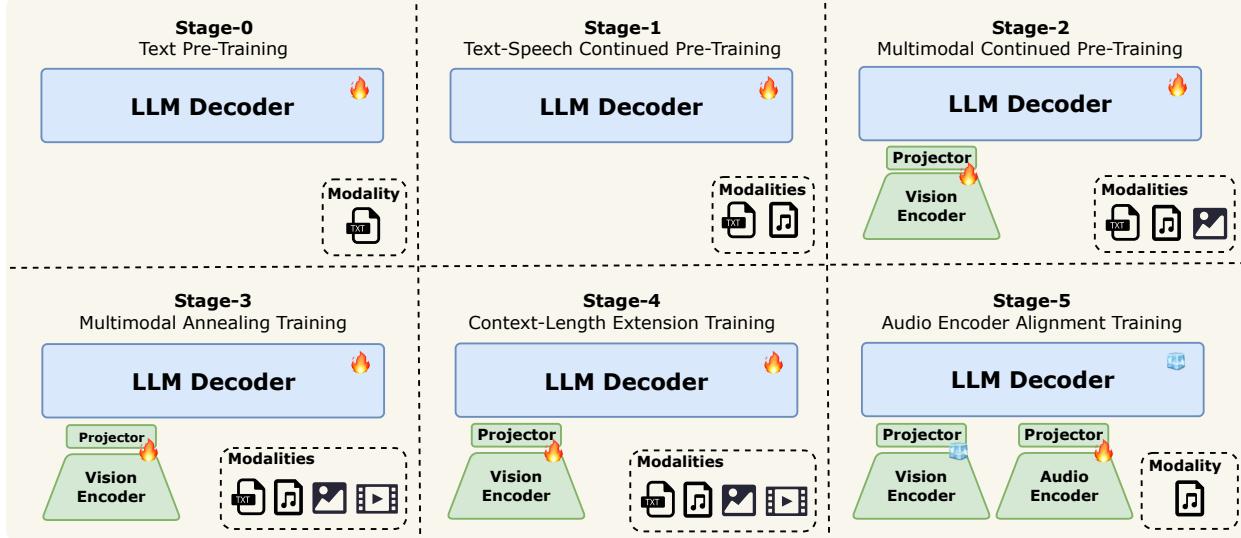
Figure 5: Pre-training stages.

### 3.1.6 Video Data

Our video dataset is primarily sourced from a broad range of publicly available corpora, covering diverse task types such as video classification, temporal grounding, detection, captioning, and question answering. We design a comprehensive data processing pipeline centered on rigorous quality filtering and targeted enhancement, resulting in a refined, high-quality dataset tailored for large-scale pre-training.

Additionally, we curate an in-house dataset from publicly available video content with three components:

**High-quality video caption data** Our video captioning process begins by dividing videos into coherent scenes using a scene detection algorithm similar to Koala [Wang et al., 2025a]. To ensure a wide range of content, we then cluster and sample these scenes. Finally, we employ a proprietary model to generate detailed captions that describe the sequence of events and the overall context of the video.

**Temporally grounded video QA data** We construct a high-quality temporally grounded dataset by applying rule-based transformations to convert annotations from tasks such as temporal action detection, segmentation, video summarization, and temporal sentence grounding into question-answer (QA) pairs. Data complexity is further enriched through a proprietary model that generates more challenging and in-depth QA pairs.

**Action recognition video QA data** To strengthen the model's action recognition capability, we transform several publicly available video recognition datasets [Carreira et al., 2019, Goyal et al., 2017] into multiple-choice and open-ended video QA pairs.

### 3.1.7 Long-Context Multimodal Data

To address the challenges in long-context multimodal scenarios, we construct a specialized long-context multi-modal dataset. The dataset contains two main parts: (1) carefully filtered samples from open-sourced data and in-house pre-training data, with a focus on videos exceeding three minutes in length, and (2) in-house long-context synthesized data.

Our in-house long-context dataset includes image-text interleaved data and long-video QA data. Image-text interleaved data is constructed by concatenating single-image data with the same topic or rendering some text segments within long text into images. This data can help the model to improve in-context learning and information retrieval from long context (i.e., "a needle in a haystack"). Long video QA data is constructed by the following pipeline: (1) First we segment videos into multiple clips and generates rich descriptive captions for each clip. (2) For consecutive clips from the same video, we analyze their scene continuity from their captions. If the clips are identified as continuous, we concatenate and refine their captions into a coherent longer-form video caption. (3) Finally, we construct temporally grounded and detailed QA pairs to form our long-video QA dataset. This data can help the model to enhance long-sequence cross-modal modeling and memory retention.

## 3.2 Training Strategy

One of the most fundamental challenges in training omni-modal models lies in the significant heterogeneity of data distributions across modalities. Each modality exhibits distinct structural properties and representational characteristics. Text, for instance, is a highly compressed and abstract symbolic representation of human knowledge, and LLMs have demonstrated remarkable success in modeling large-scale text sequences. Speech, on the other hand, is the acoustic manifestation of human thoughts and concepts, a sequential signal like text, but enriched with paralinguistic information such as speaker timbre, emotion, prosody, and accent. However, its semantic density is much lower than that of text: while a typical speech tokenizer operating at 12.5 Hz must generate roughly 12.5 tokens per second, humans only speak about 3-4 text tokens per second. This mismatch makes sequence modeling for speech inherently more challenging than for text.

Visual information introduces a fundamentally different modality. Unlike the sequential nature of text and speech, images encode spatial structures, requiring the model to reason over spatial relationships. Video data further increases the complexity by incorporating both spatial and temporal dynamics, making it a much more complicated modality to model effectively. This requires careful consideration of effective frame sampling strategies, token compression methods, and temporal modeling approaches, as well as the ability to handle long-range sequence dependencies.

Guided by these observations, we adopt a curriculum-inspired, progressive training strategy that gradually transitions from simpler to more complex sequence modeling tasks, as illustrated in Figure 5. We begin with large-scale text pretraining (**Stage-0**), leveraging the maturity and stability of LLMs as a strong initialization for subsequent multimodal learning. Building on this foundation, we introduce speech data, which is structurally closer to text, to align acoustic representations with the language model's feature space and effectively integrate paralinguistic information (**Stage-1**). Once speech-text alignment is established, we incorporate large-scale image-caption pairs and vision-language interleaved corpora (**Stage-2**) for vision-language alignment, enriching the model's visual knowledge. We then introduce the most complex video data to enable spatial-temporal reasoning (**Stage-3**), meanwhile integrating higher-quality and more diverse image datasets to enhance vision comprehension. To further support long-context reasoning and multi-turn interactions, we extend the model's context window from 8K to 128K tokens (**Stage-4**). Finally, to mitigate information loss in audio inputs represented by discrete speech tokens, we introduce an audio encoder alignment stage (**Stage-5**) that enables the model to directly process continuous audio features, thereby enhancing fidelity in downstream speech tasks.

The following part of this section presents details of each pre-training stage.

### 3.2.1 Stage-0 Text Pre-Training

To establish a robust text foundation, the text pre-training stage follows the same procedure as the initial phase of LongCat-Flash [Meituan, 2025a]. The model is trained on approximately 16 trillion tokens drawn from a high-quality and diverse text corpus, where a constant learning rate is applied. Throughout training, the proportion of high-quality reasoning data (e.g., STEM and code) is progressively increased to strengthen the model's reasoning and programming capabilities, thereby providing a solid foundation for subsequent multimodal learning.

### 3.2.2 Stage-1 Text-Speech Continued Pre-Training

Building upon the text foundation model obtained from stage-0, we continue pre-training using a mixture of text data, speech-text interleaved data, and ASR data. All speech samples are discretized into four-codebook token sequences. During training, we jointly optimize multiple objectives: (1) pure text next-token prediction (NTP) to preserve strong text understanding and generation capabilities; (2) text-speech interleaved NTP to align speech and text representations within a unified sequence modeling framework; and (3) ASR-style tasks to build basic speech perception capability. As shown in Figure 6, text and audio embeddings are fused before being fed into the LLM decoder. We introduce four audio prediction heads, enabling LongCat-Flash-Omni to directly generate audio tokens. The model simultaneously predicts text tokens, semantic tokens, and acoustic tokens, with a one-step temporal offset between semantic and acoustic token predictions. Empirical observations indicate that ASR tasks contribute minimally to modality alignment, prompting us to include only a small proportion of ASR data in the training process. The overall training objective is defined as follows:

$$\mathcal{L}_{\text{total}} = a\,\mathcal{L}_{\text{pure-text}} + b\,\mathcal{L}_{\text{audio}} + c\,\mathcal{L}_{\text{audio-text}} + d\,\mathcal{L}_{\text{first-audio}}$$

where $a$, $b$, $c$ and $d$ are loss weights. $\mathcal{L}_{\text{pure-text}}$ denotes the loss for pure text data, while $\mathcal{L}_{\text{audio}}$ and $\mathcal{L}_{\text{audio-text}}$ correspond to the audio loss and the text loss terms in the text-speech data. $\mathcal{L}_{\text{first-audio}}$ is an additional loss term applied to the semantic audio token. After carefully hyperparameter tuning and extensive preliminary experiments, we set $a = 1.75$, $b = 0.25$, $c = 1.5$, and $d = 0.1$, which achieves the best balance between preserving text capabilities and enhancing audio performance.

This training stage uses approximately 5.1 trillion tokens, with the ratio of text tokens and audio tokens is 2:1. A slight learning rate decay is applied.
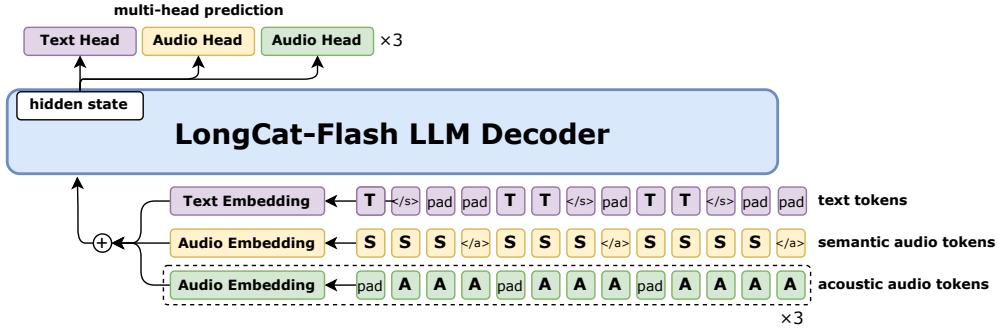


Figure 6: Schematic diagram of pre-training stage-1, where large-scale speech and text data is employed for the training process.

### 3.2.3 Stage-2 Multimodal Continued Pre-Training

Based on the model from stage-1, we further incorporate large-scale image-text data into the pre-training procedure, including image caption data, and interleaved image-text data. We use a vision transformer (ViT) with weights initialized from the LongCat-ViT model introduced in Section 2 to obtain visual features from images, and employ a randomly initialized vision projector to align the visual feature with the latent space of the LLM backbone. We maintain the text-to-audio data ratio the same as in stage-1, i.e., 2:1, and make the text-to-vision data ratio also 2:1. In total, this pre-training stage consumes more than 3 trillion tokens. The parameters of the ViT module and projector are jointly trained with the LLM decoder parameters throughout the stage-2 training, with a nearly constant learning rate. We reuse the loss weights in stage-1 and set the additional vision-related loss weight at 0.25 during this training process.

### 3.2.4 Stage-3 Multimodal Annealing Training

Following the pre-training stage, the model undergoes a multimodal annealing phase, where the model's training continues with a curated higher-quality data under an annealed learning rate to achieve superior performance. We further incorporate video data, including video captioning and QA datasets, as well as a broader range of image-related data such as OCR, grounding, GUI, multi-image, and STEM datasets. We maintain the stage-2 data ratio to ensure cross-modal stability, using a text:vision:speech token ratio of 2:1:1, consuming 0.33 trillion tokens in total.

The effectiveness of the model is highly dependent on the quality and composition of the training data. Therefore, the domain mixture plays a crucial role in determining the final performance. Vision data, in particular, exhibits greater heterogeneity in content distribution, learning difficulty, and scale, thereby requiring more careful control over data composition.

We adopt a perplexity (PPL)-gap-based signal to automatically guide data sampling allocation. The corpus is first segmented into distinct subsets according to semantics and tasks. During training, we monitor PPL convergence for each subset: if a subset's convergence lags behind the expected reference level, its sampling weight will be dynamically increased. For each subset, we build a corresponding validation set and use an off-the-shelf vision-language model to compute its per-sample PPL as the expected reference level [Xie et al., 2023, Michel et al., 2022]. To prevent high-value samples from being diluted by aggregate statistics, samples for which the PPL convergence is consistent with downstream performance will be isolated and relabeled into independent data subsets. With the original data partitions largely preserved while the high-value samples organized more finely, this approach significantly enhances data efficiency and downstream task performance.

### 3.2.5 Stage-4 Context-Length Extension Training

To augment the model's proficiency in capturing extended sequential relationships across various modalities, we gradually extend the context length to 32K and then 128K tokens, enabling the model to process and reason over more complex tasks such as long-term memory modeling and multi-turn real-time interaction. We scale the context length from 8K to 32K tokens using 100B training tokens and adjust RoPE's base frequency [Su et al., 2024] from 1M to 5M to preserve positional encoding quality. The context length is then further expanded to 128K tokens with an additional

20B tokens of training, requiring a proportional increase of the RoPE's base frequency to 10M to maintain stable attention across the extended sequence length.

A key challenge in long-sequence multimodal modeling lies in preserving fine-grained visual information across multi-image compositions or extended video sequences. To address this, integrating our native resolution encoding strategy achieves dual optimization: it maintains critical visual details at their original fidelity while intelligently managing token allocation efficiency. This approach ensures the preservation of essential spatial relationships in images and temporal coherence in videos throughout extended sequences. Building upon the stage-3 vision-related data, we further augment the training mixture by incorporating 25% additional long-context multimodal data, as detailed in Section 3.1.7.

For textual and speech modalities, we utilize the foundation language model's high-quality text corpus [Meituan, 2025a], alongside carefully curated speech data from our multilingual audio repository (over 10 million hours of processed Chinese/English recordings). Notably, we maintain the same 2:1:1 text-to-vision-to-speech ratio as established in the pre-training phase, ensuring modality balance during context extension. This comprehensive approach ensures consistent performance across all modalities while scaling to extended contexts.

### 3.2.6 Stage-5 Audio Encoder Alignment Training

In this stage, we maintain the LLM parameters in a frozen state while exclusively training the audio encoder. This approach serves dual objectives: (1) preserving the LLM's established multimodal processing capabilities, and (2) enhancing audio comprehension while aligning continuous speech inputs with the LLM's semantic space. We investigated the necessity of separate audio projector module pre-training for semantic space alignment, but empirical results showed negligible performance differences compared to direct end-to-end audio encoder training. To accelerate convergence, we initialize the audio encoder (excluding the projector) from the audio encoder trained with speech recognition introduced in Section 2.2, while randomly initializing the projector module parameters.

During training, inputs are formatted as follows: "Task Prompt + Speech Input + LLM Response", where the loss is computed exclusively on the "LLM Response" segment. To improve task generalization, we design multiple prompt variations for each task and randomly select prompts during training.

## 4　Post-Training

The post-training stage is a critical phase that transforms the pre-trained foundation model into a task-adaptive, human-aligned system with strong instruction-following, multimodal reasoning, and interactive capabilities. While the pre-training phase enhances multimodal perception and world knowledge, post-training focuses on refining response alignment, controllability, and fidelity through a combination of supervised and preference-based optimization. This stage consists of two components: (1) Supervised Fine-Tuning (SFT), which equips the model with multimodal instruction-following, reasoning, and spoken interaction capabilities through high-quality and diverse instruction data, and (2) Reinforcement Learning (RL), which further enhances the model's behavioral alignment, coherence, and consistency through Direct Preference Optimization (DPO) [Rafailov et al., 2023]. Together, these procedures ensure that the final model not only demonstrates robust omni-modal understanding and reasoning capabilities but also generates responses that are semantically accurate, perceptually natural, and contextually coherent across both offline and online interaction scenarios.

### 4.1　Supervised Fine-Tuning

The SFT stage focuses on two complementary objectives. First, it enhances the model's multimodal instruction-following and reasoning capabilities by leveraging large-scale, high-quality, and diverse instruction data. Second, it strengthens the model's multimodal interaction abilities by utilizing curated interaction data, including speech-to-speech conversational data and audio-visual interaction data. In the following parts, we describe how we collect high-quality and diverse instruction data, then elaborate on the construction of multimodal interaction datasets, and finally present the SFT training recipes that unify and optimize these datasets.

#### 4.1.1　High-quality and Diverse Instruction Data Curation

**Image-Text SFT Data** We curate a high-quality and diverse image-text SFT dataset, covering a wide spectrum of vision-language tasks, including: (1) Fundamental skills: image captioning and visual question answering, provided in both single-image and multi-image settings to capture diverse contextual dependencies. (2) Specialized skills: document and chart comprehension, OCR, visual grounding, agentic task execution, and STEM-related visual reasoning.

To ensure high data quality and strong alignment with human preferences, we employ an LLM-as-a-judge evaluation framework. In this pipeline, a strong multimodal language model evaluates candidate responses against reference answers and produces explanatory rationales. Samples labeled as inconsistent, low-quality, or semantically inaccurate are automatically filtered out, and we finally have approximately 3 million carefully curated datasets with high factual accuracy, stylistic consistency, and comprehensive task diversity.

**Video-Text SFT Data** We construct a large-scale video data pool of approximately 3 million videos sourced from diverse proprietary and open-source datasets, covering tasks such as general video understanding, classification, reasoning, grounding, temporal localization, segmentation, and highlight detection. From each source, representative subsets are annotated with capability tags using multimodal language models, following a predefined taxonomy encompassing visual perception, temporal and causal reasoning, and domain-specific knowledge.

This taxonomy, consisting of 48 subcategories (e.g., entity attributes, counting, relationship comparison, OCR, anomaly detection, spatial relations, and camera motion), guides targeted sampling to enrich underrepresented capabilities and analyze coverage gaps. For capabilities with insufficient data, we augment annotations using proprietary expert models, and for particularly challenging tasks—such as action counting, relationship comparison, event localization, and attribute change—we incorporate manually annotated samples.

Through iterative sampling, augmentation, and quality refinement, we curate a high-quality video-text SFT dataset comprising approximately 700K samples, with balanced coverage of temporal reasoning, causal inference, and complex real-world visual understanding.

**Audio Understanding Data** We re-utilize a subset of the comprehensive audio understanding dataset from the previous stage (Audio Encoder Alignment Training) for the SFT phase. This sampled data specifically targets tasks such as ASR, Audio-to-Speech Translation (AST), paralinguistic understanding, and audio-conditioned captioning and question answering. The purpose of integrating this kind of data is to improve the semantic alignment between the continuous audio representations generated by the audio encoder and the language model's semantic space.

**Vision-Speech QA Data** Fluent, speech-based question answering grounded in visual inputs represents a core capability of omni-modal models. We designed a new vision-speech QA dataset to enhance this ability. It pairs visual inputs (images or videos) with spoken prompts and requires the model to output its answers in speech. To create this data, we sourced text-based QA pairs from existing SFT datasets, selecting only those suitable for TTS synthesis. The retained samples are then rewritten using an LLM to enhance fluency, naturalness, and stylistic consistency in spoken form. Finally, all text responses are converted into high-fidelity speech using a TTS engine.

**Audio-Visual Understanding Data** The joint audio-visual understanding capability fundamentally distinguishes an omni model from conventional vision-only or speech-only multimodal language models. To develop this capability, we curate an in-house time-synchronized audio-visual dataset by prompting a strong multimodal language model to generate high-quality text QA pairs that are involved with both audio and visual content in the video. Specifically, the videos are sourced from multiple domains, including multimodal data containing rich music and audio event contents. All samples are organized in an interleaved chunk format, which enables the model to better perceive and reason over temporally aligned audio-visual content, thereby strengthening its capacity for multimodal understanding.

### 4.1.2 Multimodal Interaction Data Curation

To equip LongCat-Flash-Omni with real-time spoken and audio-visual interaction capabilities, we construct two types of specialized interaction datasets: (1) Speech-to-Speech Interaction Data, designed to enhance the naturalness and expressiveness of voice-based dialogues, and (2) Audio-Visual Interaction Data, aimed at improving the model's ability to manage multi-turn, context-dependent audio-visual conversations.

**Speech-to-Speech Interaction Data** We construct a large-scale voice dialogue dataset using a two-stage approach. (1) Text-based dataset adaptation: we filter out contents like formulas, code, markdown and rewrite responses with an LLM to produce speech-friendly conversational language. (2) Voice-oriented dialogue generation: we prompt an LLM to generate diverse topics and create new multi-turn dialogues. To enhance expressiveness and naturalness, professional voice actors record dialogues across a range of emotions, speaking styles, and major Chinese dialects (e.g., Sichuan, Beijing, Northeast). These recordings are then used to fine-tune a specialized TTS engine, ensuring consistent tone, high fidelity, and natural prosody. All dialogues are ultimately synthesized into high-quality speech using this engine.

**Audio-Visual Interaction Data** Audio-visual speech interaction data plays a crucial role in endowing the omni model with the ability to simultaneously process audio and vision information and perform real-time voice response. Unlike offline video understanding or general audio-visual understanding, online interaction is characterized by bi-directional and multi-turn dialogues, which involve immediate feedback, dynamic barge-in, contextual memorization, logical progression, and co-reference resolution. However, collecting large-scale real-world audio-visual interaction data is

resource-intensive and impractical, and synthesizing complex and reasonable interaction data is difficult for existing models that tend to generate hallucination. To address these challenges, we develop a semi-automated data production pipeline that leverages model-driven automation for initial generation, followed by a human-in-the-loop stage for verification and refinement:

- **Model-Driven Automation** To comprehensively cover diverse interaction scenarios, we first establish an ability taxonomy encompassing six major dimensions: memorization, understanding, analysis, creation, application, and entertainment. Guided by this taxonomy, we collect a mixture of public and in-house videos as the source data to ensure balanced coverage across abilities. For each video, we first perform scene segmentation using PySceneDetect [Breakthrough, 2025] to obtain a sequence of clips. For every clip, a powerful multimodal language model generates multiple rounds of progressively deep and context-aware QA pairs. In each round, subsequent queries build upon preceding conversations with logical progression, referential dependency, or anaphoric linkage to earlier turns. Such conversations encourage the model to reason over extended dialogue contexts, maintain entity and reference consistency in a natural, human-like audio-visual interaction setting. We then apply an automatic verification pipeline, employing an LLM-as-a-judge framework to evaluate and discard low-quality or inconsistent QA pairs. The remaining qualified samples are then composed into multi-turn dialogue, enabling the model to learn both intra-scene conversational continuity and inter-scene contextual transitions, thereby better reflecting real-world audio-visual interactions.

  To further enhance long-term memory and contextual reasoning in extended dialogue settings, we construct a long-context multi-turn video interaction dataset. In this dataset, dialogue sequences are reordered so that certain queries appear at temporal positions distant from their corresponding visual segments, encouraging the model to retain and retrieve information over extended temporal spans. After reordering, all QA sequences are refined with a strong multimodal language model to resolve anaphoric references, correct temporal inconsistencies, and improve contextual coherence, resulting in natural and logically consistent multi-turn conversations.

- **Human-in-the-loop** Through manual verification, we observe that the generated QA data frequently exhibits several types of errors: factual inconsistency, response insufficiency, referential ambiguity, linguistic infelicity, and semantic irrelevance. It is difficult to reuse a multimodal LLM to automatically correct these errors, as it may introduce new errors. Therefore, we employ human annotation for quality refinement. For factual inconsistency, annotators are required to edit or replace any statement that contradicts the video/audio content. For response insufficiency, we ask annotators to augment the original answer with the missing information, ensuring all sub-questions are answered. For referential ambiguity, we ask annotators to replace ambiguous or incorrect pronouns with specific nouns or corrected pronouns. As a result, we obtain a high-quality interaction dataset that contains meaningful and natural conversations that align with verifiable truth.

  Finally, we apply a TTS engine to convert the textual QA pairs into speech and integrate them into the videos to obtain the final audio-visual speech interaction dataset. The resulting data aligns the model's response style with that of a helpful, interactive assistant and enhances the naturalness of its conversational behavior.

### 4.1.3 SFT Training Recipe

During SFT, we freeze the audio encoder while updating all other modules. The frozen audio encoder retains robust acoustic representations learned from large-scale audio pre-training (i.e., pre-training stage-5 as introduced in Section 3.2.6), whereas the unfrozen layers allow the model to learn fine-grained multimodal alignment and instruction-following behavior. Empirically, we find this selective fine-tuning strategy stabilizes convergence and avoids catastrophic forgetting of low-level auditory features. Apart from multi-modal SFT data, we also incorporate pure text SFT data used in developing LongCat-Flash, which covers domains ranging from reasoning, mathematics, coding, tool use, and general-purpose dialogue.

We adopt the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.1. The learning rate follows a linear warm-up over the first 4% of steps, followed by cosine decay from a peak value of $1 \times 10^{-5}$ to zero. The SFT stage consists of a single training phase conducted for one epoch. The training is performed with a batch size of 1024.

## 4.2 Reinforcement Learning

To further enhance the model's multimodal capabilities and improve human alignment, we employ the Direct Preference Optimization (DPO) [Rafailov et al., 2023] during the reinforcement learning stage. The LongCat-Flash-Omni model supports multimodal inputs and enables parallel streaming outputs of both text and speech. However, most existing DPO variants are designed for text-only outputs or optimize text and speech separately. We argue that such decoupled optimization is suboptimal for maintaining coherence between textual and speech responses.

To address this limitation, we extend the DPO to jointly optimize text and speech outputs to improve both alignment and stability across modalities. Specifically, since the LongCat-Flash-Omni includes one text head and multiple audio heads, we modify the DPO objective to optimize all heads simultaneously. The overall loss is defined as:

$$\mathcal{L}_{\text{DPO}} = \alpha \, \mathcal{L}_{\text{DPO}}(\text{text}_{chosen}, \text{text}_{rejected}) + \beta \sum_{i=1}^{N} \mathcal{L}_{\text{DPO}}(\text{audio}^i_{chosen}, \text{audio}^i_{rejected})$$

where $N$ denotes the number of audio heads. $\mathcal{L}_{\text{DPO}}(\text{text}_{chosen}, \text{text}_{rejected})$ focuses on the semantic quality of the text head, while each audio head $\mathcal{L}_{\text{DPO}}(\text{audio}^i_{chosen}, \text{audio}^i_{rejected})$ emphasizes both the linguistic and pronunciation stability of the corresponding speech output. This joint optimization strategy leads to more coherent and natural multimodal responses, further enhancing the expressiveness and human-aligned communication capabilities.

### 4.2.1 Data Construction

Our DPO training data consists of two components: general DPO data and model-generated DPO data. The general data covers samples focusing on safety, helpfulness, and response style. The model-generated data is sampled from an SFT checkpoint and used to refine broader multimodal capabilities through preference comparisons among model-produced responses. To comprehensively enhance the capability, alignment, and safety of LongCat-Flash-Omni across all modalities, we utilize all prompt types from the SFT datasets described in Section 4.1.1 during DPO training. For each prompt, the model generates 6 rollouts to ensure diverse candidate responses for preference pair construction. To ensure data quality, we employ a hybrid evaluation strategy that combines manual annotation with automatic scoring from a robust multimodal language model, serving as a preference evaluator to assign quality scores and identify distinct and informative preference pairs. This process provides reliable supervisory signals for DPO training, enabling effective refinement of LongCat-Flash-Omni 's multimodal alignment, behavioral stability, and overall coherence.

### 4.2.2 Training Details

We train for one epoch with a batch size of 256, sampling each batch to achieve a well-balanced mix of modalities. The learning rate follows a cosine decay schedule, with a warm-up fraction of 0.03, gradually decreasing from $1 \times 10^{-6}$ to 0. To balance preference learning between the text head and the speech head, we set their loss weight ratio by setting $\alpha : \beta = 1 : 1$. To mitigate drift from the SFT model, we incorporate a Kullback-Leibler (KL) divergence regularizer with a weighting factor of 0.1.

## 5 Training Infrastructures

Our core design principles are largely inspired by the training infrastructure used in developing LongCat-Flash, with a particular emphasis on maximizing training efficiency while strictly ensuring numerical consistency. To guarantee numerical consistency, we enforce determinism, minimize errors, and maintain error interpretability, ensuring that every training run is both deterministic and reproducible. For efficiency, we decouple the components of the LLM, vision encoder, and audio encoder, enabling independent optimization of their performance and memory usage. Experimental results demonstrate the effectiveness of our approach: in multimodal settings, our system maintains over 90% of the throughput of text-only training.

### 5.1 Multimodal Decoupling Framework

In multimodal scenarios, optimizing training performance is challenging due to the heterogeneity of both data and models. The challenge of data heterogeneity arises because the training data for LongCat-Flash-Omni includes speech, vision, and text, which show significant and dynamic differences in token distribution (Figure 7). Model heterogeneity is evident in the substantially different computational workloads across the three core components of LongCat-Flash-Omni: the vision encoder, the audio encoder, and the LLM decoder (Table 2). To address the aforementioned challenges, there are two primary optimization approaches. The first relies on Fully Sharded Data Parallelism (FSDP) (e.g., OrchMLLM [Zheng et al., 2025], veOmni [Ma et al., 2025]), which reduces static memory consumption through parameter sharding, avoids pipeline parallelism (PP) bubbles caused by model heterogeneity, and mitigates data heterogeneity via data-parallel (DP) group balancing. However, for LongCat-Flash-Omni, the total number of model parameters is too large for this approach to be practical.

The second approach is to decouple multimodal models across different parallel dimensions. For example, Dist-Train [Zhang et al., 2025] mitigates model and data heterogeneity by separating model partitioning strategies and

Table 2: Computation distribution per micro-batch across different modalities during the SFT stage.

| Module | Computational Cost (TFLOPs) | | | |
| --- | --- | --- | --- | --- |
| | min | max | mean | std |
| Audio Encoder | 0.01 | 109.96 | 3.29 | 7.94 |
| Vision Encoder | 0.08 | 400.37 | 89.85 | 61.02 |
| LLM Decoder | 1920.57 | 4667.74 | 3531.32 | 1111.11 |
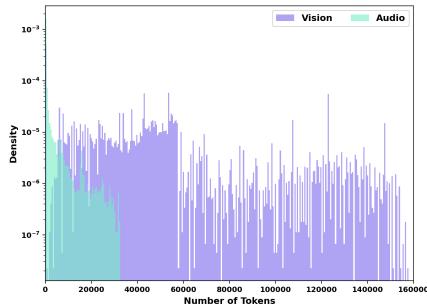


Figure 7: The distribution of sequence token lengths across different modalities during the SFT stage.

sorting data; PipeWeaver [Xue et al., 2025] reduces PP bubbles caused by heterogeneity through fine-grained data partitioning and dynamic pipeline scheduling; and Optimus [Feng et al., 2025] separates the parallelization strategies for encoders and LLMs, scheduling encoder computations into the LLM's idle periods to eliminate bubbles caused by model heterogeneity. Building on the Optimus approach, we develop modality-decoupled parallelism (MDP), a simple yet effective multimodal training strategy. The core idea is to completely decouple the modality encoders and the LLM backbone at the distributed level, enabling independent scheduling and more efficient utilization of computational resources.

### 5.1.1 Modality-Decoupled Parallelism

In our implementation, we co-locate the modality encoders and the LLM decoder. The modality encoders utilize Hybrid Sharding Data Parallelism (HSDP)[Zhao et al., 2023] to reduce static memory, and full activation recomputation is employed to reduce activation memory usage. The LLM decoder adopts a combined distributed strategy including pipeline parallelism (PP), ZeRO-1 data parallelism (DP), context parallelism (CP), and expert parallelism (EP). To simplify data mapping between modality encoders and the LLM decoder, we introduce an *InnerDP* parallelism strategy, which further partitions modality data across all microbatches. The DP rank of modality encoders corresponds one-to-one with the DP rank of the LLM decoder, and the size of the InnerDP dimension is the product of the PP and CP of the LLM decoder (i.e., $d_{inner\_dp} = d_{lm\_cp} \times d_{lm\_pp}$, $d_{world\_size} = d_{dp} \times d_{inner\_dp}$). As shown in Figure 8, the MDP execution timeline consists of four phases:

**Data Loading**: At the start of each training iteration, the rank with $inner\_dp = 0$ fetches all micro-batches. Then it broadcasts the metadata of the data to the other ranks within the DP group. This mitigates I/O overhead and memory pressure. In this phase, all micro-batches are sorted by text data sequence length to ensure workload across DP groups remain similar, thereby reducing EP bubbles caused by workload imbalances.

**Modality Encoder Forward**: The BalanceData module first distributes the modality data from the $inner\_dp = 0$ rank to the other $inner\_dp$ ranks. Then, the modality encoders on each rank compute the corresponding vision and audio embeddings. Finally, the ModalityBridge module aggregates these embeddings at the $inner\_dp = 0$ rank, which are then passed as input to the LLM decoder.

**LLM Decoder Forward and Backward**: In this phase, modality embeddings are partitioned on the CP rank and fed into the LLM decoder for its forward and backward passes. The gradients of the vision and audio embeddings are then returned to the backward phase for the modality encoders. This design ensures the training efficiency of the LLM decoder, and also enables isolated optimization for modality encoders.

**Modality Encoder Backward**: The ModalityBridge module redistributes the modality embedding gradients from the LLM decoder to the other $inner\_dp$ ranks, and then the backward pass of the modality encoders is executed.

### 5.1.2 ModalityBridge

In MDP, the ModalityBridge serves as the communication layer between the multimodal encoders and the LLM decoder. It is responsible for transforming data organization formats to resolve discrepancies arising from the different parallelism strategies employed by the modality encoders and the LLM decoder. This process specifically involves handling modality embeddings during the forward phase and gradients during the backward phase. However, when
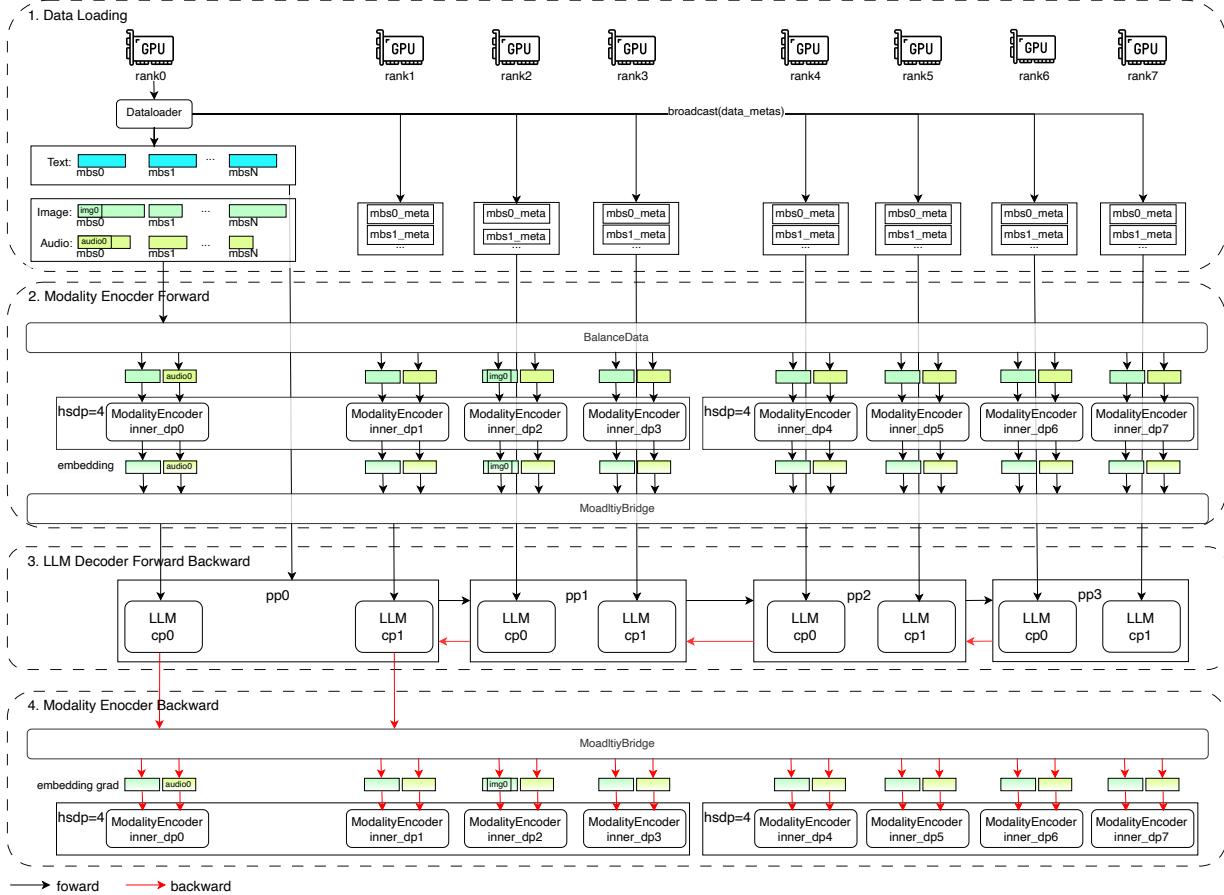
Figure 8: **Overview of modality-decoupled parallelism (MDP).** The modality encoders and the LLM backbone are fully decoupled at the distributed level, enabling independent scheduling and improved computational efficiency.

processing longer context length, significant memory pressure emerges as the $inner\_dp = 0$ rank must read all micro-batches and perform gather and scatter operations on the modality embeddings and their gradients. To address this challenge, we adopt chunk-based processing within ModalityBridge, which effectively alleviates the memory bottleneck while maintaining bitwise numerical consistency. As illustrated in Figure 9, this module comprises three core components:

**Embedding Redistribution**: Employs a two-stage chunking approach for data gather and scatter, decomposing the complete data transformation into $num\_chunk$ iterations. Each iteration consists of two stages:

1. **Aggregation:** Inner-DP embeddings are aggregated to the $inner\_dp = 0$ rank.
2. **CP partitioning:** The aggregated data is split along the $hidden\_size$ dimension to ensure uniform embedding distribution and balanced memory usage, followed by scattering to the respective CP ranks to further reduce per-device memory consumption.

During each chunking iteration, the overall memory footprint exhibits a rise-then-fall pattern, significantly lowering the peak memory usage.

**Modality Embedding Storage**: Stores all gathered-then-scattered chunk data while maintaining global offset information for subsequent indexing.

**Embedding Indexing**: Provides micro-batch-level embedding retrieval and gradient backpropagation during the LLM forward and backward phases.

Based on these three components, we accomplish the transformation of data formats across different stages. In the forward phase, the transformation is performed according to the two-stage chunking approach, while in the backward phase, *Embedding Redistribution* executes the reverse process: gradients are first gathered via CP gather and then
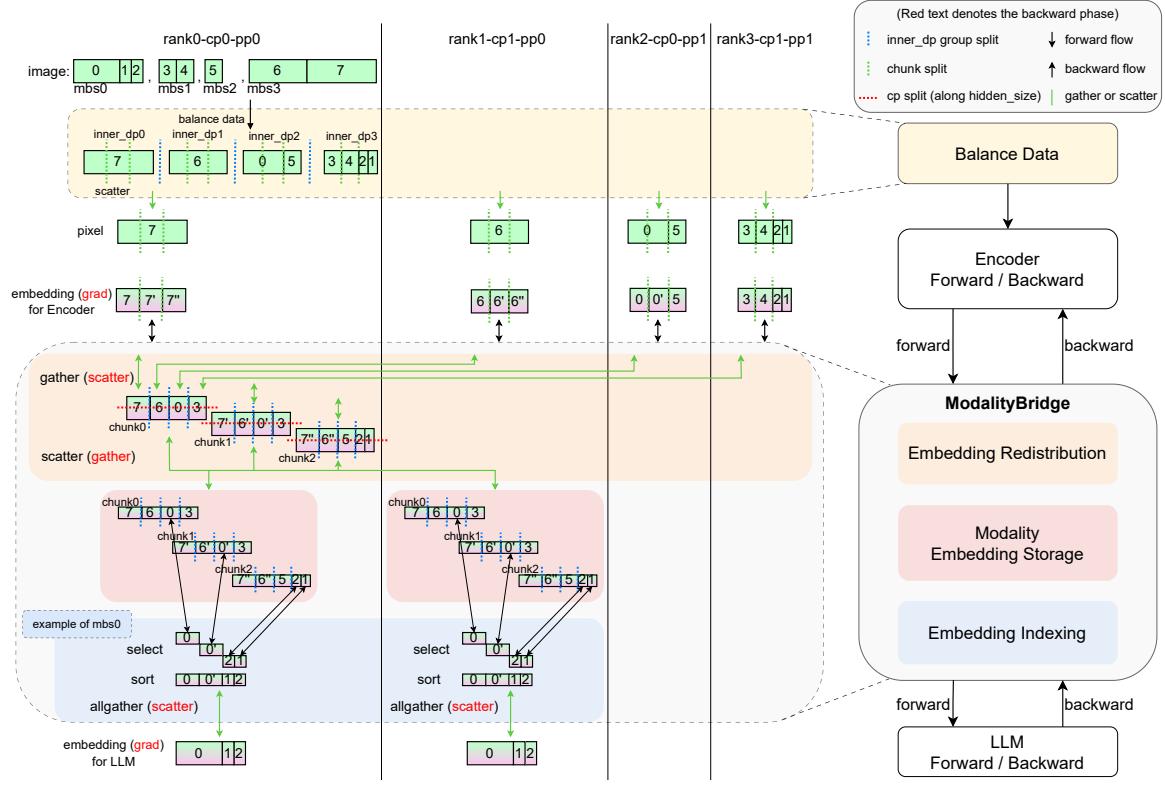
Figure 9: **Overview of the chunk-based ModalityBridge**. The example configuration with num_chunk = 3, processing 4 microbatches of 8 images each across 4 GPUs (DP = 1, CP = 2, PP = 2)

scattered to the corresponding $inner\_dp$ ranks according to the global offset information for backward computation. Meanwhile, the chunking scheme reduces the peak memory usage to $1/num\_chunk$ of the original, significantly alleviating memory pressure while ensuring bitwise numerical alignment.

## 5.2 Performance Tuning

Our overarching strategy is twofold: first, select a distributed configuration by profiling core operator efficiency; second, meet the configuration's memory budget via targeted memory optimizations. On the communication side, the shortcut architecture enables overlap between EP communication and computation within each micro-batch. In CPU-bound regimes, we employ fused operators to reduce kernel launch overhead and improve end-to-end throughput.

### 5.2.1 Optimal Distributed Configuration

In large-scale LLM pretraining, the efficiency of core operators is primarily constrained by the sequence length; increasing sequence length typically improves Model FLOPs Utilization (MFU). Accordingly, our system design increases the effective sequence length per EP rank to approach the operators' high-efficiency regime. Hardware-specific benchmarking results further indicate that reducing CP substantially improves the efficiency of the core operator, as shown in Figure 10.

### 5.2.2 Communication Optimizations

Our training infra contains multiple compute-communication overlaps, mainly including shortcut-based EP overlap and point-to-point (P2P) overlap. We tune the number of streaming multiprocessors (SMs) assigned to communication and compute kernels for each scenario to maximize hardware utilization, and we use distinct P2P groups/streams so that inter-stage pipeline-parallel (PP) communication does not interfere across stages.
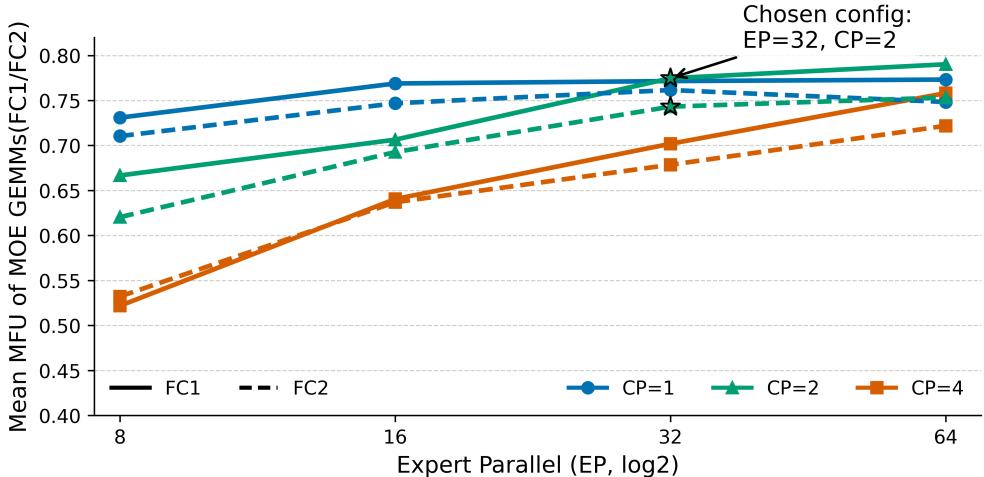
Figure 10: Benchmark results of MoE GEMMs under different CP/EP configurations with an 8K sequence length.

### 5.2.3 Kernel Optimizations

To optimize the training efficiency of MoE models, we maintain computational determinism while implementing kernel optimization and fusion along critical computational paths. Our technical contributions encompass the following areas:

**Optimized Grouped GEMM** Our optimization strategies for Grouped GEMM achieve approximately 75% MFU in the evaluated scenarios. (1) **Dynamic SwapAB:** The WGMMA instruction is evaluated with the $N$ dimension varying from 8 to 256, while the $M$ dimension is fixed at 64. We design an optimized kernel that automatically performs SwapAB operations to maximize WGMMA performance. (2) **Configurable SM Usage:** The Grouped GEMM kernel allows configurable SM counts to avoid resource contention when overlapping with dispatch or combined communication operations. (3) **Fine-Tuning:** *TileShape*, *PipelineStage*, and *ClusterShape* are carefully tuned for the target workload to maximize hardware utilization and prevent register spilling. (4) **Scheduling and Pipeline Strategy:** A swizzled block mapping schedule is adopted to significantly improve L2 cache hit rates. In addition, a ping-pong mechanism is introduced during the backward pass to overlap WGMMA computation with the loading and storing of weight gradients.

**Fused Permute** An efficient fused permute kernel is employed to rearrange tokens for expert alignment, while also integrating several critical functionalities, including metadata computation for the backward pass and token dropping within the CP group.

**Fused RoPE** We fuse RoPE into the MLA prologue kernel, eliminating the overhead of intermediate data writing and reloading, achieving a $3\times$ speedup compared to the baseline.

**Deterministic FA** We employ a semaphore-based synchronization scheme that ensures deterministic QK reduction in the backward phase and outperforms the default deterministic implementation, which requires an additional temporary workspace. This approach achieves approximately $0.8\times$ the performance of the non-deterministic version.

### 5.3 Memory Optimization Strategies

Without optimization, the required memory for our target configuration is approximately 137 GB per device (Table 3). Given 80 GB devices and accounting for EP imbalance peaks, we constrain the theoretical memory footprint to approximately 72 GB.

We combine the techniques in Table 4: (i) a V-shaped PP schedule to bound concurrently alive activation micro-batches [Qi et al., 2024]; (ii) selective recomputation for low-FLOP, high-activation operators such as SwiGLU and LayerNorm [DeepSeek-AI et al., 2025]; (iii) memory-efficient permute, which moves aggregation of routing probabilities and hidden states into SwiGLU to reduce MoE unpermute memory [Shoeybi et al., 2019]; (iv) fine-grained SM budgeting for NCCL communicators to reduce NCCL memory; and (v) Hybrid-Sharding Data Parallel (HSDP) for the modality encoder to further reduce its static footprint [Paszke et al., 2019]. To accommodate potential EP load imbalance, we implement *dynamic expert recomputation*: when any rank is assigned too many tokens, we dynamically recompute its down-projection, freeing all MoE memory except the down-projection inputs on trigger. This prevents training crashes from expert imbalance while incurring only modest overhead.

Table 3: Naïve memory usage breakdown by component.

| Component | Static (ZeRO-1, GB) | Dynamic (GB) |
|---|---|---|
| LLM | 16.8 | 103.24 |
| NCCL | – | 12.93 |
| DeepEP | – | 0.45 |
| Modality Encoders | 3.6 | – |
| **Total** | **137.02 GB** (peak at non-pp0 stage) | |

Table 4: Memory footprint under different optimization settings.

| Setting | Static | Dynamic(GB) | Others(GB) | Total(GB) |
|---|---|---|---|---|
| Baseline | 20.4 | 103.2 | 13.4 | 137.0 |
| +HSDP for Modality Encoder | 17.5 | 103.2 | 13.4 | 134.1 |
| +Vhalf | 17.5 | 58.6 | 13.4 | 89.4 |
| +Select Recompute | 17.5 | 49.2 | 13.4 | 80.1 |
| +Memory Efficient Permute | 17.5 | 42.8 | 13.4 | 73.6 |
| +NCCL Memory Optimization | 17.5 | 42.8 | 8.9 | **69.1** |

## 5.4 Numerical Consistency

Large language model pretraining is extremely resource-intensive, often requiring weeks and incurring millions in computational costs. Ensuring the correctness of the training framework is essential to avoid costly retraining due to implementation errors.

In our framework, we enforce deterministic implementations for all computation, communication, and data flow, ensuring that every experiment is reproducible and maintains bitwise-aligned loss values.

For new features, we prioritize bit-aligned implementations to ensure numerical consistency. For example, features like `fused_permute` and `fused_swiglu` ensure bitwise-aligned loss whether they are enabled or not. Similarly, kernels like Grouped GEMM and Router GEMM employ a non Split-K implementation to preserve alignment across varying batch sizes or distributed settings.

For features where bit alignment is not feasible, we systematically analyze all sources of numerical deviation and mitigate their impact by benchmarking against the gold reference implementation during verification. For example, we identified and aligned the accumulation order in DeepEP and All2All EP strategies, thereby achieving bit-level consistency in loss values and verifying implementation correctness.

## 6 Inference and Deployment

### 6.1 Decoupled Framework

We propose a decoupled multimodal inference framework that separates modality-specific encoders/decoders and the LLM for optimized deployment. Each module is deployed on dedicated hardware and accelerators tailored to its computational characteristics, mitigating cross-modal resource contention. The optimizations of LLM deployment follow LongCat-Flash, including Prefill-Decode (PD) Disaggregation and Single Batch Overlap for ScMoE to improve inference efficiency. Compared with conventional hybrid deployment, this separation achieves lower latency and higher throughput despite minor communication overhead.

### 6.2 Asynchronous Streaming Pipeline

We design a highly efficient asynchronous streaming pipeline for model serving. As shown in Figure 11, it consists of four sequentially linked and concurrently executed stages: VAD & Frame Sampling, Audio-Visual Encoding, LLM Prefilling & Decoding, and Audio Decoding. Each module supports incremental inference of streaming input and adaptive batching strategies, facilitating concurrent scheduling to reduce latency.

**Sparse-Dense Sampling Strategy** A voice activity detection (VAD) module is deployed to detect whether the user is speaking in real-time. When the user is speaking, densely sampled frames as well as the audio will be used as model
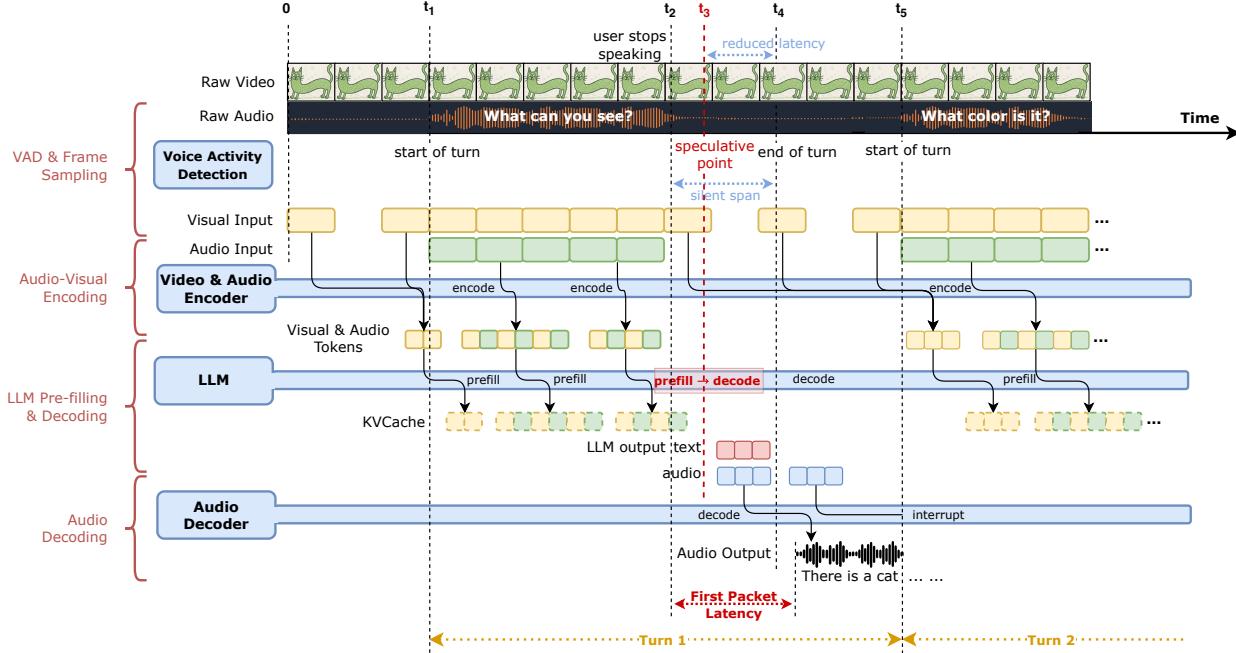
Figure 11: The asynchronous streaming pipeline for minimizing first packet latency.

input, facilitating a deeper analysis of the video content. Otherwise, only sparsely sampled frames are used to track the high-level overview of the video stream.

**Speculative Prefill-Decode Switching** Typically, the VAD model determines the end of a user turn by assessing whether the silence is longer than a predefined duration, referred to as the *silent span* $[t_2, t_4]$. However, deferring the LLM state transition from prefill to decode until $t_4$ would incur substantial latency to the first packet. To overlap this latency with the latency of LLM decoding, we adopt a speculative prefill-decode switching strategy: The LLM starts decoding at an earlier speculative point $t_3$, reducing the latency from $t_3$ to $t_4$. If the user resumes speaking subsequently, a *rollback* is triggered, wherein the LLM's generated content is discarded and the system reverts to the prefill state.

**Audio Delivery and Interruption** The audio will be delivered to the user once the VAD model explicitly detects the end of a turn, which is marked as $t_4$ in the figure, even if the first packet of audio response is generated earlier. However, should a user interrupt the audio generation at $t_5$, the process is immediately terminated. Consequently, the token sequence from the LLM is truncated at the nearest natural breakpoint, such as the most recent punctuation mark.

In our implementation, each request data packet from the user side consists of 1 second of audio and 2 corresponding video frames. Using the streaming prefill strategy, each request packet can be immediately prefilled, eliminating the need to wait until the user's turn to finish before initiating computation. By overlapping the VAD endpoint detection $(600 \sim 700ms)$ and the prefill process, users can receive the model response within $100ms$ after the endpoint detection. This asynchronous pipeline enables real-time omni-modal interaction.

# 7  Evaluation

In this section, we conduct comprehensive evaluations of LongCat-Flash-Omni, compare it with various proprietary and open-source multimodal models, including vision understanding, audio comprehension, text understanding and generation, cross-modality understanding, and audio-visual interaction.

Table 5: Evaluation of image understanding. Values marked with * are sourced from public reports. As GPT-4o does not support image grounding, we do not report its results on RefCOCO and ScreenSpot-v2.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | Gemini-2.5-Flash non-thinking | Qwen3-Omni Instruct | Seed-1.6 | GPT-4o 1120 | Qwen3-VL 235B-A22B-Instruct | Qwen2.5-VL 72B-Instruct |
|---|---|---|---|---|---|---|---|---|
| **General** | | | | | | | | |
| MMBench-EN$_{test}$ | 87.5 | 89.8 | 89.3 | 86.8 | 88.5 | 83.7 | 88.3 | 88.6* |
| MMBench-ZH$_{test}$ | 88.7 | 89.2 | 88.5 | 86.4 | 83.8 | 82.8 | 89.8 | 87.9* |
| RealWorldQA | 74.8 | 76.0 | 73.9 | 72.9 | 74.5 | 74.1 | 79.3* | 75.7* |
| MMStar | 70.9 | 78.5* | 75.5 | 68.5* | 71.5 | 63.2 | 78.4* | 68.2 |
| **STEM & Reasoning** | | | | | | | | |
| MathVista$_{mini}$ | 77.9 | 77.7* | 77.1 | 75.9 | 78.7 | 62.8 | 84.9* | 74.8* |
| MMMU$_{val}$ | 70.7 | 80.9* | 76.3 | 69.1* | 74.9 | 69.4 | 78.7* | 70.2* |
| MMVet | 69.0 | 80.7 | 79.5 | 68.9 | 74.4 | 76.6 | 75.9 | 74.5 |
| **Multi-Image** | | | | | | | | |
| BLINK | 63.1 | 70.0* | 65.7 | 56.1 | 65.0 | 65.5 | 70.7* | 60.1 |
| MuirBench | 77.1 | 74.0* | 73.7 | 62.1 | 74.6 | 70.5 | 72.8* | 70.7* |
| Mantis | 84.8 | 83.9 | 83.4 | 80.7 | 81.1 | 79.3 | 79.7 | 82.0 |
| **Text Recognition & Chart/Document Understanding** | | | | | | | | |
| ChartQA | 87.6 | 71.7 | 77.6 | 86.8* | 82.4 | 74.5 | 89.2 | 89.5* |
| DocVQA | 91.8 | 94.0* | 93.6* | 95.7 | 94.3 | 80.9 | 94.6 | 96.4* |
| OCRBench | 84.9 | 87.2* | 85.6 | 85.6 | 85.6 | 82.3 | 91.2 | 88.5 |
| OmniDocBench$_{EN/ZH}\downarrow$ | 22.8/29.0 | 31.9/24.5 | 22.8/32.9 | 28.4/40.5 | 22.0/27.6 | 25.9/37.7 | 13.6/17.5 | 22.6/32.4* |
| **Grounding & Counting** | | | | | | | | |
| RefCOCO-avg | 93.9 | 80.2 | 74.8 | 91.6 | 87.5 | - | 88.2 | 92.3 |
| CountBench | 92.4 | 91.0* | 78.6 | 90.0* | 94.1 | 85.6* | 94.3 | 93.6* |
| **Graphical User Interface (GUI)** | | | | | | | | |
| VisualWebBench | 78.7 | 81.1 | 73.5 | 79.3 | 81.1 | 77.1 | 80.8 | 82.3* |
| ScreenSpot-v2 | 91.2 | 75.8 | 63.9 | 94.7 | 91.7 | - | 93.4 | 92.9 |
| AndroidControl | 91.2 | 79.2 | 79.1 | 90.5 | 84.6 | 65.2 | 90.0 | 93.7* |

## 7.1 Vision Capability Evaluation

In this section, we compare LongCat-Flash-Omni with omni-models (Gemini-2.5-Pro[Comanici et al., 2025], GPT-4o[1], Seed-1.6[2], and Qwen3-Omni[Xu et al., 2025]) and vision language models (Qwen3-VL[Yang et al., 2025] and Qwen2.5-VL-72B[Bai et al., 2025]) without thinking mode.

Note that GPT-4o and Seed-1.6 only provide a vision understanding API. Therefore, we evaluate them solely on visual capabilities. To ensure a fair comparison with instruct model, we follow the evaluation setting used in Qwen3-VL benchmark, limiting Gemini-2.5-Pro's thinking budget to 128 tokens. All models are evaluated under their official configurations.

### 7.1.1 Image-to-Text Evaluation

We conduct a comprehensive evaluation of image understanding across six key dimensions, leveraging a diverse suite of benchmarks:

- **General Domains**: MMBench [Liu et al., 2024b], RealWorldQA [xAI, 2023], and MMStar [Chen et al., 2024b].
- **STEM & Reasoning**: MathVista [Lu et al., 2024], MMMU$_{val}$ [Yue et al., 2024], and MMVet [Yu et al., 2024].
- **Multi-Image**: BLINK [Fu et al., 2024a], MuirBench [Wang et al., 2024a], and Mantis [Jiang et al., 2024].
- **Text Recognition & Document Understanding**: ChartQA [Masry et al., 2022], DocVQA [Mathew et al., 2021], and OCRBench [Liu et al., 2024c], OmniDocBench$_{EN/ZH}$ [Ouyang et al., 2024].
- **Grounding & Counting**: RefCOCO, RefCOCO+ [Kazemzadeh et al., 2014] (averaged in RefCOCO-avg), and CountBench[Paiss et al., 2023].
- **Graphical User Interface (GUI)**: VisualWebBench [Liu et al., 2024d], ScreenSpot-v2 [Wu et al., 2024a], and AndroidControl [Li et al., 2024a].

Results on image understanding benchmarks are summarized in Table 5. Overall, LongCat-Flash-Omni achieves performance comparable to Gemini-2.5-Flash, while outperforming the open-sourced Qwen3-Omni. Its advantage is particularly pronounced on multi-image tasks, which benefit from the model's training on high-quality interleaved image-text, multi-image data and video datasets.

[1] https://openai.com/index/hello-gpt-4o
[2] https://seed.bytedance.com/en/seed1_6

### 7.1.2 Video-to-Text Evaluation

To assess the capability of video understanding, we consider three dimensions as below:

- **Short Video**: MVBench [Li et al., 2024b], NextQA [Xiao et al., 2021], and TempCompass [Liu et al., 2024e].
- **Long Video**: VideoMME [Fu et al., 2024b] and LongVideoBench [Wu et al., 2024b].
- **STEM & Reasoning**: MMVU [Zhao et al., 2025b], and Video-MMMU [Hu et al., 2025].

We evaluate audio-visual understanding on VideoMME under the no-subtitle protocol (w/o sub), and report results for two conditions: with audio ("VideoMME w/ audio") and without audio ("VideoMME w/o audio").

As the Gemini models do not provide official evaluation entries for videos without audio, we report only the "VideoMME w/ audio" results. All video benchmarks are evaluated using a maximum context length of 32K tokens to ensure consistent comparison.

As shown in Table 6, LongCat-Flash-Omni achieves state-of-the-art performance on video-to-text tasks. Specifically, it surpasses all compared models by a significant margin on short video understanding, demonstrating superior video comprehension capabilities. On long video tasks, LongCat-Flash-Omni demonstrates performance on par with leading models such as Gemini-2.5-Pro and Qwen3-VL. Notably, in the VideoMME benchmark, it achieves the best performance among omni-modal models. This can be attributed to a combination of an advanced video processing strategy—using dynamic frame sampling and hierarchical token aggregation—and the strong long-context modeling capacity afforded by its efficient backbone.

Table 6: Evaluation of video understanding. Values marked with * are sourced from public reports.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | Gemini-2.5-Flash non-thinking | Qwen3-Omni Instruct | Seed-1.6 | GPT-4o 1120 | Qwen3-VL 235B-A22B-Instruct | Qwen2.5-VL 72B-Instruct |
|---|---|---|---|---|---|---|---|---|
| **Short Video** | | | | | | | | |
| MVBench | 75.2 | 66.4 | 63.0 | 69.3* | 68.4 | 62.1 | 71.3 | 70.4* |
| NextQA | 86.2 | 84.2 | 81.4 | 82.4 | 84.1 | 79.7 | 81.3 | 82.3 |
| TempCompass | 82.2 | 80.8 | 80.2 | 73.5 | 79.4 | 76.4 | 80.5 | 74.8* |
| **Long Video** | | | | | | | | |
| VideoMME w/o audio | 76.2 | - | - | 70.5* | 75.2 | 73.2 | 79.2* | 73.3* |
| VideoMME w/ audio | 78.2 | 80.6* | 78.5 | 73.0 | - | - | - | - |
| LongVideoBench | 69.3 | 69.4 | 66.4 | 65.4 | 64.8 | 63.9 | - | 60.7* |
| **STEM & Reasoning** | | | | | | | | |
| MMVU | 67.1 | 75.6 | 72.4 | 62.4 | 67.3 | 67.4 | 69.3 | 62.9* |
| Video-MMMU | 67.5 | 79.4* | 76.6 | 60.3 | 75.4 | 68.0 | 73.7 | 59.3 |

## 7.2 Audio Capability Evaluation

### 7.2.1 Base Model Evaluation

To systematically assess the audio capability of the base models produced in the pre-training stages (i.e., stage-1, stage-2, stage-3, and stage-4), we conduct evaluations with respect to the following three aspects: automatic speech recognition (ASR), text-to-speech (TTS), and speech continuation. For ASR evaluation, we employ the SPEECHIO_ASR_ZH00002 (speechio02) [3] to test Chinese speech recognition, while utilizing the LibriSpeech [Panayotov et al., 2015] for English speech recognition assessment. The evaluation results are presented in Table 7. Remarkably, our model demonstrates competitive performance across these benchmark datasets, despite relying on discretized audio features.

The TTS evaluation framework uses text samples from the speechio02 and LibriSpeech [Panayotov et al., 2015] test-clean/other datasets as input sources. The model uses discrete semantic-acoustic tokens as speech prompts, conducting speech token generation with text input as teacher-forcing guidance. The output speech tokens are reconstructed into waveform by the audio decoder. We compute the word error rate (WER) (for English) or character error rate (CER) (for Chinese) between transcription of the generated speech produced by a robust ASR system and the original input text as the measure for the evaluation. The results are shown in the right part of Table 7. We observe that the base models are capable of generating robust speech from textual input, providing a solid foundation for speech output in spoken and audio-visual interaction modes.

To evaluate speech continuation capabilities, we develop a specialized test set comprising 1,200 synthesized speech samples derived from text data from the CMMLU [Li et al., 2023] benchmark. The assessment requires the model

---

[3] https://github.com/SpeechColab/Leaderboard

Table 7: ASR and TTS performance of base models during the pre-training stages. Word error rate (WER) (for English) or character error rate (CER) (for Chinese) in percentage are reported.

| Base Model | ASR | | | TTS | |
|---|---|---|---|---|---|
| | SpeechIO02↓ | LibriSpeech test-clean↓ | LibriSpeech test-other↓ | SpeechIO02↓ | LibriSpeech↓ |
| Stage-1 | 2.93 | 1.98 | 3.96 | 4.12 | 4.72 |
| Stage-2 | 3.18 | 2.11 | 4.59 | 2.16 | 8.64 |
| Stage-3 | 3.01 | 1.93 | 3.74 | 2.53 | 3.68 |
| Stage-4 (32K) | 3.49 | 2.30 | 4.00 | 1.73 | 5.99 |
| Stage-4 (128K) | 3.46 | 2.12 | 4.15 | 2.62 | 7.38 |

to generate appropriate responses to multiple-choice questions based on provided speech input. For text outputs, we directly measure response accuracy, while speech outputs undergo ASR transcription before accuracy evaluation. The evaluation protocol employs a 1-shot learning approach, with detailed results presented in Table 8. Our analysis reveals that the base models in different stage perform well in in-context speech continuation evaluation and there is only a negligible performance disparity between text and speech output.

Table 8: Speech continuation performance of base models across pre-training stages. Accuracy is reported as a percentage.

| Base Model | Speech Continuation (CMMLU)↑ | |
|---|---|---|
| | Audio In Text Out | Audio In Audio Out |
| Stage-1 | 88.80 | 84.80 |
| Stage-2 | 89.60 | 84.80 |
| Stage-3 | 92.80 | 92.00 |
| Stage-4 (32K) | 91.20 | 91.20 |
| Stage-4 (128K) | 90.40 | 90.40 |

### 7.2.2　Instruct Model Evaluation

We conduct comprehensive evaluation of the audio capabilities of LongCat-Flash-Omni, covering speech recognition and translation, audio understanding and audio-to-text chat.

**Speech Recognition and Translation**

For ASR evaluation, as presented in Table 9, we use LibriSpeech [Panayotov et al., 2015], AISHELL-1 [Bu et al., 2017], AISHELL-2 [Du et al., 2018], FLEURS [Conneau et al., 2023], CommonVoice15 [Ardila et al., 2019], and WenetSpeech [Zhang et al., 2022] as benchmarks. We observe that LongCat-Flash-Omni consistently achieves superior performance compared to other competitors, including Gemini-2.5-Pro [Comanici et al., 2025], GPT-4o-Audio, Qwen3-Omni-Instruct [Xu et al., 2025], Kimi-Audio [Ding et al., 2025], and Step-Audio-2-mini [Wu et al., 2025]. For speech-to-text translation (S2TT) evaluation, we adopt CoVost2 [Wang et al., 2021]. As shown in Table 9, LongCat-Flash-Omni exhibits strong S2TT capabilities among all models[4]. Taken together, the speech recognition and translation results demonstrate that LongCat-Flash-Omni possesses robust and comprehensive fundamental speech understanding capabilities.

**Audio Understanding** As an omni-modal model, LongCat-Flash-Omni can effectively function as a native audio understanding model when vision input is not provided. We conduct a comprehensive evaluation of LongCat-Flash-Omni across a diverse set of audio comprehension tasks, including music, sound events, and speech understanding. Specifically, we use MMAU [Sakshi et al., 2024], VocalSound [Gong et al., 2022], TUT2017 [Mesaros et al., 2016], ClothoAQA [Lipping et al., 2022], Nonspeech7k [Rashid et al., 2023], CochlScene [Jeong and Park, 2022], and MELD [Poria et al., 2018] as benchmarks. The results, presented in Table 10, show that LongCat-Flash-Omni consistently outperforms most competing models across all evaluated dimensions, and achieves state-of-the-art performance in several benchmarks. These findings highlight that LongCat-Flash-Omni possesses advanced capabilities in understanding a wide spectrum of general and complex acoustic information, going well beyond conventional speech recognition.

---

[4]Kimi-audio defaults to ASR rather than executing the requested translation task.

Table 9: Automatic speech recognition (ASR) and speech-to-text translation (S2TT) evaluation results.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | GPT-4o-Audio | Qwen3-Omni Instruct | Kimi-Audio | Step-Audio-2-mini |
|---|---|---|---|---|---|---|
| **ASR** | | | | | | |
| **LibriSpeech** test-clean \| test-other | 1.57 \| 4.01 | 1.74 \| 3.80 | 30.00 \| 41.83 | 1.22 \| 2.48 | 1.28 \| 2.42 | 1.33 \| 2.86 |
| **AISHELL-1** | 0.63 | 3.11 | 34.81 | 0.84 | 0.60 | 0.78 |
| **AISHELL-2** | 2.78 | 5.24 | 77.73 | 2.34 | 2.56 | 2.16 |
| **Fleurs** zh \| en | 3.99 \| 5.02 | 2.24 \| 4.77 | 3.91 \| 5.56 | 2.20 \| 2.72 | 2.69 \| 4.44 | 2.53 \| 3.05 |
| **CommonVoice 15** zh \| en | 4.98 \| 13.59 | 47.30 \| 49.86 | 42.83 \| 23.88 | 4.31 \| 6.05 | 8.46 \| 7.92 | 5.00 \| 6.75 |
| **WenetSpeech** test-meeting \| test-net | 6.69 \| 6.09 | 136.13 \| 32.82 | 54.35 \| 67.90 | 5.89 \| 4.69 | 6.28 \| 5.37 | 4.87 \| 4.82 |
| **S2TT (BLEU)** | | | | | | |
| **CoVost2 en→zh** | 47.23 | 41.94 | 29.32 | 48.72 | - | 49.12 |
| **CoVost2 zh→en** | 27.32 | 25.38 | 16.01 | 21.51 | - | 29.47 |

Table 10: Evaluation results of audio understanding.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | GPT-4o-Audio | Qwen3-Omni Instruct | Kimi-Audio | Step-Audio-2-mini |
|---|---|---|---|---|---|---|
| **MMAU** | 75.90 | 72.80 | 68.40 | 77.50 | 65.20 | 73.20 |
| **VocalSound** | 92.76 | 89.45 | 82.37 | 91.60 | 94.85 | 87.58 |
| **TUT2017** | 65.43 | 33.15 | 20.74 | 40.74 | 65.25 | 30.67 |
| **ClothoAQA** | 72.83 | 69.67 | 61.87 | 75.16 | 72.21 | 68.39 |
| **Nonspeech7k** | 93.79 | 87.59 | 72.28 | 80.83 | 93.93 | 73.24 |
| **CochlScene** | 70.02 | 45.34 | 34.94 | 43.03 | 80.42 | 44.58 |
| **MELD** | 54.60 | 46.74 | 39.00 | 50.80 | 59.13 | 31.44 |

**Audio-to-Text Chat** We evaluate the ability of LongCat-Flash-Omni to engage in text-based conversations driven by audio instructions using the OpenAudioBench [Li et al., 2025] and VoiceBench [Chen et al., 2024c] benchmarks. These benchmarks assess a wide range of capabilities, including world knowledge, domain-specific understanding, instruction following, and reasoning. To ensure standardized and reproducible results, we employ the official evaluation code provided by the benchmark authors. The results, presented in Table 11, show that LongCat-Flash-Omni achieves strong performance across all benchmark subsets, reaching state-of-the-art results in several cases. Overall, these results demonstrate LongCat-Flash-Omni 's superior ability to conduct audio-driven conversations and perform complex reasoning tasks.

## 7.3 Text Capability Evaluation

### 7.3.1 Base Model Evaluation

We compare the LongCat-Flash-Omni Base model (i.e., after pre-training stage 5) with other strong text-based models across the following core capabilities and corresponding benchmarks.: (1) **General Tasks:** MMLU [Hendrycks et al., 2021a], MMLU-Pro [Wang et al., 2024b], C-Eval [Huang et al., 2023], and CMMLU [Li et al., 2023]. (2) **Reasoning Tasks:** GPQA [Rein et al., 2023], SuperGPQA [M-A-P Team, ByteDance., 2025], BBH [Suzgun et al., 2023], PIQA [Bisk et al., 2019], DROP [Dua et al., 2019], CLUEWSC [Xu et al., 2020], and WinoGrande [Sakaguchi et al., 2019]. (3) **Math Tasks:** GSM8K [Cobbe et al., 2021], MATH [Hendrycks et al., 2021b]. (4) **Coding Tasks:** MBPP+ [Liu et al., 2024f], HumanEval+ [Liu et al., 2024f], MultiPL-E [Cassano et al., 2022], and CRUXEval [Gu et al., 2024]. We follow the same evaluation protocol as in [Meituan, 2025a] to ensure maximum fairness.

Table 12 presents the evaluation results across diverse benchmarks. LongCat-Flash-Omni Base model achieves performance on par with state-of-the-art base models despite its compact active/total parameter size. Furthermore, our model demonstrates no degradation in text capabilities after extensive training with multimodal data, indicating the effectiveness of our training strategy.

Table 11: Evaluation results of audio-to-text chat.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | GPT-4o-Audio | Qwen3-Omni Instruct | Kimi-Audio | Step-Audio-2-mini |
|---|---|---|---|---|---|---|
| **OpenAudioBench** | | | | | | |
| **LlamaQuestions** | 83.33 | 83.00 | 86.30 | 83.30 | 79.33 | 69.70 |
| **ReasoningQA** | 79.71 | 80.30 | 68.71 | 84.16 | 58.02 | 55.64 |
| **TriviaQA** | 86.20 | 90.20 | 76.00 | 75.90 | 62.10 | 45.30 |
| **Webquestions** | 76.00 | 80.90 | 81.20 | 75.20 | 70.20 | 54.40 |
| **AlpacaEval** | 75.43 | 76.58 | 81.61 | 85.43 | 75.73 | 53.92 |
| **VoiceBench** | | | | | | |
| **AlpacaEval** | 4.94 | 4.70 | 4.73 | 4.74 | 4.46 | 3.84 |
| **CommonEval** | 4.32 | 4.11 | 4.37 | 4.54 | 3.97 | 3.19 |
| **OpenBookQA** | 93.41 | 95.16 | 87.90 | 89.70 | 83.52 | 72.97 |
| **SDQA** | 82.46 | 83.54 | 90.10 | 76.90 | 63.12 | 44.85 |
| **MMSU** | 81.95 | 88.32 | 78.90 | 69.00 | 62.17 | 52.00 |
| **AdvBench** | 100 | 97.69 | 99.23 | 99.30 | 100 | 97.00 |
| **IFEval** | 77.99 | 77.83 | 66.81 | 77.80 | 61.10 | 29.80 |

### 7.3.2 Instruct Model Evaluation

A comprehensive and rigorous text capability evaluation of the LongCat-Flash-Omni Instruct model is performed, covering diverse capability dimensions, including general domains, instruction following, mathematical reasoning, general reasoning, and coding, using the following benchmarks:

- **General Domains:** MMLU [Hendrycks et al., 2021a], MMLU-Pro [Wang et al., 2024b], CEval [Huang et al., 2023], and CMMLU [Li et al., 2023].

- **Instruction Following:** IFEval [Zhou et al., 2023], COLLIE [Yao et al., 2024], and Meeseeks [Wang et al., 2025b], Meeseeks evaluates models' instruction-following capabilities in multi-turn scenarios through an iterative feedback framework that simulates realistic human-LLM interactions, enabling models to self-correct based on turn-specific failures and better reflect real-world usage patterns.

- **Mathematical Reasoning:** MATH500 [Lightman et al., 2023], AIME24 [MAA, 2024], and BeyondAIME [ByteDance-Seed, 2025].

- **General Reasoning:** GPQA-diamond [Rein et al., 2023], DROP [Dua et al., 2019], ZebraLogic [Lin et al., 2025], and GraphWalks [OpenAI, 2025a].

- **Coding:** Humaneval+ [Liu et al., 2024f], MBPP+ [Liu et al., 2024f], and LiveCodeBench (2024.08-2025.05) [Jain et al., 2025].

We compare LongCat-Flash-Omni with some representative text chat models including DeepSeek-V3.1 [DeepSeek-AI et al., 2025], Qwen3-235B-A22B (2507 version) [Yang et al., 2025], Kimi-K2 [MoonshotAI, 2025], GPT-4.1 [OpenAI, 2025b], Claude Sonnet-4 [Anthropic, 2025], Gemini2.5-Flash [Comanici et al., 2025] and LongCat-Flash[Meituan, 2025a]. For closed-source models, we conduct evaluations through their official APIs. For models supporting both thinking and non-thinking modes (Qwen3-235B-A22B, Gemini-2.5-Flash, and Claude Sonnet-4), we explicitly configure these models to operate in non-thinking mode for a fair comparison. The evaluation results are presented in Table 13. the comprehensive evaluation demonstrates that LongCat-Flash-Omni maintains superior text capability, with consistently leading performance in different domains. Compared with LongCat-Flash, whose early base model serves as the foundation for LongCat-Flash-Omni, the latter not only exhibits no degradation in text capabilities but even achieves superior performance in certain domains. This highlights the effectiveness of our training strategy and underscores the potential synergy among different modalities in omni-modal model training.

### 7.4 Cross-modality Evaluation

In addition to the aforementioned capabilities, LongCat-Flash-Omni introduces advanced cross-modal understanding and human-like speech interaction with cross-modality inputs. To evaluate these abilities, we compare our model against several strong baselines, including Gemini-2.5-Pro, Gemini-2.5-Flash, Gemini-2.0-Flash, Qwen3-Omni, and Qwen2.5-Omni. For a fair comparison between the LongCat-Flash-Omni instruct model and Gemini-2.5-Pro with thinking mode enabled, we follow the approach of Qwen3-VL by constraining Gemini's thinking budget to 128 tokens, denoted as Gemini-2.5-Pro-ThinkingBudget128. It's important to mention that we couldn't use APIs to test GPT-4o

Table 12: Comparison between LongCat-Flash-Omni and other base models on text-only benchmarks. Values marked with * are sourced from public reports.

| Benchmark | LongCat-Flash-Omni Base | LongCat-Flash Base | DeepSeek-V3.1 Base | Kimi-K2 Base |
|---|---|---|---|---|
| Architecture | MoE | MoE | MoE | MoE |
| # Total Params | 560B | 560B | 671B | 1043B |
| # Activated Params | 27B | 27B | 37B | 32B |
| **General Domains** | | | | |
| MMLU (acc) | 86.81 | 87.05 | 87.46 | 87.47 |
| MMLU-Pro (acc) | 69.05 | 70.32 | 59.29 | 68.36 |
| CEval (acc) | 87.95 | 87.73 | 89.33 | 91.24 |
| CMMLU (acc) | 87.14 | 87.19 | 88.21 | 90.35 |
| **General Reasoning** | | | | |
| GPQA (acc) | 51.76 | 51.09 | 47.16 | 45.89 |
| SuperGPQA (acc) | 54.71 | 54.19 | - | 44.70* |
| BBH (acc) | 90.42 | 90.54 | 89.46 | 89.19 |
| DROP (f1) | 80.75 | 78.39 | 80.74 | 69.81 |
| PIQA (acc) | 92.27 | 92.33 | 93.00 | 95.10 |
| WinoGrande (acc) | 85.95 | 85.08 | 83.50 | 82.87 |
| CLUEWSC (acc) | 91.45 | 91.12 | 88.16 | 76.32 |
| **Mathematical Reasoning** | | | | |
| GSM8K (acc) | 93.10 | 92.19 | 92.22 | 92.27 |
| MATH (acc) | 66.80 | 64.82 | 61.56 | 66.74 |
| **Coding** | | | | |
| MBPP+ (pass@1) | 76.46 | 77.25 | 59.26 | 80.49 |
| HumanEval+ (pass@1) | 69.51 | 65.85 | 67.07 | 69.84 |
| MultiPL-E (pass@1) | 70.76 | 69.25 | 62.00 | 59.22 |
| CRUXEval-I (pass@1) | 71.88 | 71.63 | 65.87 | 65.87 |
| CRUXEval-O (pass@1) | 73.50 | 75.88 | 71.25 | 68.75 |

and Seed-1.6 because they don't have open ones for audio. Instead, we tested their performance on corresponding evaluations by interacting with their live applications in a real-time audio-visual scenario.

### 7.4.1 Cross-modality Understanding Evaluation

In this section, we evaluate cross-modal understanding using publicly available benchmarks, including OmniBench [Li et al., 2024c], WorldSense [Hong et al., 2025], and DailyOmni [Zhou et al., 2025]. We use an internally corrected version of OmniBench, as the publicly released version contains scoring deficiencies. To address the limited quality and coverage of existing benchmarks, we further introduce a new benchmark, UNO-Bench [Chen et al., 2025], comprising 1,880 human-crafted questions spanning 44 task types, with 98% of the questions requiring cross-modal reasoning. We constructed this benchmark by manually annotating our in-house dataset. This approach prevents data contamination and ensures the benchmark is highly representative of real-world application scenarios. In addition to conventional multiple-choice questions, the evaluation includes innovative multi-step open-ended questions, providing a more realistic and discriminative assessment of complex reasoning abilities.

As shown in Table 14, LongCat-Flash-Omni outperforms Gemini-2.5-Flash-non-thinking and achieves performance comparable to Gemini-2.5-Pro-ThinkingBudget128. In particular, on WorldSense and DailyOmni, which emphasize real-world audio-video understanding, LongCat-Flash-Omni demonstrates superior performance, significantly surpassing other open-source omni-modal models. On UNO-Bench, which evaluates cross-modal perception and reasoning, LongCat-Flash-Omni also performs exceptionally well among open-source omni-modal models. These results indicate that LongCat-Flash-Omni achieves highly effective multimodal integration, establishing it as the leading open-source omni-modal model.

Table 13: Evaluation results of frontier chat/instruct models. Values marked with * are sourced from other public reports. Note that DeepSeek-V3.1, Qwen3-235B-A22B, Gemini2.5-Flash, and Claude Sonnet-4 are evaluated under their non-thinking mode.

| Benchmark | LongCat-Flash-Omni Instruct | LongCat-Flash | DeepSeek V3.1 | Qwen3 MoE-2507 | Kimi-K2 | GPT-4.1 | Claude Sonnet-4 | Gemini-2.5 -Flash |
|---|---|---|---|---|---|---|---|---|
| Architecture | MoE | MoE | MoE | MoE | MoE | - | - | - |
| # Total Params | 560B | 560B | 671B | 235B | 1043B | - | - | - |
| # Activated Params | 27B | 27B | 37B | 22B | 32B | - | - | - |
| **General Domains** | | | | | | | | |
| MMLU $_{(acc)}$ | 90.30 | 89.71 | 90.96 | 90.23 | 89.86 | 89.64 | 91.75 | 86.33 |
| MMLU-Pro $_{(acc)}$ | 82.73 | 82.68 | 84.45 | 84.83 | 82.06 | 81.72 | 83.74 | 81.95 |
| CEval $_{(acc)}$ | 91.68 | 90.44 | 89.21 | 92.70 | 91.26 | 79.53 | 86.63 | 78.78 |
| CMMLU $_{(acc)}$ | 89.39 | 84.34 | 88.04 | 88.14 | 89.66 | 77.65 | 86.51 | 78.30 |
| **Instruction Following** | | | | | | | | |
| IFEval $_{(acc)}$ | 82.44 | 89.65 | 86.69 | 88.54 | 88.91 | 85.58 | 88.35 | 83.92 |
| COLLIE $_{(acc)}$ | 45.69 | 57.10 | 43.80 | 49.71 | 56.34 | 50.00 | 51.22 | 48.60 |
| Meeseeks-zh $_{(acc)}$ | 39.05 | 43.03 | 33.83 | 35.32 | 42.79 | 41.54 | 35.07 | 34.84 |
| **Mathematical Reasoning** | | | | | | | | |
| MATH500 $_{(acc)}$ | 97.60 | 96.40 | 96.08 | 98.80 | 97.60 | 90.60 | 93.80 | 98.40 |
| AIME24 $_{(avg@10)}$ | 72.92 | 70.42 | 66.30* | 81.67 | 69.60* | 47.00 | 47.00 | 79.67 |
| BeyondAIME $_{(avg@10)}$ | 47.40 | 43.00 | 36.50 | 57.60 | 36.60 | 22.10 | 20.50 | 44.20 |
| **General Reasoning** | | | | | | | | |
| GPQA-diamond $_{(acc)}$ | 74.41 | 73.23 | 74.90* | 77.43 | 75.76 | 67.68 | 70.71 | 80.30 |
| DROP $_{(f1)}$ | 83.53 | 79.06 | 84.19 | 78.57 | 89.04 | 66.94 | 73.06 | 45.03 |
| ZebraLogic $_{(acc)}$ | 86.00 | 89.30 | 85.30 | 94.22 | 89.11 | 56.30* | 80.10 | 57.00 |
| GraphWalks-128K $_{(precision)}$ | 56.00 | 51.05 | 73.54 | 80.72 | 47.50 | 85.02 | 80.57 | 64.83 |
| **Coding** | | | | | | | | |
| LiveCodeBench $_{(pass@1)}$ | 52.64 | 48.02 | 56.40* | 46.48 | 46.70 | 39.21 | 45.59 | 39.65 |
| Humaneval+ $_{(pass@1)}$ | 90.85 | 88.41 | 92.68 | 94.51 | 85.98 | 93.29 | 94.51 | 87.80 |
| MBPP+ $_{(pass@1)}$ | 80.16 | 79.63 | 79.89 | 79.89 | 81.75 | 79.37 | 80.16 | 76.19 |

Table 14: Evaluation of cross-modality understanding. We use an internally corrected version of OmniBench, as the publicly released version contains scoring deficiencies.

| Benchmark | LongCat-Flash-Omni Instruct | Gemini-2.5-Pro ThinkingBudget128 | Gemini-2.5-Flash non-thinking | Qwen3-Omni Instruct | Qwen2.5-Omni Instruct |
|---|---|---|---|---|---|
| OmniBench | 61.38 | 66.80 | 54.99 | 58.41 | 48.16 |
| WorldSense | 60.89 | 63.96 | 58.72 | 52.01 | 46.69 |
| DailyOmni | 82.38 | 80.61 | 80.78 | 69.33 | 47.45 |
| UNO-Bench | 49.90 | 64.48 | 54.30 | 42.10 | 32.60 |

### 7.4.2 Real-time Audio-Visual Interaction Evaluation

Existing cross-modal evaluation benchmarks still exhibit a noticeable gap from real-world user experiences, where real-time audio-visual interaction (audio-visual input with audio output) is essential. To the best of our knowledge, no prior work has systematically evaluated this form of real-time multimodal interaction. To that end, we built a proprietary, end-to-end framework to measure the quality of a model's audio-visual interaction, specifically its ability to engage users naturally and fluently in real-world scenarios.

Table 15: Real-time audio-visual interaction evaluation.

| Metrics | Doubao | GPT-4o | iFlytek Spark | StepFun | ChatGLM | Qwen2.5-Omni | Qwen3-Omni | LongCat-Flash-Omni |
|---|---|---|---|---|---|---|---|---|
| Score | 1.92 | 1.79 | 1.25 | 1.22 | 0.99 | 0.96 | 0.81 | 1.37 |
| 95% CI | [1.85, 1.98] | [1.72, 1.85] | [1.18, 1.32] | [1.15, 1.28] | [0.93, 1.05] | [0.89, 1.02] | [0.75, 0.87] | [1.30, 1.44] |

The data construction process involves 10 trained professional conversationalists who conducted multi-turn dialogue sessions (approximately three minutes each) with each model under evaluation. These interactions are carried out via real-time audio-visual interfaces on the official app or web platforms provided by the respective model developers. A total of 200 dialogue sessions have been collected for each model, covering common real-world video call scenarios
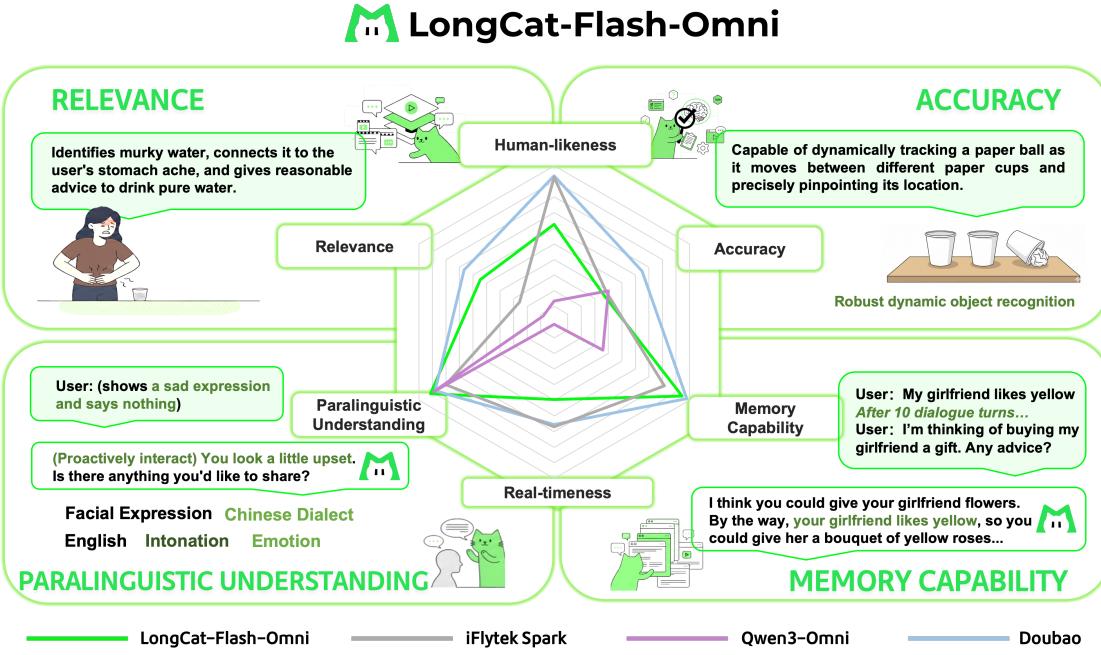
Figure 12: Qualitative analysis of real-time audio-visual interaction.

Table 16: Percentage of good cases in qualitative analysis.

| Metrics | Doubao | GPT-4o | iFlytek Spark | StepFun | ChatGLM | Qwen2.5-Omni | Qwen3-Omni | **LongCat-Flash-Omni** |
|---|---|---|---|---|---|---|---|---|
| Real-timeness | 65.5 | 71.5 | 67.0 | 18.0 | 3.0 | 4.0 | 1.5 | 49.5 |
| Human-likeness | 93.5 | 69.0 | 92.5 | 70.5 | 76.5 | 59.0 | 13.5 | 62.5 |
| Paralinguistic Understanding | 88.0 | 87.5 | 80.0 | 87.0 | 84.5 | 89.0 | 89.0 | 91.5 |
| Relevance | 66.5 | 60.0 | 25.5 | 31.5 | 25.0 | 16.0 | 8.0 | 54.5 |
| Accuracy | 65.0 | 55.0 | 38.5 | 42.5 | 33.5 | 35.5 | 40.0 | 36.0 |
| Memory Capability | 98.0 | 98.0 | 81.5 | 92.5 | 83.0 | 64.0 | 36.0 | 94.5 |

across four categories: problem solving, entertainment, self-improvement, and emotional support (50 samples per category).

The assessment process combines quantitative user ratings with qualitative expert analysis. For quantitative evaluation, 250 real users independently triple-annotated the complete interaction videos, rating naturalness and fluency on a four-point scale: 0 for *completely unnatural*, 1 for *partially unnatural and affecting interaction*, 2 for *partially unnatural but not affecting interaction*, and 3 for *completely natural and fluent*. For qualitative analysis, expert annotators conduct a dimensional breakdown to identify specific factors affecting naturalness and fluency across six key factors, including *real-timeness*, *human-likeness*, *paralinguistic understanding*, *relevance*, *accuracy* and *memory capability*. This approach provides comprehensive qualitative insights into each model's performance.

We compare LongCat-Flash-Omni with five audio-visual interaction products including Doubao[5], GPT-4o[6], iFlytek Spark[7], StepFun[8] and ChatGLM[9], as well as two open-source models Qwen2.5-Omni and Qwen3-Omni.

The quantitative ratings are presented in Table 15. LongCat-Flash-Omni achieves the third-highest score for naturalness and fluency in end-to-end interaction. Comparing with audio-visual interaction products, LongCat-Flash-Omni ranks behind Doubao and GPT-4o but outperforms iFlytek Spark and StepFun. Notably, LongCat-Flash-Omni demonstrates a substantial advantage over open-source alternatives, scoring 0.56 points higher than the current SOTA open-source model, Qwen3-omni.

---

[5] https://www.doubao.com/chat/, the evaluation period was August 11-15, 2025.

[6] https://chat.chatbot.app/?model=gpt-4o, the evaluation period was August 18-22, 2025.

[7] https://xinghuo.xfyun.cn/desk, the evaluation period was August 25-29, 2025.

[8] https://www.stepfun.com/chats/new, the evaluation period was September 8-12, 2025.

[9] https://chat.z.ai/, the evaluation period was September 1-5, 2025.

We present the qualitative analysis results in Table 16, together with some case study in Figure 12. LongCat-Flash-Omni excels in paralinguistic understanding, relevance, and memory capability, performing on par with top-tier models. Notably, LongCat-Flash-Omni actively interprets user emotions from both facial expressions and vocal cues, demonstrating its superior paralinguistic understanding. Regarding relevance, LongCat-Flash-Omni exhibits strong comprehension capabilities by closely tracking dialogue topics and generating highly correlated responses. This comprehension capability, combined with robust memory retention, enables LongCat-Flash-Omni to recall information from many turns earlier, resulting in more fluent and natural interactions. However, certain gaps remain compared with leading models, particularly in real-timeness, human-likeness, and accuracy. Specifically, in terms of real-timeness, our model tends to be overly sensitive to user pauses, often initiating responses prematurely and interrupting users mid-conversation. Regarding human-likeness, it occasionally exhibits pronunciation errors, stuttering, and robotic or electronic audio artifacts. In terms of accuracy, while our model shows strong capability in recognizing dynamic objects, its recognition performance declines when processing text and numerical information. Additionally, we observe a tendency for our model to over-agree with users' statements while overlooking relevant visual content. These aspects will be further optimized in the future work.

## 8  Conclusion

In this report, we present LongCat-Flash-Omni, a next-generation open-source omni-modal model that unifies robust offline multimodal understanding with real-time audio-visual interaction in a single, cohesive framework. LongCat-Flash-Omni demonstrates that large-scale models can effectively perceive, integrate, and generate across diverse modalities, including text, audio, image, and video, without sacrificing performance in any individual domain.

We address major challenges in building such a system: cross-modal heterogeneity, unified offline and streaming capabilities, and low-latency interaction. Through a carefully designed multi-stage early-fusion pretraining pipeline, LongCat-Flash-Omni achieves deeply integrated representations that enable synergistic multimodal reasoning while preserving unimodality strength. Our introduction of human-in-the-loop data construction and a 128K-token context window further enhances multi-turn dialogue, temporal reasoning, and memory capabilities in dynamic interactive scenarios. On the architectural side, the adoption of the ScMoE backbone with zero-computation experts, together with lightweight modality encoders and decoder, allows the model to support real-time audio-visual interaction.

Extensive evaluations show that LongCat-Flash-Omni not only achieves state-of-the-art performance on omni-modal benchmarks such as Omni-Bench and WorldSense, but also matches or exceeds closed-source systems in key unimodal tasks, including image and video understanding as well as audio comprehension. Moreover, subjective assessments confirm the model's ability to deliver natural, low-latency, high-quality interactive experiences, highlighting its potential as a foundation for next-generation human-AI interfaces.

Looking forward, LongCat-Flash-Omni lays a strong foundation for the continued evolution of omni-modal intelligence. Future work will focus on expanding the diversity and scale of training data, integrating an adaptive thinking mode, refining streaming and generation capabilities, and exploring richer forms of embodied and interactive intelligence. We believe that the release of LongCat-Flash-Omni will not only accelerate research on multimodal understanding and generation but also inspire new applications and paradigms for building human-centered, AGI-oriented systems.

# 9 Contributions

The list of authors is in alphabetical order. Names marked with an asterisk (∗) indicate people who have left our team.

Bairui Wang
Bayan
Bin Xiao
Bo Zhang *
Bolin Rong
Borun Chen
Chang Wan
Chao Zhang
Chen Chen (陈晨)
Chen Chen (陈琛)
Chen Huang
Chengxu Yang
Chengzuo Yang
Cong Han *
Dandan Peng
Delian Ruan
Detai Xin
Disong Wang
Dongchao Yang
Fanfan Liu
Fengjiao Chen
Fengyu Yang
Gan Dong
Gang Huang
Gang Xu
Guanglu Wan
Guoqiang Tan
Guoqiao Yu
Haibo Qiu
Hao Lu
Hongbo Liu
Hongyu Xiang
Jiaheng Wu

Jian Yang
Jiaxing Liu
Jing Huang
Jingang Wang
Jinrui Ding
Juchao Jiang
Jun Kuang
Jun Wang
Junhui Mei
Ke Ding
Kefeng Zhang
Lei Chen
Liang Shi
Limeng Qiao
Liming Zheng
Lin Ma
Liuyang Guo
Liya Ma
Luying Sun
Man Gao
Mengshen Zhu
Miao Cao
Minliang Lin
Nuo Xu
Peng Shi
Qi Zhang
Qian Fang
Qian Wang
Qian Yang
Quanxiu Wang
Rongxiang Weng
Rongxin Guo *
Ruoxuan Liang

Senbin Yang
Shanbo Xu
Shanglin Lei
Shengze Ye
Shimin Chen
Shuaiqi Chen
Shujie Hu
Shuo Li
Siqi Yang
Siyu Ren
Siyu Xu
Song Li
Songxiang Liu
Tianhao Bai
Tianye Dai
Wei Hong
Wei Wang
Weixiao Zhao
Wengang Cao
Wenlong He
Wenlong Zhu
Xi Nan
Xi Su
Xiaohan Zhao
Xiaohao Wang
Xiaoyu Li
Xiaoyu Wang
Xiaoyu Zhao
Xin Chen
Xin Pan
Xiusong Sun
Xu Xiang
Xudong Xing

Xuezhi Cao
Xunliang Cai
Yang Yang
Yanli Tan
Yao Yao
Yerui Sun
Yi Chen
Yifan Lu
Yin Gong
Yining Zhang
Yitian Chen
Yiyang Gan
Yuchen Tang
Yuchen Xie
Yueqian Wang
Yuewen Zheng
Yufei Zhang
Yufeng Zhong
Yulei Qian
Yuqi Peng
Yuwei Jiang
Zeyang Hu
Zheng Zhang
Zhengkun Tian *
Zhiqing Hong
Zhixiong Zeng
Zhuqi Mi
Ziran Li
Ziwen Wang
Ziyi Zhao
Ziyuan Zhuang
Zizhe Zhao

# References

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

OpenAI. Hello gpt-4o, May 2024. URL https://openai.com/index/hello-gpt-4o/.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025.

Qingpei Guo, Kaiyou Song, Zipeng Feng, Ziping Ma, Qinglong Zhang, Sirui Gao, Xuzheng Yu, Yunxiao Sun, Tai-Wei Chang, Jingdong Chen, et al. M2-omni: Advancing omni-mllm for comprehensive modality support with competitive performance. *arXiv preprint arXiv:2502.18778*, 2025.

Tianpeng Li, Jun Liu, Tao Zhang, Yuanbo Fang, Da Pan, Mingrui Wang, Zheng Liang, Zehuan Li, Mingan Lin, Guosheng Dong, et al. Baichuan-audio: A unified framework for end-to-end speech interaction. *arXiv preprint arXiv:2502.17239*, 2025.

Zuyan Liu, Yuhao Dong, Jiahui Wang, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Ola: Pushing the frontiers of omni-modal language model. *arXiv preprint arXiv:2502.04328*, 2025.

Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, et al. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv preprint arXiv:2501.01957*, 2025.

Meituan. Longcat-flash technical report. *arXiv preprint arXiv:2509.01322*, 2025a.

Meituan. Longcat-flash-thinking technical report, 2025b. URL https://arxiv.org/abs/2509.18883.

Limeng Qiao, Yiyang Gan, Bairui Wang, Jie Qin, Shuang Xu, Siqi Yang, and Lin Ma. Univitar: Unified vision transformer with native resolution. *arXiv preprint arXiv:2504.01792*, 2025.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

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.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

Xiaohan Zhao, Hongyu Xiang, Shengze Ye, Song Li, Zhengkun Tian, Guanyu Chen, Ke Ding, and Guanglu Wan. Longcat-audio-codec: An audio tokenizer and detokenizer solution designed for speech large language models, 2025a. URL https://arxiv.org/abs/2510.15227.

Shiliang Zhang, Ming Lei, Zhijie Yan, and Lirong Dai. Deep-fsmn for large vocabulary continuous speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5869–5873. IEEE, 2018.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024a.

Weilin Cai, Juyong Jiang, Le Qin, Junwei Cui, Sunghun Kim, and Jiayi Huang. Shortcut-connected expert parallelism for accelerating mixture-of-experts. *arXiv preprint arXiv:2404.05019*, 2024.

Shimin Chen, Xiaohan Lan, Yitian Yuan, Zequn Jie, and Lin Ma. Timemarker: A versatile video-llm for long and short video understanding with superior temporal localization ability. *arXiv preprint arXiv:2411.18211*, 2024a.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52, 2024.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023.

Amro Abbas, Evgenia Rusak, Kushal Tirumala, Wieland Brendel, Kamalika Chaudhuri, and Ari S Morcos. Effective pruning of web-scale datasets based on complexity of concept clusters. *arXiv preprint arXiv:2401.04578*, 2024.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019.

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 e-prints*, pages arXiv–2409, 2024.

Zhixiong Zeng, Jing Huang, Liming Zheng, Wenkang Han, Yufeng Zhong, Lei Chen, Longrong Yang, Yingjie Chu, Yuzhi He, and Lin Ma. Uitron: Foundational gui agent with advanced perception and planning. *arXiv preprint arXiv:2508.21767*, 2025.

Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8428–8437, 2025a.

Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/dcba6be91359358c2355cd920da3fcbd-Abstract-Conference.html.

Paul Michel, Tatsunori Hashimoto, and Graham Neubig. Distributionally robust models with parametric likelihood ratios. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=a34GrNaYEcS.

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. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Brandon Breakthrough. Pyscenedetect: Video scene cut detection tool and library, 2025. URL https://github.com/Breakthrough/PySceneDetect. Available at https://pyscenedetect.readthedocs.io/.

Yijie Zheng, Bangjun Xiao, Lei Shi, Xiaoyang Li, Faming Wu, Tianyu Li, Xuefeng Xiao, Yang Zhang, Yuxuan Wang, and Shouda Liu. Orchestrate multimodal data with batch post-balancing to accelerate multimodal large language model training. *arXiv preprint arXiv:2503.23830*, 2025.

Qianli Ma, Yaowei Zheng, Zhelun Shi, Zhongkai Zhao, Bin Jia, Ziyue Huang, Zhiqi Lin, Youjie Li, Jiacheng Yang, Yanghua Peng, et al. Veomni: Scaling any modality model training with model-centric distributed recipe zoo. *arXiv preprint arXiv:2508.02317*, 2025.

Zili Zhang, Yinmin Zhong, Yimin Jiang, Hanpeng Hu, Jianjian Sun, Zheng Ge, Yibo Zhu, Daxin Jiang, and Xin Jin. Disttrain: Addressing model and data heterogeneity with disaggregated training for multimodal large language models. In *Proceedings of the ACM SIGCOMM 2025 Conference*, pages 24–38, 2025.

Zhenliang Xue, Hanpeng Hu, Xing Chen, Yimin Jiang, Yixin Song, Zeyu Mi, Yibo Zhu, Daxin Jiang, Yubin Xia, and Haibo Chen. Pipeweaver: Addressing data dynamicity in large multimodal model training with dynamic interleaved pipeline. *arXiv preprint arXiv:2504.14145*, 2025.

Weiqi Feng, Yangrui Chen, Shaoyu Wang, Yanghua Peng, Haibin Lin, and Minlan Yu. Optimus: Accelerating {Large-Scale}{Multi-Modal}{LLM} training by bubble exploitation. In *2025 USENIX Annual Technical Conference (USENIX ATC 25)*, pages 161–177, 2025.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023. URL https://arxiv.org/abs/2304.11277.

Penghui Qi, Xinyi Wan, Nyamdavaa Amar, and Min Lin. Pipeline parallelism with controllable memory, 2024. URL https://arxiv.org/abs/2405.15362.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:arXiv preprint arXiv:2412.19437*, 2025.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

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? In *European conference on computer vision*, pages 216–233. Springer, 2024b.

xAI. Realworldqa. https://huggingface.co/datasets/xai-org/RealworldQA, 2023. Version 1.0, Accessed: 2024.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models?, 2024b. URL https://arxiv.org/abs/2403.20330.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of CVPR*, 2024.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. In *International conference on machine learning*. PMLR, 2024.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024a.

Fei Wang, Xingyu Fu, James Y Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, et al. Muirbench: A comprehensive benchmark for robust multi-image understanding. *arXiv preprint arXiv:2406.09411*, 2024a.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. *arXiv preprint arXiv:2405.01483*, 2024.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*, 2022.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images, 2021. URL https://arxiv.org/abs/2007.00398.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12), December 2024c. ISSN 1869-1919. doi:10.1007/s11432-024-4235-6. URL http://dx.doi.org/10.1007/s11432-024-4235-6.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, Jin Shi, Fan Wu, Pei Chu, Minghao Liu, Zhenxiang Li, Chao Xu, Bo Zhang, Botian Shi, Zhongying Tu, and Conghui He. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations, 2024. URL https://arxiv.org/abs/2412.07626.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 787–798, Doha, Qatar, October 2014. Association for Computational Linguistics. URL https://aclanthology.org/D14-1086/.

Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3170–3180, 2023.

Junpeng Liu, Yifan Song, Bill Yuchen Lin, Wai Lam, Graham Neubig, Yuanzhi Li, and Xiang Yue. Visualwebbench: How far have multimodal llms evolved in web page understanding and grounding?, 2024d.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, et al. Os-atlas: A foundation action model for generalist gui agents. *arXiv preprint arXiv:2410.23218*, 2024a.

Wei Li, William Bishop, Alice Li, Chris Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on computer control agents. *arXiv preprint arXiv:2406.03679*, 2024a.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. Mvbench: A comprehensive multi-modal video understanding benchmark, 2024b. URL https://arxiv.org/abs/2311.17005.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9777–9786, June 2021.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? *arXiv preprint arXiv: 2403.00476*, 2024e.

Chaoyou Fu, Yuhan Dai, Yondong 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*, 2024b.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding, 2024b. URL https://arxiv.org/abs/2407.15754.

Yilun Zhao, Lujing Xie, Haowei Zhang, Guo Gan, Yitao Long, Zhiyuan Hu, Tongyan Hu, Weiyuan Chen, Chuhan Li, Junyang Song, Zhijian Xu, Chengye Wang, Weifeng Pan, Ziyao Shangguan, Xiangru Tang, Zhenwen Liang, Yixin Liu, Chen Zhao, and Arman Cohan. Mmvu: Measuring expert-level multi-discipline video understanding, 2025b. URL https://arxiv.org/abs/2501.12380.

Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multi-discipline professional videos. 2025. URL https://arxiv.org/abs/2501.13826.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. CMMLU: Measuring massive multitask language understanding in chinese. *arXiv preprint arXiv:2306.09212*, 2023.

Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In *2017 20th conference of the oriental chapter of the international coordinating committee on speech databases and speech I/O systems and assessment (O-COCOSDA)*, pages 1–5. IEEE, 2017.

Jiayu Du, Xingyu Na, Xuechen Liu, and Hui Bu. Aishell-2: Transforming mandarin asr research into industrial scale. *arXiv preprint arXiv:1808.10583*, 2018.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE, 2023.

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

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 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6182–6186. IEEE, 2022.

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

Boyong Wu, Chao Yan, Chen Hu, Cheng Yi, Chengli Feng, Fei Tian, Feiyu Shen, Gang Yu, Haoyang Zhang, Jingbei Li, et al. Step-audio 2 technical report. *arXiv preprint arXiv:2507.16632*, 2025.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. Covost 2 and massively multilingual speech translation. In *Interspeech*, volume 2021, pages 2247–2251, 2021.

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

Yuan Gong, Jin Yu, and James Glass. Vocalsound: A dataset for improving human vocal sounds recognition. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 151–155, 2022. doi:10.1109/ICASSP43922.2022.9746828.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. TUT database for acoustic scene classification and sound event detection. In *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, Budapest, Hungary, 2016.

Samuel Lipping, Parthasaarathy Sudarsanam, Konstantinos Drossos, and Tuomas Virtanen. Clotho-aqa: A crowdsourced dataset for audio question answering. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 1140–1144. IEEE, 2022.

Muhammad Mamunur Rashid, Guiqing Li, and Chengrui Du. Nonspeech7k dataset: Classification and analysis of human non-speech sound. *IET Signal Processing*, 17(6):e12233, 2023.

Il-Young Jeong and Jeongsoo Park. Cochlscene: Acquisition of acoustic scene data using crowdsourcing. In *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 17–21. IEEE, 2022.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*, 2018.

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

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2021a.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*, 2024b.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. C-Eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In *Advances in Neural Information Processing Systems*, 2023.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

M-A-P Team, ByteDance. SuperGPQA: Scaling LLM evaluation across 285 graduate disciplines. *arXiv preprint arXiv:2502.14739*, 2025.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, 2023.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. *arXiv preprint arXiv:1911.11641*, 2019.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*, 2019.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. CLUE: A Chinese language understanding evaluation benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, 2020.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*, 2019.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021b.

Jiawei Liu, Songrun Xie, Junhao Wang, Yuxiang Wei, Yifeng Ding, and Lingming Zhang. Evaluating language models for efficient code generation. *arXiv preprint arXiv:2408.06450*, 2024f.

Federico Cassano, John Gouwar, Daniel Nguyen, Sydney Nguyen, Luna Phipps-Costin, Donald Pinckney, Ming-Ho Yee, Yangtian Zi, Carolyn Jane Anderson, Molly Q Feldman, Arjun Guha, Michael Greenberg, and Abhinav Jangda. MultiPL-E: A scalable and extensible approach to benchmarking neural code generation. *arXiv preprint arXiv:2208.08227*, 2022.

Alex Gu, Baptiste Rozière, Hugh Leather, Armando Solar-Lezama, Gabriel Synnaeve, and Sida I. Wang. Cruxeval: A benchmark for code reasoning, understanding and execution. *arXiv preprint arXiv:2401.03065*, 2024.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Shunyu Yao, Howard Chen, Austin W. Hanjie, Runzhe Yang, and Karthik R Narasimhan. COLLIE: Systematic construction of constrained text generation tasks. In *The Twelfth International Conference on Learning Representations*, 2024.

Jiaming Wang, Yunke Zhao, Peng Ding, Jun Kuang, Zongyu Wang, Xuezhi Cao, and Xunliang Cai. Ask, fail, repeat: Meeseeks, an iterative feedback benchmark for llms' multi-turn instruction-following ability. *arXiv preprint arXiv:2504.21625*, 2025b.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.

MAA. Aime 2024, 2024. URL https://maa.org/math-competitions/american-invitational-mathematics-examination-aime.

ByteDance-Seed. Beyondaime: Advancing math reasoning evaluation beyond high school olympiads. https://huggingface.co/datasets/ByteDance-Seed/BeyondAIME, 2025.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. Zebralogic: On the scaling limits of LLMs for logical reasoning. In *Forty-second International Conference on Machine Learning*, 2025.

OpenAI. Graphwalks dataset, 2025a. URL https://huggingface.co/datasets/openai/graphwalks.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025.

MoonshotAI. Kimi-K2 documentation, 2025. URL https://moonshotai.github.io/Kimi-K2/.

OpenAI. Introducing GPT-4.1 in the api, April 2025b. URL https://openai.com/index/gpt-4-1/.

Anthropic. Introducing claude 4, May 2025. URL https://www.anthropic.com/news/claude-4.

Yizhi Li, Ge Zhang, Yinghao Ma, Ruibin Yuan, Kang Zhu, Hangyu Guo, Yiming Liang, Jiaheng Liu, Zekun Wang, Jian Yang, Siwei Wu, Xingwei Qu, Jinjie Shi, Xinyue Zhang, Zhenzhu Yang, Xiangzhou Wang, Zhaoxiang Zhang, Zachary Liu, Emmanouil Benetos, Wenhao Huang, and Chenghua Lin. Omnibench: Towards the future of universal omni-language models. abs/2409.15272, 2024c.

Jack Hong, Shilin Yan, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Weidi Xie. Worldsense: Evaluating real-world omnimodal understanding for multimodal llms. abs/2502.04326, 2025.

Ziwei Zhou, Rui Wang, and Zuxuan Wu. Daily-omni: Towards audio-visual reasoning with temporal alignment across modalities, 2025. URL https://arxiv.org/abs/2505.17862.

Chen Chen, ZeYang Hu, Fengjiao Chen, Liya Ma, Jiaxing Liu, Xiaoyu Li, and Xuezhi Cao. Uno-bench: A unified benchmark for exploring the compositional law between uni-modal and omni-modal in omnimodels, 2025. URL https://github.com/meituan-longcat/UNO-Bench.