FunAudio-ASR Technical Report

FunAudio team
funasr.email

Abstract

In recent years, automatic speech recognition (ASR) has witnessed transformative advancements driven by three complementary paradigms: *data scaling*, *model size scaling*, and *deep integration with large language models (LLMs)*. However, LLMs are prone to hallucination, which can significantly degrade user experience in real-world ASR applications. In this paper, we present **FunAudio-ASR**, a large-scale, LLM-based ASR system that synergistically combines massive data, large model capacity, LLM integration, and reinforcement learning to achieve state-of-the-art performance across diverse and complex speech recognition scenarios. Moreover, FunAudio-ASR is specifically optimized for practical deployment, with enhancements in streaming capability, noise robustness, code-switching, hotword customization, and satisfying other real-world application requirements. Experimental results show that while most LLM-based ASR systems achieve strong performance on open-source benchmarks, they often underperform on real industry evaluation sets. Thanks to production-oriented optimizations, FunAudio-ASR achieves SOTA performance on real application datasets, demonstrating its effectiveness and robustness in practical settings.

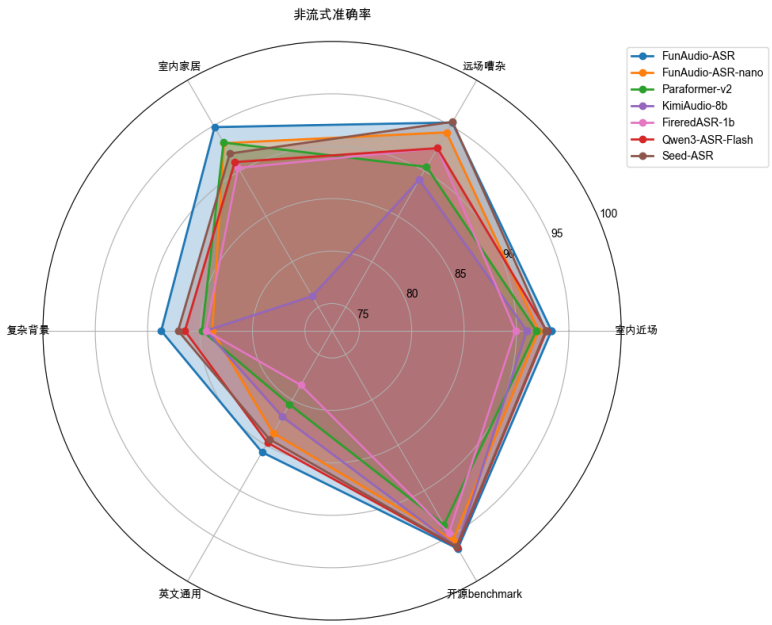


Figure 1: Performance comparison between our FunAudio-ASR and top-tier ASR and speech-text multimodal models, including Paraformer-v2, Kimi-Audio, FireRedASR, Seed-ASR.

1 Introduction

In recent years, under the paradigms of data scaling, model size scaling, and deep integration with large language models (LLMs), automatic speech recognition (ASR) has undergone transformative advancements in both methodology and real-world application scenarios. These three complementary paradigms have collectively driven the evolution of ASR systems from traditional neural-network-based approaches to modern large-model-based architectures, culminating in state-of-the-art (SOTA) performance across diverse acoustic and linguistic conditions.

Data scaling has proven to be a fundamental driver of ASR improvements. The seminal work on Whisper (Radford et al., 2023) provides empirical evidence that ASR performance exhibits a strong positive correlation with the scale of the training data. Their comprehensive experiments demonstrated that increasing the training data volume from 3K hours to over 680K hours results in more than 20-point drop in the English word error rate (WER). This significant improvement underscores the critical role of data diversity and quantity in ASR development, as larger datasets enable models to capture a more comprehensive representation of linguistic and acoustic variations across different languages, accents, speaking styles, and environmental conditions. The availability of massive, multilingual speech corpora has become a cornerstone of modern ASR development, with the most successful systems now leveraging datasets spanning tens of millions of hours.

Model size scaling, particularly the increase in the number of model parameters, has further amplified the benefits of data scaling. The scaling laws observed in large language models (LLMs) have been extended to speech recognition, where increasing model size while maintaining data scaling has yielded substantial performance gains. For example, the Whisper model family demonstrated that increasing the model size from 38M to over 1500M led to more than 40-point WER reduction in multilingual ASR. This synergy between data and model scaling has been pivotal in the development of modern ASR systems, with the largest Whisper variants achieving WERs on par with human transcriptions.

The third paradigm, deep integration with LLMs, represents a paradigm shift in the ASR methodology. Rather than treating ASR as a standalone task, this approach leverages the rich linguistic knowledge and contextual understanding of LLMs to enhance speech recognition. Models such as Seed-ASR (Bai et al., 2024) and FireRedASR (Xu et al., 2025b) have demonstrated that incorporating LLMs can significantly improve ASR performance, particularly in resolving semantic ambiguities and generating more coherent and contextually appropriate transcriptions. These models effectively bridge the gap between speech and text understanding.

Building upon these significant advancements, we propose **FunAudio-ASR**, a large-scale LLM-based ASR system trained on large-scale data. FunAudio-ASR exhibits the following key characteristics:

- **Scaling and Innovative LLM Integration.** FunAudio-ASR is designed to harness the synergistic benefits of data scaling, model size scaling, and LLM integration.
- **State-of-the-art speech recognition accuracy.** Through synergistic advancements in data scaling, model size scaling, and innovative architectural integration with LLMs, FunAudio-ASR achieves unprecedented recognition accuracy across diverse linguistic and acoustic domains, establishing a new state of the art for ASR systems. Our comprehensive evaluations demonstrate that FunAudio-ASR substantially outperforms both our previous small-scale models and leading ASR systems in industry, in terms of critical metrics across multiple challenging benchmarks.
- **Optimization for practical production usage.** Beyond achieving state-of-the-art performance on standardized benchmarks, FunAudio-ASR is meticulously engineered to meet the complex demands of real-world deployment scenarios, with a particular focus on practical usability, reliability, and user experience. We implement a comprehensive suite of optimizations in multiple dimensions, each addressing specific challenges encountered in commercial applications. (1) First, we implement **a highly efficient streaming ASR architecture** for FunAudio-ASR that supports real-time processing with minimal latency, enabling seamless integration into live applications such as video conferencing, live captioning, and voice-controlled devices. (2) Second, we enhance **noise robustness** substantially through a multi-stage approach. (3) Third, we implement **advanced code-switching capabilities** that seamlessly handle transitions between Chinese and English within the same utterance, which is critical for multilingual users in global business environments. (4) Fourth, we integrate **customizable hotword recognition** that allows users to define domain-specific terms

or phrases for enhanced recognition accuracy. This feature is particularly valuable in specialized domains such as healthcare, enterprise, and automotive technology. Hotword recognition achieves To thoroughly evaluate these production-oriented optimizations, we develop a **comprehensive evaluation protocol** that includes both standardized benchmarks and real-world usage scenarios. This protocol encompasses distinct test sets, each simulating different application contexts. Our evaluation results reveal that FunAudio-ASR not only excels in recognition accuracy but also provides superior practical performance. These results demonstrate that FunAudio-ASR successfully bridges the gap between academic research and commercial production readiness, offering a comprehensive solution for addressing real-world speech recognition challenges.

This report is organized as follows. Section 2 and Section 3 introduce the model architecture and the training data. Section 4 elaborates the training paradigm. Section 5 describes how we implement critical production-oriented optimizations. Experiments are presented in Section 6, followed by discussions on the limitations of this work and our future plans.

2 Model Architecture

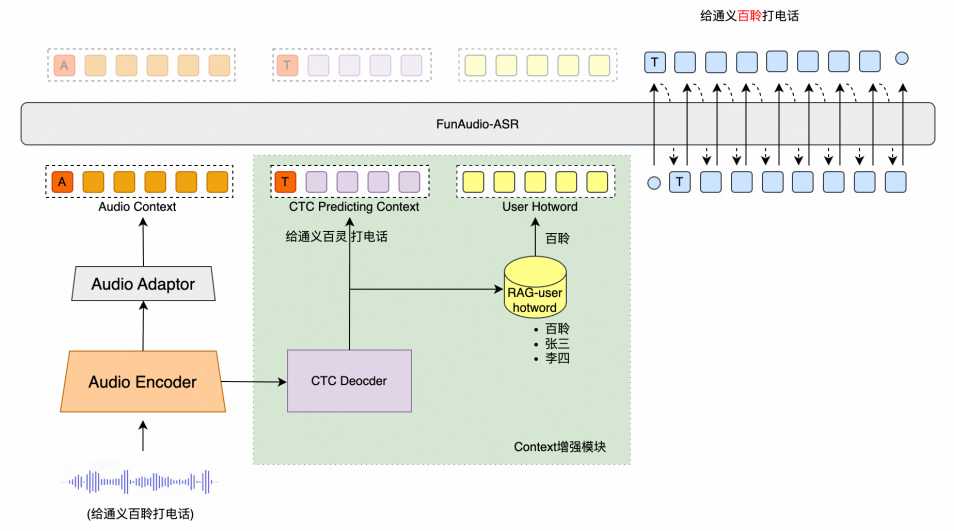


Figure 2: Overview of the FunAudio-ASR model architecture.

FunAudio-ASR comprises four key components: (1) an audio encoder that extracts representations from the input speech, implemented as multiple layers of transformer encoder, (2) an audio adaptor that connects the audio encoder output with the LLM, implemented as two layers of transformer encoder, (3) a CTC decoder that is built upon the audio encoder to obtain the initial recognition hypothesis, which will be used for hotword customization, as described in Section 5.5, and (4) an LLM-based decoder that produces output based on the audio condition and CTC prediction.

To address varying computational resource constraints and inference efficiency requirements, we propose two models with different sizes: FunAudio-ASR and FunAudio-ASR-nano. FunAudio-ASR comprises a 0.7B encoder and a 7B LLM-based decoder, aiming for the highest recognition accuracy, and FunAudio-ASR-nano comprises a 0.2B encoder and a 0.6B LLM-based decoder, seeking to strike a balance between accuracy and efficiency to meet the demands of low-resource scenarios.

3 Data

3.1 Pre-taining Data

The pre-training dataset comprises approximately **tens of millions hours** of audio data, including both unlabeled audio and labeled audio-text data. The unlabeled audio data span a broad range of real-world scenarios in domains such as artificial intelligence, biotechnology, e-commerce, education,

entertainment, finance, and mobility. For labeled data, a comprehensive data processing pipeline is employed, which incorporates voice activity detection (VAD), pseudo-label generation by multiple ASR systems (such as Paraformer-V2 (An et al., 2024b), Whisper, and SenseVoice (An et al., 2024a)), followed by inverse text normalization (ITN). The primary languages in the labeled dataset are Chinese and English.

3.2 Supervised Fine-tuning Data

The supervised fine-tuning (SFT) data consist of approximately **millions of hours** of data, including: human-transcribed data, pseudo-labeled data, environmental noise data, CosyVoice3 (Du et al., 2025) TTS generated data, simulated streaming data, noise augmented data and hotword customized data.

4 Training

4.1 Pre-training of Audio Encoder

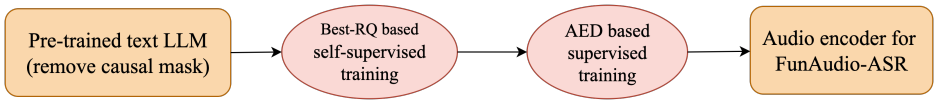


Figure 3: The pre-training pipeline for the audio encoder.

To develop a robust and effective audio encoder for integration into an LLM-based ASR (LLM-ASR) system, we adopt two complementary approaches, as shown in Figure 3. These strategies aim to leverage both self-supervised and supervised learning paradigms to produce high-quality speech representations that can be effectively aligned with linguistic knowledge in the LLM.

Stage 1: Self-supervised pre-training via Best-RQ framework with pre-trained text LLM initialization. The first stage leverages self-supervised pretraining through the Best-RQ (BERT-based Speech pre-Training with Random-projection Quantizer) framework (Chiu et al., 2022), a state-of-the-art self-supervised learning method for speech representation learning. Best-RQ operates by masking and reconstructing speech units, using a quantization module to discretize continuous representations. This approach enables the model to learn general-purpose speech representations without requiring any labeled data, making it highly scalable to vast amounts of unlabeled audio data. Notably, **a key innovation in our implementation of Best-RQ lies in the initialization strategy**. Drawing inspiration from our recent findings (An et al., 2024c), where we demonstrate that layers in pre-trained text LLM can effectively initialize the encoder in ASR systems, we initialize the Best-RQ encoder with weights from layers of a pre-trained text LLM—specifically, Qwen3 model (Yang et al., 2025a). This cross-modal initialization strategy is based on the hypothesis that the deep linguistic and semantic knowledge encoded in the LLM can provide a beneficial inductive bias for learning speech representations. We observe that initializing Best-RQ with a pre-trained text LLM significantly accelerates the training convergence and improves the quality of the learned speech representations compared to random initializations.

Stage 2: Supervised pre-training via attention-based encoder-decoder (AED) framework. The second stage involves supervised pre-training of the audio encoder within a conventional attention-based encoder-decoder architecture. This method follows a well-established and empirically validated training paradigm used in our prior works, specifically the SenseVoice-Large model (An et al., 2024a). In this setup, the encoder is trained end-to-end on large-scale labeled ASR datasets, using standard sequence-to-sequence learning objectives. The primary objective of this supervised pre-training phase is to obtain an encoder that has learned rich acoustic and linguistic features from transcribed speech data. Once trained, the encoder of the AED framework is used to initialize the audio encoder in the downstream LLM-ASR system. This initialization provides a strong starting point for subsequent joint training of the audio and language components, reducing the need for extensive low-level feature learning from scratch and thereby accelerating the training convergence.

4.2 Supervised Fine-tuning

Supervised Fine-tuning (SFT) comprises four sequential stages:

Stage 1: The parameters of the pre-trained audio encoder and the LLM are kept frozen, while the adaptor module is trained to align the audio encoder's output representations with the LLM's semantic space. The training data for this stage is about 200k hours

Stage 2: The LLM parameters are still kept frozen, while the audio encoder and the adaptor module are trained to learn a better semantic representations. This stage uses about 10M hours of low-cost ASR training data and trains one epoch.

Stage 3: The encoder and the adaptor module are frozen, while we update the LLM parameters with Low-Rank Adaptation (LoRA). The purpose of LoRA-based LLM adaptation is to preserve the model's text generation capabilities while ameliorating catastrophic forgetting of the pre-trained knowledge. This LoRA fine-tuning stage uses 20K hours of ASR data

Stage 4: Full-parameter fine-tuning is applied to both the audio encoder and adaptor, while LoRA is employed to fine-tune the LLM simultaneously. In this stage, we only use the high quality data, which contains about 3M hours of speech. The transcriptions are evaluated by three different ASR models, including Whisper-Large-V3, FireRed-ASR, and SenseVoice.

Stage 5: As depicted in Figure 2, we add a CTC decoder on top of the audio encoder. During this training stage, the audio encoder is frozen and only the CTC decoder is trained. This CTC decoder is used to obtain the initial recognition hypothesis by greedy search. Then, this one-pass result is used for retrieval-augmented generation (RAG) to obtain the context information.

4.3 Contextual Supervised Fine-tuning

As a content prior, context can effectively help the model identify and disambiguate key text content from easily confusable pronunciation in ASR tasks, and improve the accuracy of long-term continuous recognition in complex scenarios. Consequently, after the SFT training (Section 4.2), we further train FunAudio-ASR on the contextual and long-duration data to expand its contextual modeling capability.

The duration of the audio samples can be up to 5 minutes. For the longer samples, we segment the sample and add the transcript of the previous segment in front of the current audio segment as prompts. Since high-quality contextual audio data is severely limited, we construct over 50K hours of SFT data with contextual content through the following steps.

Step 1: Keyword Extraction: To generate contextual information related to the current conversation content, we first extract keywords from its transcript using Qwen3-32B (Yang et al., 2025b). Keywords typically include entities, professional terms, and specific time periods. They are words that ASR systems often fail to recognize.

Step 2: Relevant Context Synthesis: We use the Qwen3-32B model to synthesize contextual content. Given the current conversation content and the extracted keywords, we prompt Qwen3-32B to synthesize multiple, diverse contextual content that align with spoken conversation characteristics. For the synthesized context, we then use keyword matching to filter out contextual pieces that do not contain the specified keywords. If no keywords are extracted from the current conversation in the previous step, the LLM is prompted to synthesize context based solely on the current conversation content.

Step 3: Irrelevant context combination: To prevent the model from being overly dependent on the context, we randomly sample five irrelevant contextual pieces for each conversation sample from the dataset and mix them with the synthesized relevant context to form the final contextual SFT training data.

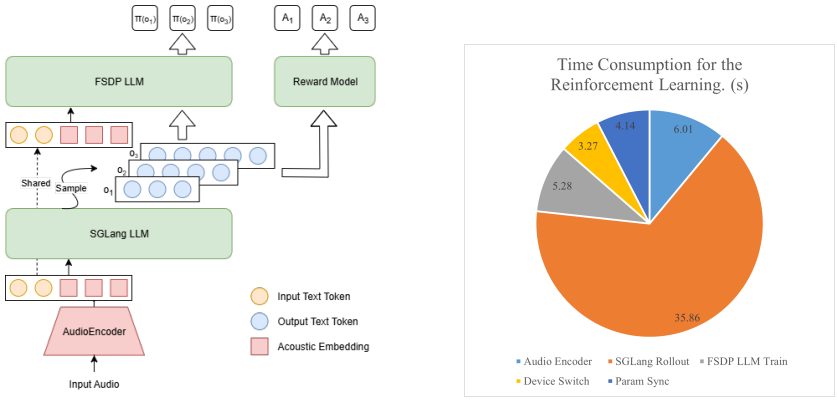
4.4 Reinforcement Learning

4.4.1 The RL Framework for Large Audio-Language Models

We design **FunRL**, an efficient reinforcement learning (RL) framework tailored for large audio-language models (LALMs). Different from text LLMs, FunAudio-ASR, as an LALM, incorporates an audio encoder to convert speech into embeddings—a component that is not natively supported by existing RL frameworks such as Verl (Sheng et al., 2024) or Trl (von Werra et al., 2020). As illustrated in Figure 4a, FunRL orchestrates the audio encoder, rollout, and policy modules using Ray,

enabling them to alternately utilize GPU resources. In the audio encoder inference stage, all input audio clips are batched and processed through a Torch-based encoder. The encoder extracts audio embeddings in parallel and transfers the resulting embeddings from GPU to CPU. Subsequently, the SGLang-based LLM rollout takes control of the GPU to generate multiple hypothesis sequences based on the audio embeddings and the instruction text tokens. Each hypothesis is assigned a reward according to the predefined rules, which will be detailed later. Finally, the FSDP-based LLM policy model uses the audio embeddings and the generated hypotheses to compute output probabilities and performs policy optimization via RL. After each update, the optimized policy is synchronized back to the rollout module, ensuring that the RL process remains on-policy.

We evaluate the training efficiency of FunRL on 8 A100 GPUs, with results shown in Figure 4b. For approximately one hour of input audio, one training step takes about 54.6 seconds, yielding a real-time factor (RTF) of approximately 0.015. As shown in Figure 4b, the SGLang rollout phase dominates the computation time, while device-switching overhead accounts for less than 6% of the total computation time. This indicates that the strategy of alternating GPU utilization in FunRL is highly efficient, making it a scalable and effective solution for RL training in LALMs.



(a) The FunRL framework. (b) Time consumption analysis.

Figure 4: The framework and time consumption analysis for our FunRL.

4.4.2 GRPO-based RL for ASR

Based on the FunRL framework, we enhance the GRPO-based RL algorithm for FunAudio-ASR. Among various RL algorithms, GRPO

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}\big(\{R_j\}_{j=1}^{G}\big)}{\text{std}\big(\{R_j\}_{j=1}^{G}\big)} \tag{1}$$

The policy is optimized with a clipped objective and a directly imposed KL penalty term.

$$L_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\big(r_{i,t}(\theta)\,\hat{A}_{i,t},\ \text{clip}\big(r_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon\big)\,\hat{A}_{i,t}\big) - \beta\, D_{\text{KL}}\big(\pi_\theta \| \pi_{\text{ref}}\big) \tag{2}$$

where

$$r_{i,t}(\theta) = \frac{\pi_\theta\big(o_{i,t} \mid q,\, o_{i,<t}\big)}{\pi_{\theta_{\text{old}}}\big(o_{i,t} \mid q,\, o_{i,<t}\big)}. \tag{3}$$

We observe that when WER is used as the value function, GRPO becomes similar to Minimum Word Error Rate (MWER), a widely adopted criterion in the ASR community. In this work, we further design a set of new value functions $\{R^k(y_i^*, y_i)\}_{k=1}^{K}$ to enhance both ASR performance and user experience:

- **ASR Accuracy (R_i^1).** To directly optimize the recognition quality, we use the $1 - \text{WER}(y^*, y)$ as the basic value function, and the value range is $[0, 1]$.
- **Keyword Accuracy and Recall (R_i^2).** Since keywords have a significant impact on user experience, we incorporate keyword recall as a reward component. Keywords for each utterance are either manually annotated or identified by an LLM. However, using recall alone tends to increase insertion errors; therefore, we also include keyword accuracy to balance precision and recall.
- **Noise Robustness and Hallucination Suppression (R_i^3).** Hallucination is a common issue in LLM-based ASR systems, especially under noisy conditions. To mitigate this, we detect hallucinated content via regular expression matching and apply penalties proportional to the length of the hallucinated span.
- **Language Match (R_i^4).** In certain cases, the model may inadvertently produce speech translation instead of transcription. To enforce language consistency, we assign a final reward of -1 if the output language does not match the source language.

Except for R_i^4, all of the function results are summed to obtain the final R_i. Although R_i^2 to R_i^4 can be reflected by the ASR accuracy, our experimental results show that adding these rules significantly improves the user experience and reduces the WER on the hard cases.

4.4.3 Constructing the RL Training Data

Addressing practical issues in application scenarios, we construct a small but high-quality RL training data using the following approach.

- **Hardcase Samples.** We collect a large amount of unlabeled speech and transcribe each utterance using the FunAudio-ASR model after contextual SFT along with three other distinct ASR systems including Whisper, FireRed-ASR and SenseVoice. If the outputs from the three external systems are consistent with each other (WER < 5%) but differ significantly from FunAudio-ASR's result (WER > 10%), the sample is identified as a hard case and included in the RL training set.
- **Long-duration Samples.** We select audio segments longer than 20 seconds to improve model performance on extended speech inputs, which are common in real-world applications but leak in the training data (less than 10%).
- **Hallucination-related Samples.** We specifically include data where the base model exhibits hallucination behavior, which could be significant longer than the ground-truth or contains repeated phase. Additionally, we incorporate the utterances with the reference transcripts containing long repetitions of words or phrases—cases that resemble hallucinations but are genuinely present in the audio—to help the model distinguish between real and spurious patterns.
- **Keyword and Hotword Samples.** For utterances without predefined hotwords, we use Qwen-2.5 7B to identify salient keywords. For hotword-specific training, we use the hotwords in the reference transcription as target keywords.
- **Regular ASR Data.** A subset of standard ASR data is included to mitigate catastrophic forgetting and maintain general recognition performance during RL training.

The final RL training data consists of 100K samples, with 20K utterances in each of the above five subsets. Thanks to the efficiency of the FunRL framework, the entire RL training of FunAudio-ASR can be completed within one day with 8 A 100 GPUs.

5 Production-oriented Optimization

5.1 Streaming Ability

To enhance the streaming capability of large audio language model FunAudio-ASR, we construct streaming-style training data that explicitly emulate the streaming decoding process, thereby reducing the mismatch between training and inference. Specifically, we sample a subset of the offline training corpus and transform it into incremental, chunked inputs that expose only past context. Fine-tuning by combining this simulated streaming with previous offline training data improves the model's performance under streaming decoding.

5.2 Noise Robust Training

Given the diverse range of real-world deployment scenarios, it is essential for FunAudio-ASR to maintain reliable performance under challenging acoustic conditions, such as those found in restaurants, train stations, and shopping malls, without significant performance degradation. However, creating a dataset that fully captures the complexity and variability of real-world noise environment is impractical. To address this challenge, we employ a large-scale noisy data augmentation strategy. We begin by selecting approximately 110K hours of low-noise speech and 10K hours of noise samples from our in-house corpus. These are combined to generate around 110K hours of offline simulated noisy speech, with an average signal-to-noise ratio (SNR) of 10 dB and a standard deviation of 5 dB. To further increase data diversity, 30% of the training speech is randomly chosen for online data augmentation, where environmental noise is mixed during training. This comprehensive approach for noise robustness yields approximately **13% average relative performance improvement** on our complex-noise evaluation set.

5.3 Multilingual ASR

The availability of training data varies widely across different languages. Resource-rich languages such as Mandarin Chinese and English have abundant data, whereas languages such as Vietnamese and Thai have comparatively limited resources. The primary FunAudio-ASR model is a Chinese-English model. To improve multilingual ASR performance, we train an additional multilingual FunAudio-ASR model to support these languages. The current multilingual FunAudio-ASR model (denoted by **FunAudio-ASR-ML**) supports Mandarin Chinese, English, Vietnamese, Thai, and Indonesian, with a broader language coverage planned for future releases. During training, we downsample the Mandarin and English data used for the Chinese-English FunAudio-ASR model and upsample the Vietnamese, Thai, and Indonesian data to balance the distribution. In total, this multilingual dataset comprises approximately 500K hours of audio. The training methodology is the same as that used for the Chinese-English FunAudio-ASR model (Section 4).

5.4 Code-switching

Recognition of code-switched speech has always been a challenge. To optimize ASR performance on Chinese-English code-switched speech, we synthesize the code-switching training data as follows. Firstly, we collect over 40K English key words or phrases, covering common domains such as technology, education, finance, and sports. Secondly, we use the Qwen3 (Yang et al., 2025a) model to generate Chinese-English code-switched texts related to given key words randomly selected from the pool described above. Thirdly, we use a Text-to-Speech model to synthesize speech data in a variety of voices for the LLM-generated code-switched texts, and obtain the code-switched training data.

5.5 Hotword Customization

Within FunAudio-ASR, we implement a RAG-based mechanism for hotword customization. Specifically, we construct a hotword vocabulary in which each prespecified hotword is converted into a phoneme sequence (for Chinese) or a word-piece sequence (for other languages) using a predefined lexicon. During inference, we retrieve hotword candidates from the vocabulary based on the phoneme-level or the word-piece-level edit distance between the CTC hypotheses and entries in the hotword vocabulary. The retrieved hotword candidates, together with the audio input and the CTC prediction, are used as the input to the LLM, as depicted in Figure 2, to produce the hotword-customized output.

5.6 Hallucination Mitigation

Hallucination in ASR, where an ASR system generates text that is not present in the input audio, is particularly problematic during silence, abrupt speaker interruptions, or in noisy environments where the model may produce spurious transcriptions even without speech. To mitigate hallucination, FunAudio-ASR adopts the following strategy. During data augmentation, we introduce zero-padding into audio signals before adding noise, thereby creating pure-noise segments. This strategy forces the model to learn recognizing noise-only inputs and aligning its outputs accordingly, hence reducing the likelihood of hallucinated text. We find that this approach helps enhance the robustness, accuracy, and stability of FunAudio-ASR across diverse acoustic conditions.

6 Evaluation

6.1 Evaluation Setting

We evaluate FunAudio-ASR and FunAudio-ASR-ML on both open-source ASR benchmark datasets and real-world industry evaluation sets. For the open-source evaluation, we use corresponding test sets of AIShell1/2, Librispeech, Fleurs, WeNetSpeech, Gigaspeech2 data sets. These open-source datasets have been publicly available for a long time, increasing the risk of data leakage into model training sets. To ensure a more reliable and leakage-free evaluation, we collect newly uploaded videos from YouTube and Bilibili posted after June 30th, which are then manually transcribed to form an independent test set. For noise robustness evaluation, we use real-world audio recordings captured in various environments, including canteen, dinner, meeting, office, outdoor, park, shop, street, subway, supermarket, and walk-street. These are further categorized by acoustic conditions and topic to better assess performance under diverse and challenging scenarios.

6.2 Evaluation results

6.2.1 Overall Results

We first evaluate recently published ASR systems on open-source benchmarks, with results shown in Table 1. On these datasets, all models achieve very low WER, and some open-source models even outperform commercial APIs on Librispeech and AIShell. However, on real industry evaluation sets, Seed-ASR-API demonstrates a clear advantage over other open-source models, particularly in noisy conditions. This indicates that performance on open-source test data may not reliably reflect real-world ASR capabilities, highlighting the importance of regularly updating evaluation sets to prevent data leakage. Compared to both open-source models and commercial APIs, our FunAudio-ASR achieves SOTA performance on both open-source benchmarks and industry datasets. Since all training data was collected before June 30th, we ensure no data leakage during evaluation, making the results trustworthy and reproducible. And the FunAudio-ASR-nano can also outperform the opensource model and close to the Seed-ASR, even it has only 0.8B parameters.

Test set	Whisper-large-v3	Seed-ASR	Seed-ASR*	Kimi-Audio	Step-Audio2	FireRed-ASR	FunAudio-ASR-nano	FunAudio-ASR
AIShell1	4.72	0.68	1.63	0.71	0.63	0.54	1.8	1.22
AIShell2	4.68	2.27	2.76	2.86	2.10	2.58	2.95	2.3
Fleurs-zh	5.18	3.43	3.23	3.11	2.68	4.81	3.47	2.64
Fleurs-en	6.23	9.39	9.39	6.99	3.03	10.79	8.42	5.84
Librispeech-clean	1.86	1.58	2.8	1.32	1.17	1.84	1.94	1.57
Librispeech-other	3.43	2.84	5.69	2.63	2.42	4.52	4.69	3.24
WenetSpeech Meeting	18.39	5.69	7.07	6.24	4.75	4.95	6.82	6.49
WenetSpeech Net	11.89	4.66	4.84	6.45	4.67	4.94	6.04	5.46

Table 1: Evaluation results on **open-source datasets** in terms of WER (%). The Seed-ASR* are evaluated from the official API on the volcengine.

Test set	Seed-ASR	Whisper-large-v3	FireRed-ASR	Kimi-Audio	Paraformer v2	FunAudio-ASR-nano	FunAudio-ASR
In-house	7.20	16.58	10.10	9.02	8.11	7.89	6.66
Fairfield	4.59	22.21	7.49	10.95	9.55	5.75	4.66
Home Scenario	8.08	18.17	9.67	23.79	6.87	6.94	5.17
Complex Background	12.90	32.57	15.56	15.56	15.19	16.17	11.29
English General	15.65	18.56	21.62	18.12	19.48	16.34	14.22
Opensouce	3.83	7.05	5.31	3.20	6.23	4.52	3.60
Average	8.71	19.19	11.63	13.54	10.91	9.60	7.60

Table 2: Word Error Rate (WER, %) evaluation Result on Industry Dataset.

6.2.2 Streaming ASR Performance

In order to evaluate the streaming ability of our FunAudio-ASR model, we evaluate the performance on the same test set used when evaluating the offline speech recognition ability of our FunAudio-ASR model. Table 3 lists the testing results. When comparing with Seed-ASR (Bai et al., 2024), our FunAudio-ASR model have better performance at different test sets and test scenario. Our FunAudio-ASR-nano model also has a competitive effect compared with Seed-ASR, although the model size is only 0.8B.

Test set	Seed-ASR	FunAudio-ASR
In-house	8.64	7.00
Fairfield	5.51	5.33
Home Scenario	9.7	5.33
Complex Background	15.48	12.50
English General	18.78	14.74
OpenSouce Test Sets	3.80	3.60

Table 3: Word Error Rate (WER, %) evaluation result with streaming decoding.

6.2.3 Evaluation on Noise Robustness

	FunAudio-ASR		
Environment	w/o NRT	w/ NRT	NRT + RL
canteen	20.67	20.34	19.88
dinner	14.02	9.88	9.55
meeting	6.45	6.27	6.24
office	15.02	11.58	11.42
outdoor	10.12	9.85	9.58
park	13.67	11.88	11.37
shop	12.22	11.48	11.24
street	12.05	10.58	10.86
subway	14.11	13.31	13.29
supermarket	14.27	8.81	8.75
walkstreet	13.89	13.87	13.94
Average	13.32	11.58	11.45

Table 4: Noise robust evaluation under different environments.

We present the noise robustness evaluation in Table 4. It is evident that noise robust training (NRT) is crucial for industrial applications. In challenging environments such as dinner and supermarket settings, NRT brings over 30% relative improvement, as LLM-based ASR systems tend to generate hallucinated outputs under such complex acoustic conditions. Furthermore, RL further enhances the model's noise robustness.

6.2.4 Code-switching Evaluation

For evaluation, two test sets A and B are used to evaluate the effectiveness of the constructed code-switched training data (Section 5.4). Test sets A and B are randomly selected from the daily_dialogue_mixed_chinese_english_speech_tts dataset (yiwu2, 2025) and our in-house recordings, respectively. The results are shown in Table 5.

6.2.5 Evaluation on Hotword Customization

For the hotword evaluation, we choose the audios with some special topics, including biology, math, religion, food, name, astronomy, chemistry, philosophy, and physics, as the recognition of the

Test set	Offline			Streaming		
	w/o CS	w/o RL	w/ RL	w/o CS	w/o RL	w/ RL
A	4.53	1.70	1.59	6.19	5.85	2.28
B	4.76	4.56	4.50	6.32	5.68	5.07

Table 5: Word Error Rate (WER, %) evaluation Result on code-switched test sets.

Topic	Offline w/o RL			Offline w/RL			Streaming w/o RL			Streaming w/ RL		
	WER	acc	rec	WER	acc	rec	WER	acc	rec	WER	acc	rec
biology	1.67	0.98	0.99	1.70	0.97	1.00	2.04	0.98	0.98	1.97	0.99	0.98
math	0.86	0.99	0.99	0.86	0.99	0.99	1.29	0.99	0.99	1.01	0.99	1.00
religion	3.20	0.98	0.98	2.87	0.99	0.99	3.71	0.99	0.98	3.35	0.99	0.97
food	1.90	0.98	0.99	1.55	0.99	1.00	2.01	0.99	0.99	1.47	0.99	0.99
name	0.53	1.00	0.95	0.35	1.00	1.00	1.29	1.00	0.95	0.88	1.00	0.98
brand	0.41	1.00	0.99	0.33	1.00	0.99	1.08	0.99	0.95	0.38	1.00	1.00
astronomy	2.11	1.00	0.97	1.97	0.99	0.97	2.28	0.98	0.95	2.39	1.00	0.98
chemistry	1.76	0.99	0.97	1.91	0.99	0.98	2.81	0.98	0.97	1.83	0.99	0.97
philosophy	3.03	0.99	0.96	2.84	0.99	0.97	3.31	0.99	0.96	3.03	0.99	0.95
physics	1.72	0.99	1.00	1.82	0.98	1.00	2.31	0.99	0.98	1.8	0.99	0.99

Table 6: Hotword customization comparison between the models w/ or w/o reinforcement learning.

technical terms is crucial but still challenging for most ASR systems. Results in Table 6 shows that FunAudio-ASR can benefit from the hotword customization. On most topics, the recall rate can raise to more than 0.97 for FunAudio-ASR. The FunAudio-ASR shows good performance on the name topic, the recall can increase from 0.75 to 1.0. This indicates the hotword customization can really inspire the target keyword, rather than just provide contextual information.

6.2.6 Multilingual ASR Results

We also evaluate our multilingual ASR model FunAudio-ASR-ML on several open source test sets and in-house industry test sets. Table 7 lists the testing results. From Table 7, we can see that, on the Chinese and English open source test sets and in-house industry test sets, our multi-lingual ASR model FunAudio-ASR-ML have better or comparable effects when comparing with Kimi-Audio (KimiTeam et al., 2025), Qwen2.5-Omini (Xu et al., 2025a). We also compared our model with other multi-lingual ASR models, such as Whisper large v3 (Radford et al., 2023), dolphin-small (Meng et al., 2025), and seamless-m4t large v2 (Communication et al., 2023). When comparing with these models, our FunAudio-ASR-ML model can also get the SOTA performance.

6.2.7 Effect of Reinforcement Learning

Table 8 shows that RL plays a crucial role in FunAudio-ASR training, bringing approximately 4.1% and 9.2% relative improvement under offline and streaming conditions, respectively. For offline ASR, the performance gain is more pronounced on audio from noisy and complex environments compared to clean or open-source data. Notably, the improvement is even more significant in the streaming ASR setting. RL helps suppress both insertion and deletion errors, which may be attributed to early termination or premature prediction before full pronunciation.

As shown in Table 6, RL effectively enhances hotword integration, leading to improvements in both accuracy and recall across most test sets. In certain domains, such as philosophy and religion, the RL model may achieve slightly lower accuracy or recall compared to the baseline; however, the overall WER still decreases. This is because, during RL training, keywords are selected based on the actual transcriptions rather than the input prompts, enabling FunAudio-ASR to better recognize domain-specific terminology—even for professional terms not explicitly included in the hotword list.

Language	Test set	Kimi-Audio	Qwen2.5-Omni	Whisper Large v3	dolphin-small	seamless-m4t-large-v2	FunAudio-ASR-ML
Chinese	fleurs	2.69	3	4.71	5.46	5.15	3.0
	commonvoice	7.21	5.20	12.61	9.94	10.76	5.76
	wenetspeech-test-net	5.37	7.7	9.83	9.63	9.87	6.48
	aishell2-ios-test	2.56	2.63	4.83	4.37	4.79	2.60
	in-house-test-set	36.42	8.04	16.54	9.67	14.85	7.91
English	fleus	4.4	4.1	4.11	N.A.	6.59	3.18
	commonvoice	10.31	7.6	9.66	N.A.	7.63	7.67
	librispeech-test-clean	1.28	1.8	2.56	N.A.	2.56	1.62
	librispeech-test-other	2.42	3.4	4.34	N.A.	4.84	3.39
	in-house-test-set	12.40	14.59	11.78	N.A.	43.74	11.19
Indonesian	fleurs	N.A.	N.A.	6.07	15.86	9.36	8.09
	commonvoice	N.A.	N.A.	7.27	8.91	6.1	4.19
	gigaspeech2-test	N.A.	N.A.	19.11	26.56	22.3	16.18
	in-house-test-set	N.A.	N.A.	23.19	40.16	24.41	21.56
Thai	fleurs	N.A.	N.A.	8.48	9.66	9.25	7.04
	commonvoice	N.A.	N.A.	5.92	3.04	2.81	1.44
	gigaspeech2-test	N.A.	N.A.	19.35	19.15	21.7	16.6
Vietnamese	fleurs	N.A.	N.A.	6.51	15.62	8.07	6.33
	commonvoice	N.A.	N.A.	13.51	12.73	13.85	13.49
	gigaspeech2-test	N.A.	N.A.	13.82	31.98	43.31	8.66
	in-house-test-set	N.A.	N.A.	11.46	40.82	32.10	6.9

Table 7: Word Error Rate (WER, %) or Character Error Rate (CER, %) evaluation result on different multilingual test sets.

Test set	Offline		Streaming	
	w/o RL	w/ RL	w/o RL	w/ RL
In-house	6.55	6.66	7.24	7.00
Fairfield	5.14	4.66	6.96	5.33
Home Scenario	5.19	5.17	6.53	5.73
Complex Background	12.16	11.29	13.53	12.50
Accented	15.17	14.22	15.54	14.74
Opensouce	3.98	3.96	5.17	4.37
Average	8.78	8.42	10.05	9.13

Table 8: Comparison between the models w/ or w/o reinforcement learning.

7 Limitations and Future Plans

Despite strong results across diverse evaluations, our FunAudio-ASR model still has some limitations. First, it is primarily optimized for Chinese and English—particularly for streaming performance and hotword customization—so support for other languages remains limited. Second, the effective context window is constrained; without an external voice activity detection (VAD) module, the system struggles to handle long-duration recordings robustly. Third, the current release does not support far-field or multi-channel audio. We plan to address these limitations in future work.

8 Conclusion

In this paper, we present FunAudio-ASR, a large-scale, LLM-based automatic speech recognition (ASR) system that leverages massive data, extensive model capacity, seamless integration of LLMs, and refinement learning to achieve state-of-the-art performance across diverse and challenging scenarios. Designed with practical deployment in mind, FunAudio-ASR incorporates key optimizations for real-world applications, including enhanced streaming capabilities, robustness to noise, effective handling of code-switching, and customizable hotword support. Experimental results reveal that while many LLM-based ASR systems perform well on open-source benchmarks, they often underperform on real-world industrial evaluation sets. In contrast, FunAudio-ASR—through production-oriented design and optimization—demonstrates superior accuracy on practical application datasets, establishing a new benchmark for high-performance, deployable ASR systems.

9 Authors (in alphabetical order of last name)

- Keyu An
- Yanni Chen
- Chong Deng
- Changfeng Gao
- Zhifu Gao
- Bo Gong
- Xiangang Li
- Yabin Li
- Xiang Lv
- Yunjie Ji
- Yiheng Jiang
- Bin Ma
- Haoneng Luo
- Chongjia Ni
- Zexu Pan
- Yiping Peng
- Zhendong Peng
- Peiyao Wang
- Hao Wang
- Wen Wang
- Wupeng Wang
- Biao Tian
- Zhentao Tan
- Nan Yang
- Jieping Ye
- Jixing Yu
- Qinglin Zhang
- Kun Zou
- Shengkui Zhao
- Jingren Zhou

10 Acknowledgment

We are immensely grateful for the invaluable discussions, support, and assistance we received from many colleagues during the development. Special thanks go to: Yafeng Chen, Yuezhang Wang.

References

Keyu An, Qian Chen, Chong Deng, Zhihao Du, Changfeng Gao, Zhifu Gao, Yue Gu, Ting He, Hangrui Hu, Kai Hu, Shengpeng Ji, Yabin Li, Zerui Li, Heng Lu, Haoneng Luo, Xiang Lv, Bin Ma, Ziyang Ma, Chongjia Ni, Changhe Song, Jiaqi Shi, Xian Shi, Hao Wang, Wen Wang, Yuxuan Wang, Zhangyu Xiao, Zhijie Yan, Yexin Yang, Bin Zhang, Qinglin Zhang, Shiliang Zhang, Nan Zhao, and Siqi Zheng. Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms. *CoRR*, abs/2407.04051, 2024a. doi: 10.48550/arXiv.2407.04051. URL https://doi.org/10.48550/arXiv.2407.04051.

Keyu An, Zerui Li, Zhifu Gao, and Shiliang Zhang. Paraformer-v2: An improved non-autoregressive transformer for noise-robust speech recognition. *CoRR*, abs/2409.17746, 2024b. doi: 10.48550/arXiv.2409.17746. URL https://doi.org/10.48550/arXiv.2409.17746.

Keyu An, Shiliang Zhang, and Zhijie Yan. Are transformers in pre-trained LM A good ASR encoder? an empirical study. *CoRR*, abs/2409.17750, 2024c. doi: 10.48550/arXiv.2409.17750. URL https://doi.org/10.48550/arXiv.2409.17750.

Ye Bai, Jingping Chen, Jitong Chen, Wei Chen, Zhuo Chen, Chuang Ding, Linhao Dong, Qianqian Dong, Yujiao Du, Kepan Gao, Lu Gao, Yi Guo, Minglun Han, Ting Han, Wenchao Hu, Xinying Hu, Yuxiang Hu, Deyu Hua, Lu Huang, Mingkun Huang, Youjia Huang, Jishuo Jin, Fanliu Kong, Zongwei Lan, Tianyu Li, Xiaoyang Li, Zeyang Li, Zehua Lin, Rui Liu, Shouda Liu, Lu Lu, Yizhou Lu, Jingting Ma, Shengtao Ma, Yulin Pei, Chen Shen, Tian Tan, Xiaogang Tian, Ming Tu, Bo Wang, Hao Wang, Yuping Wang, Yuxuan Wang, Hanzhang Xia, Rui Xia, Shuangyi Xie, Hongmin Xu, Meng Yang, Bihong Zhang, Jun Zhang, Wanyi Zhang, Yang Zhang, Yawei Zhang, Yijie Zheng, and Ming Zou. Seed-asr: Understanding diverse speech and contexts with llm-based speech recognition. *CoRR*, abs/2407.04675, 2024. doi: 10.48550/arXiv.2407.04675. URL https://doi.org/10.48550/arXiv.2407.04675.

Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. Self-supervised learning with random-projection quantizer for speech recognition. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 3915–3924. PMLR, 2022. URL https://proceedings.mlr.press/v162/chiu22a.html.

Seamless Communication, Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, Ning Dong, Mark Duppenthaler, Paul-Ambroise Duquenne, Brian Ellis, Hady Elsahar, Justin Haaheim, John Hoffman, Min-Jae Hwang, Hirofumi Inaguma, Christopher Klaiber, Ilia Kulikov, Pengwei Li, Daniel Licht, Jean Maillard, Ruslan Mavlyutov, Alice Rakotoarison, Kaushik Ram Sadagopan, Abinesh Ramakrishnan, Tuan Tran, Guillaume Wenzek, Yilin Yang, Ethan Ye, Ivan Evtimov, Pierre Fernandez, Cynthia Gao, Prangthip Hansanti, Elahe Kalbassi, Amanda Kallet, Artyom Kozhevnikov, Gabriel Mejia Gonzalez, Robin San Roman, Christophe Touret, Corinne Wong,

Carleigh Wood, Bokai Yu, Pierre Andrews, Can Balioglu, Peng-Jen Chen, Marta R. Costa-jussà, Maha Elbayad, Hongyu Gong, Francisco Guzmán, Kevin Heffernan, Somya Jain, Justine Kao, Ann Lee, Xutai Ma, Alex Mourachko, Benjamin Peloquin, Juan Pino, Sravya Popuri, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, Anna Sun, Paden Tomasello, Changhan Wang, Jeff Wang, Skyler Wang, and Mary Williamson. Seamless: Multilingual expressive and streaming speech translation, 2023. URL https://arxiv.org/abs/2312.05187.

Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Chongjia Ni, Xian Shi, Keyu An, Guanrou Yang, Yabin Li, Yanni Chen, Zhifu Gao, Qian Chen, Yue Gu, Mengzhe Chen, Yafeng Chen, Shiliang Zhang, Wen Wang, and Jieping Ye. Cosyvoice 3: Towards in-the-wild speech generation via scaling-up and post-training. *CoRR*, abs/2505.17589, 2025. doi: 10.48550/ARXIV.2505.17589. URL https://doi.org/10.48550/arXiv.2505.17589.

KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, Zhengtao Wang, Chu Wei, Yifei Xin, Xinran Xu, Jianwei Yu, Yutao Zhang, Xinyu Zhou, Y. Charles, Jun Chen, Yanru Chen, Yulun Du, Weiran He, Zhenxing Hu, Guokun Lai, Qingcheng Li, Yangyang Liu, Weidong Sun, Jianzhou Wang, Yuzhi Wang, Yuefeng Wu, Yuxin Wu, Dongchao Yang, Hao Yang, Ying Yang, Zhilin Yang, Aoxiong Yin, Ruibin Yuan, Yutong Zhang, and Zaida Zhou. Kimi-audio technical report, 2025. URL https://arxiv.org/abs/2504.18425.

Yangyang Meng, Jinpeng Li, Guodong Lin, Yu Pu, Guanbo Wang, Hu Du, Zhiming Shao, Yukai Huang, Ke Li, and Wei-Qiang Zhang. Dolphin: A large-scale automatic speech recognition model for eastern languages, 2025. URL https://arxiv.org/abs/2503.20212.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518. PMLR, 2023.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl, 2020.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report, 2025a. URL https://arxiv.org/abs/2503.20215.

Kaituo Xu, Feng-Long Xie, Xu Tang, and Yao Hu. Fireredasr: Open-source industrial-grade mandarin speech recognition models from encoder-decoder to LLM integration. *CoRR*, abs/2501.14350, 2025b. doi: 10.48550/arXiv.2501.14350. URL https://doi.org/10.48550/arXiv.2501.14350.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jian Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025a. doi: 10.48550/arXiv.2505.09388. URL https://doi.org/10.48550/arXiv.2505.09388.

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*, 2025b.

yiwu2. daily dialogue mixed chinese english speech tts dataset. https://huggingface.co/datasets/yiwu2/daily_dialogue_mixed_chinese_english_speech_tts, 2025.