Speech Recognition Meets Large Language Model: Benchmarking, Models, and Exploration

Ziyang Ma[1], Guanrou Yang[1], Yifan Yang[1], Zhifu Gao[2], Jiaming Wang[2], Zhihao Du[2], Fan Yu[2], Qian Chen[2], Siqi Zheng[2], Shiliang Zhang[2], Xie Chen[1]*

[1]MoE Key Lab of Artificial Intelligence, X-LANCE Lab, Shanghai Jiao Tong University
[2]Alibaba Group
{zym.22, chenxie95}@sjtu.edu.cn

Abstract

In this paper, we focus on prompting one of the most important tasks in the field of speech processing, i.e., automatic speech recognition (ASR), with speech foundation encoders and large language models (LLM). Despite the growing body of research in this area, we find that many crucial design decisions in LLM-based ASR systems are often inadequately justified. This lack of clarity impedes the field's progress, making it challenging to pinpoint which design choices truly improve model performance. To address these challenges, we conduct a comprehensive series of experiments that explore various aspects, leading to the optimal LLM-based ASR system. We found that delicate designs are not necessary, while a clean setup with little task-specific design is competent. The models achieve strong performance on the Librispeech and Gigaspeech datasets, compared to both LLM-based models and non-LLM-based models. Finally, we explore the capability emergence of LLM-based ASR in the process of modal alignment. We hope that our study can facilitate the research on extending LLM with cross-modality capacity and shed light on the LLM-based ASR community.

Codes & Checkpoints —
https://github.com/X-LANCE/SLAM-LLM

1 Introduction

Automatic speech recognition (ASR) stands as a cornerstone in the realm of intelligent speech technology, enabling machines to understand and transcribe human speech. The significance of ASR in enhancing human-computer interaction and accessibility makes it a crucial area of research and applications in the field of speech processing.

The evolution of ASR technology has been marked by the adoption of various paradigms, each representing a leap forward in terms of accuracy, efficiency, and applicability (Li 2022). Among these, supervised methods including connectionist temporal classification (CTC) (Graves et al. 2006), attention-based encoder-decoder (AED) (Chan et al. 2016), recurrent neural network transducer (RNN-T) (Graves, Mohamed, and Hinton 2013) and their variants have been pivotal. In addition, employing self-supervised methods for pre-training followed by supervised methods for fine-tuning has also proven to be effective (Baevski et al. 2020; Hsu et al. 2021; Chen et al. 2022; Ma et al. 2023; Yang et al. 2024).

The evolution of the ASR paradigm from previous NN-based ASR models to LLM-based ASR models, stresses differences across loss and criterion design, text prior knowledge, and model scale. The architecture of LLM-based ASR can be conceptualized as consisting of three primary components: a speech encoder, a projector, and an LLM. Recent works in LLM-based ASR often venture into diverse designs, such as compressing the output temporally from the speech encoder (Wu et al. 2023; Fathullah et al. 2024), tackling modal alignment with the projector (Tang et al. 2024; Yu et al. 2024), and fine-tuning the LLM partly or fully (Wu et al. 2023; Li et al. 2023b; Tang et al. 2024; Wang et al. 2023). This paradigm harnesses pre-existing linguistic knowledge, enabling a more holistic understanding of language, which in turn, translates to significant improvements in the speech recognition task.

Despite advancements in the field, existing papers show a variety of design choices that are frequently either inadequately justified through experiments or only briefly addressed. This situation makes it challenging to discern which decisions genuinely contribute to model performance, thereby hindering meaningful and grounded progress in the field. Thus, we pose the question: **What matters when building LLM-based ASR models?**

In this work, we aim to provide experimental clarity on these key design decisions, identify optimal model configurations, and explore interesting phenomena. We first benchmark the performance of LLM-based ASR with different combinations of well-known speech encoders and the latest released large language models and get a bunch of empirical conclusions. For example, LLMs with supervised fine-tuning (SFT, a.k.a. chat model) perform better than raw pre-trained LLMs for the ASR task. Building upon these insights, we get optimal model configurations and achieve exciting performance on the Librispeech (Panayotov et al. 2015) and Gigaspeech (Chen et al. 2021) corpus without delicate designs, compared with the previous best-performing NN-based ASR models and other LLM-based ASR models. Further, our work embarks on an in-depth exploration of the ability of LLM-based ASR models. Interestingly, we observe the capability emergence phenomenon during LLM-based ASR training. The benchmark, models, and explo-

*Corresponding author.

ration show how we harvest the result step by step with a clean setup and little task-specific design. Our streamlined design provides a new baseline for LLM-based ASR.

2 Speech Recognition Meets Large Language Model

Model	Loss	Learnable
Previous NN-based ASR		
Quartznet (Kriman et al. 2020)	CTC	All
Whisper (Radford et al. 2023)	AED	All
Branchformer (Peng et al. 2022)	CTC + AED	All
Conformer (Gulati et al. 2020)	RNN-T	All
Zipformer (Yao et al. 2024b)	Pruned RNN-T	All
Paraformer (Gao et al. 2022)	CIF	All
LLM-based ASR		
LauraGPT (Wang et al. 2023)		All
SpeechGPT (Zhang et al. 2023)		LLM
Li et al. (2023b)	Decoder-	Encoder, LLM Adapter
SpeechLLaMA (Wu et al. 2023)	Only,	Encoder, LLM LoRA
Qwen-Audio (Chu et al. 2023)	Cross	Encoder, MLP
SALMONN (Tang et al. 2024)	Entropy	QF, LLM LoRA
Fathullah et al. (2024)		MLP, LLM LoRA
Yu et al. (2024)		MLP/QF

Table 1: ASR Paradigm with representative models. **QF** means variants of Q-Former (Li et al. 2023a). Both **QF** and **MLP** are projector modules used to align the speech encoder and the LLM.

2.1 Previous NN-based ASR

Previous NN-based ASR systems are designed to align the speech signal with the label sequence accurately. As shown in Table 1, different paradigms are carried out with a series of representative models. Quartznet (Kriman et al. 2020) leverages CTC (Graves et al. 2006), the first E2E technology widely adopted in ASR, yet facing performance limitations due to its frame-independent assumption. Whisper (Radford et al. 2023) utilizes massive pair speech-text data to train the attention-based encoder-decoder (Chan et al. 2016) (AED, a.k.a. LAS in ASR) architecture, empowering the model with the ability to recognize and translate speech in multiple languages. Branchformer (Peng et al. 2022) employs a hybrid architecture that combines CTC and AED (Chan et al. 2016), the integration of the attention mechanism addresses this limitation by introducing implicit language modeling across speech frames. Conformer (Gulati et al. 2020) utilizes neural transducer (Graves, Mohamed, and Hinton 2013), which directly discards the frame-independent assumption by incorporating a label decoder and a joint network, resulting in superior performance. Zipformer (Yao et al. 2024b) adopts Pruned RNN-T (Kuang et al. 2022), which is a memory-efficient variant of the transducer loss, utilizing the pruned paths with minor posterior probabilities. Paraformer (Gao et al. 2022) uses Continuous Integrate-and-Fire (CIF) (Dong and Xu 2020), which offers a soft and monotonic alignment mechanism, estimating the number of tokens and generating hidden variables.

2.2 Existing LLM-based ASR

LLM-based ASR models adopt decoder-only architectures based on a pre-trained LLM as a new paradigm. LauraGPT (Wang et al. 2023) connects a modified Conformer (Gulati et al. 2020) encoder with Qwen-2B (Bai et al. 2023) for end-to-end training for multiple speech and audio tasks, with full parameter fine-tuning performed. SpeechGPT (Zhang et al. 2023) discretizes speech tokens with HuBERT (Hsu et al. 2021) and fine-tunes the LLaMA-13B (Touvron et al. 2023a) with multiple stages. Although both models are computationally expensive, their performance is limited. (Li et al. 2023b) and (Wu et al. 2023) propose to use inserted Gated-XATT-FFN (Alayrac et al. 2022) or side-branched LoRA (Hu et al. 2022) to fine-tune the LLM partially for conducting ASR task, along with a trainable speech encoder. Qwen-Audio (Chu et al. 2023) is an audio-universal model, which uses massive pair data to fine-tune the encoder initialized from the Whisper-large (Radford et al. 2023) model, optimized using the loss of the frozen Qwen-7B (Bai et al. 2023) output for backpropagation. All these models require finetuning the encoder. SALMONN (Tang et al. 2024) uses Whisper-large (Radford et al. 2023) and BEATs (Chen et al. 2023) to encode speech and audio, respectively, along with a window-level Q-Former (win-QF), can perform a variety of audio tasks. (Fathullah et al. 2024) connects Conformer with LLaMA-7B to conduct monolingual and multilingual ASR successfully. These models require the use of LoRA to be effective. Some work (Radhakrishnan et al. 2023; Chen et al. 2024; Li et al. 2024) directly utilize LLMs for generative error correction in a cascade manner. A recent work (Yu et al. 2024) achieves good results on ASR using the only trainable Q-Former or MLP as the projector. The random concatenation training strategy is designed to alleviate the natural problem of Whisper (Radford et al. 2023) requiring an input speech of 30 seconds. These models construct effective models from different aspects; however, the question of what matters when building LLM-based ASR models is unanswered, and a comprehensive exploration is urgent.

2.3 Benckamrking System

Our experimental procedure obeys the KISS (Dalzell 2008) ***(Keep It Simple, Stupid!)*** principle to investigate what matters when building LLM-based ASR models. We construct a concise framework to train a benchmarking system, which contains an off-the-shelf speech encoder, a large language model, and the only trainable MLP projector. There are multiple reasons why MLP is chosen in the benchmarking system. On the one hand, previous work on speech (Yu et al. 2024) and vision (McKinzie et al. 2024) found that different projectors have similar effects under similar parameter scales. On the other hand, our preliminary experiments show that training with a Q-Former-based projector is not as stable and efficient as an MLP-based projector, which has also been demonstrated by recent work (Yao et al. 2024a) in the field of Vision-Language Model (VLM).

Given speech $\mathbf{X^S}$, the corresponding transcript $\mathbf{X^T}$, and the prompt $\mathbf{X^P}$, we first convert the speech into speech features through the speech encoder, which can be written as:

$$\mathbf{H^S} = Encoder(\mathbf{X^S}), \tag{1}$$

where $\mathbf{H^S} = [h_1^S, \cdots, h_T^S]$ has T frames in the temporal

dimension. Due to the sparsity of speech representation, the speech features sequence $\mathbf{H^S}$ is still very long for the LLM to tackle[1], we downsample the speech with a downsampler. More explicitly, we concatenate every k consecutive frames in the feature dimension to perform a k times downsampling, leading to $\mathbf{Z^S} = [z_1^S, \cdots, z_N^S]$, where

$$z_i^S = h_{k*i}^S \oplus h_{k*i+1}^S \oplus \cdots \oplus h_{k*i+k-1}^S, \tag{2}$$

and

$$N = T//k. \tag{3}$$

Next, a projector is applied to transform the speech features $\mathbf{Z^S}$ into $\mathbf{E^S}$ with the same dimension as the LLM input embedding. In our experiments, we use a single hidden layer followed by a ReLU activation and a regression layer as the projector, donated as:

$$\mathbf{E^S} = Linear(ReLU(Linear(\mathbf{Z^S}))). \tag{4}$$

Finally, we feed the speech embedding $\mathbf{E^S}$, transcript embedding $\mathbf{E^T}$, and prompt embedding $\mathbf{E^P}$ into the template to compose the final input $\mathbf{E}$ of LLM, donated as:

$$\mathbf{E^T} = Tokenizer(\mathbf{X^T}), \tag{5}$$

$$\mathbf{E^P} = Tokenizer(\mathbf{X^P}), \tag{6}$$

$$\mathbf{E} = \begin{cases} Template(\mathbf{E^S}, \mathbf{E^P}, \mathbf{E^T}) & \text{if training,} \\ Template(\mathbf{E^S}, \mathbf{E^P}) & \text{if inference.} \end{cases} \tag{7}$$

3 Experiment Setup

3.1 Models and Modules

Speech Encoder Two types of speech encoders are investigated in this paper, which are supervised speech encoders trained on massive speech-text pair data and self-supervised speech encoders trained on large-scale unlabeled speech data. For supervised foundation models, we mainly survey the well-known Whisper (Radford et al. 2023) family of models[2] ranging from tiny to large, including *whisper-tiny, whisper-base, whisper-small, whisper-medium* and *whisper-large-v2*. We discard the decoder of each Whisper model and only use the encoder as a feature extractor. We also investigate *Qwen-Audio Encoder*[3], the encoder fine-tuned from *whisper-large-v2* checkpoint on large-scale speech, audio and music data, released along with Qwen-Audio (Chu et al. 2023) model. For self-supervised models, we investigate *HuBERT*[4] and *WavLM*[5] in different scales, either raw pre-trained or further fine-tuned. For the base-size models, both HuBERT (Hsu et al. 2021) and WavLM (Chen et al. 2022) perform self-supervised pre-training on LibriSpeech (Panayotov et al. 2015) corpus with 960 hours. For the large-size models, HuBERT is trained on Libri-Light (Kahn et al. 2020) corpus with $60,000$ hours, while WavLM is trained on the much larger $94,000$ hours data including LibriLight (Kahn et al. 2020), VoxPopuli (Wang et al. 2021), and GigaSpeech (Chen et al. 2021). Furthermore, HuBERT provides pre-trained models of X-Large size, which is the largest publicly available self-supervised speech encoder. All the models mentioned in this section are obtained from their official repositories.

LLM Two types of large language models are investigated in this paper, which are raw pre-trained LLMs without supervised fine-tuning and chat LLMs with SFT (along with RLHF if conducted). For the pre-trained LLMs, we try *TinyLLaMA* (Zhang et al. 2024)[6] of the 1B-magnitude and *LLaMA-2* (Touvron et al. 2023b)[7] of the 7B-magnitude. For the chat LLMs, *TinyLLaMA-Chat*[8] of the 1B-magnitude, *Phi-2*[9] of the 2B-magnitude, *LLaMA-2-Chat*[10] and *Vicuna* (Chiang et al. 2023)[11] of the 7B-magnitude are considered.

Projector The projector can be viewed as an adaptor for other modalities to perform alignment with LLM. In all our experiments, the output of the speech encoder is 50 Hz, and the downsampling rate $k = 5$, leading to the input speech features $\mathbf{E^S}$ of the large model being 10 Hz. The hidden layer dimension is set to 2048, while the dimension of the speech encoder output $\mathbf{H^S}$ and the LLM input dimension vary depending on the model used, respectively.

3.2 Datasets

To evaluate the capabilities of the LLM-based ASR models, we use the most widely used benchmark for the ASR task, the standard Librispeech (Panayotov et al. 2015) benchmark with 960 hours of training data without any data augmentation or splicing. We use the dev-other subset as the validation set and test-clean/test-other as the test sets, each of which contains 10 hours of speech. We also test our findings on a more diverse, noisy, and challenging dataset, the Gigaspeech (Chen et al. 2021) dataset. We train the model with Gigaspeech-M with $1,000$ hours, select on the DEV set with 10 hours, and test on the TEST set with 40 hours.

3.3 Training Detail

During training, the data is organized in the following format: "*USER: <S> <P> ASSISTANT: <T>*", where *<S>* represents speech embedding, *<P>* represents the prompt, and *<T>* represents the corresponding transcribed text. We only compute the loss on *<T>*, as is common practice. For the optimizing strategy, we use AdamW (Loshchilov and Hutter 2019) with a max learning rate of 1×10^{-4} without a weight decay. For the learning rate scheduler, we conduct warmup at the first $1,000$ steps and then keep the maximum

[1] Speech features are 25, 50, or 100 frames per second in general.

[2] https://github.com/openai/whisper

[3] https://github.com/QwenLM/Qwen-Audio

[4] https://github.com/facebookresearch/fairseq/tree/main/examples/hubert

[5] https://github.com/microsoft/unilm/tree/master/unilm

[6] https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v0.4

[7] https://huggingface.co/meta-llama/Llama-2-7b-hf

[8] https://huggingface.co/TinyLlama/TinyLlama-1.1B-intermediate-step-1431k-3T

[9] https://huggingface.co/microsoft/phi-2

[10] https://huggingface.co/meta-llama/Llama-2-7b-chat-hf

[11] https://huggingface.co/lmsys/vicuna-7b-v1.5

Speech Encoder	Pre-trained Model				Chat Model			
	TinyLLaMA		LLaMA-2		TinyLLaMA-Chat		LLaMA-2-Chat	
	test-clean	test-other	test-clean	test-other	test-clean	test-other	test-clean	test-other
Whisper-tiny	12.72	21.64	16.16	25.17	9.55	21.01	8.97	18.77
Whisper-base	7.35	15.89	17.46	21.84	7.03	15.92	6.37	12.98
Whisper-small	6.61	11.81	6.41	10.88	5.94	11.5	4.51	8.94
Whisper-medium	4.65	8.95	3.35	6.10	5.01	8.67	2.71	6.37
Whisper-large	4.39	8.22	3.01	7.15	4.33	8.62	2.72	6.79

Table 2: A benchmark with different combinations of speech encoders and LLMs to conduct LLM-based ASR. We benchmark Whisper models with different sizes on pre-trained models and chat models with different scales.

learning rate for training all the time. The max training step is set to $100,000$, but we will stop early if the loss on the validation set does not decrease. For the audio embedding provided by the Whisper family of models, we found that not padding would affect the performance. As a result, we pad the speech to 30 seconds for all Whisper models and the batch size is set to 4. For other models, the length of the input audio remains consistent with the original length in the temporal dimension, and the batch is set to 6, which greatly improves the efficiency of training and inference, compared to Whisper models.

3.4 Inference Detail

During inference, the data is organized in the following format: "*USER: <S> <P> ASSISTANT:*", where large language models answer autoregressively. Typically, LLMs utilize sampling algorithms to generate diverse textual outputs. Since speech recognition is a sequence-to-sequence task with deterministic outputs, we use beam search with $beam = 4$ to output the hypothesis corresponding to the speech.

LLM	PPL ↓		WER(%) ↓	
	clean	other	clean	other
LLaMA-2	**53.74**	**58.78**	3.01	7.15
LLaMA-2-Chat	77.60	85.74	2.72	6.79
Vicuna	76.44	84.95	**2.58**	**6.47**

Table 3: Word-level text perplexity (PPL) and word error rate (WER) of different LLMs on Librispeech test-clean (clean) and test-other (other) subsets. Among the listed models, the LLM-based ASR model with Vicuna has the best word error rate, while LLaMA performs the worst.

4 Insights From Benchmarking

In this section, we give benchmarks of combinations of different LLMs and speech encoders and obtain a series of conclusions. We find that chat models perform better than raw pre-trained LLMs on the ASR task and verify that this superiority does not come from the transcribed text data leakage in LLM. We further find that fine-tuned versions of self-supervised speech encoders with limited data outperform supervised foundation ASR encoders.

4.1 Is The Chat Model Better Than The Pre-trained Model in LLM-based ASR?

To answer this question, we benchmark Whisper models with different sizes on pre-trained LLMs and supervised fine-tuned LLMs. We pick TinyLLaMA of the 1B-magnitude and LLaMA-2 of the 7B-magnitude to make a preliminary assessment. As shown in Table 2, the performance of the ASR task improves as the speech encoder parameter size increases, but the improvement is of diminishing marginal benefit for the Whisper family of models. For the choice of LLMs, the chat models work better than the pre-trained models, regardless of the size. One possible explanation is that the chat models take speech embedding as a form of "language" and perform a machine translation task, which is activated during the SFT process.

4.2 Is There Transcribed Text Leakage in The Chat LLM?

Another possible reason for the chat LLM being better for LLM-based ASR is that it introduces the transcribed text information to LLM in the SFT stage, resulting in the model easily outputting the corresponding text after obtaining the speech signal. Thus, word-level text perplexity (PPL) of different LLMs is measured to investigate if the better performance of the chat model is related to domain agreement, rather than supervised fine-tuning.

As shown in Table 3, we measure perplexity on test-clean and test-other subsets. Surprisingly, LLaMA-2 without SFT achieves the lowest perplexity by a large margin compared with chat models, while performing the worst on the word error rate. This proves that the better results of chat models are not due to domain agreement with the transcripts.

4.3 What Matters When Choosing A Chat LLM?

We fix the speech encoder as Whisper-large and then explore a better large language model. As shown in Table 4, the Phi-2 chat model with 2.78B parameters has a comparable word error rate with LLaMA-2 with 6.74B parameters on test-other. Vicuna is an open-source chat LLM fine-tuned on user-shared conversational data collected from ShareGPT[12], utilizing LLaMA as a pre-trained LLM. The LLM-based ASR model shows better results when Vicuna is used as the LLM compared with LLaMA-2 and LLaMA-2-Chat. All the above experimental results confirm larger sizes and better

[12]https://sharegpt.com

LLM	#LLM Params	Hidden Size	#Projector Params	WER(%) ↓ test-clean	WER(%) ↓ test-other
Pre-trained Model					
TinyLLaMA	1.10B	2048	17.31M	4.39	8.22
LLaMA-2	6.74B	4096	21.50M	3.01	7.15
Chat Model					
TinyLLaMA-Chat	1.10B	2048	17.31M	4.33	8.62
Phi-2	2.78B	2560	18.35M	3.88	7.19
LLaMA-2-Chat	6.74B	4096	21.50M	2.72	6.79
Vicuna	6.74B	4096	21.50M	2.58	6.47

Table 4: Explore the performance with different LLMs for LLM-based ASR. The projector is fixed with linear layers and the speech encoder is fixed with Whisper-large-v2.

chat models contribute to the performance of LLM-based ASR systems.

4.4 What Matters When Choosing A Speech Encoder?

We fix Vicuna as the LLM and benchmark the performance of different speech encoders. As shown in Table 5, for the supervised speech encoders, the performance gets better gradually as the parameter size of the speech encoder increases, which is consistent with the conclusion on the exploration of LLMs. When the Qwen-Audio Encoder is used as the speech encoder, the ASR performance is further improved compared with Whisper-large, which indicates that the encoder fine-tuned on other LLM (i.e. Qwen-7B) with gradient backpropagation, can be transferred to another LLM (i.e. Vicuna-7B), and maintain a certain degree of performance.

For the self-supervised learning speech encoders, HuBERT Base and WavLM Base have about 95M parameters, with 768 dimensions of hidden size. In this configuration, the ASR performance is similar compared with Whisper-small with the same scale, where self-supervised learning does not play a role. When scaling the self-supervised speech encoders to 0.3B, WavLM Large outperforms all listed supervised speech encoders, including Whisper-medium with 0.3B parameters and Whisper-large with 0.6B parameters, while the improvement from HuBERT Base to HuBERT Large is not obvious. However, if the HuBERT Large encoder is first fine-tuned on Librispeech 960 hours of training data, and used as the speech encoder to train the projector in LLM-based ASR model, the model achieves a WER of 2.10% on test-clean and 4.26% on test-other, exceeding the performance with WavLM Large as the speech encoder. With Librispeech-960 fine-tuned WavLM Large as the speech encoder, our LLM-based ASR model gets a word error rate of 1.96% on test-clean and 4.18% on test-other, achieving 27.9% and 38.4% relative WER reduction over the model whose encoder is Whisper-medium with similar parameters, respectively. Additionally, inspired by Fuyu (Bavishi et al. 2024), we also try to drop the speech encoder and directly feed the 80-dimensional FBank features into the projector, which lags far behind utilizing well-trained speech encoders, as shown in the first row of Table 5. The experimental results show the effectiveness of using self-supervised speech encoders and scaling the size of speech encoders.

5 Models

In this section, we integrate a bunch of conclusions above together and compare our models with state-of-the-art NN-based ASR models either trained on specific datasets or trained with massive speech, as well as other LLM-based ASR models with delicate designs.

5.1 Compared with NN-based ASR Models

We compare our best recipe with state-of-the-art NN-based models. For specialist models trained on Librispeech-960, we compare with ContextNet (Han et al. 2020), Conformer(Gulati et al. 2020), Branchformer (Peng et al. 2022), and Zipformer (Yao et al. 2024b). All models are of large size, and the results from their papers are demonstrated. These ASR models employ sophisticated system engineering, including SpecAugment and speed perturbation for data augmentation, and the exponential moving average technique for model averaging. To further improve performance, in-domain language models trained on the LibriSpeech language model corpus along with the LibriSpeech-960 transcripts are added for fusing or rescoring. Our LLM-based ASR model achieves a better ASR performance than the best-performing models without using complex system engineering. Compared with general-propose models trained on massive data, Our LLM-based ASR model outperforms Whisper-large-v2 (Radford et al. 2023) in industry, and OWSM-v3.1 (Peng et al. 2024) in the academic community.

For a more challenging and noisy dataset, Gigaspeech, we also compare our model with other well-known models. For specialist models trained on $1,000$ hours Gigaspeech-M, we compare with the model of the original paper train with Kaldi[13], Conformer-Transducer trained with Fairseq[14], and Zipformer-Pruned RNN-T trained with K2[15]. Our LLM-based ASR model also achieves better performance than theirs without using sophisticated systems and data engineering. However, our model does not perform as well as the universal Whisper-large-v2 (Radford et al. 2023), which indicates that the LLM-based ASR model still has limited

[13]https://github.com/kaldi-asr/kaldi

[14]https://github.com/facebookresearch/fairseq

[15]https://github.com/k2-fsa/k2

Speech Encoder	#Encoder Params	Hidden Size	#Projector Params	WER(%) ↓ test-clean	WER(%) ↓ test-other
Acoustic Feature					
FBank	-	80	10.03M	68.95	99.37
Supervised Speech Encoder					
Whisper-tiny	7.63M	394	12.33M	7.07	16.01
Whisper-base	19.82M	512	13.64M	5.07	13.07
Whisper-small	87.00M	768	16.26M	4.19	9.50
Whisper-medium	305.68M	1024	18.88M	2.72	6.79
Whisper-large	634.86M	1280	21.50M	2.58	6.47
+ Qwen-Audio Fine-tuning	634.86M	1280	21.50M	2.52	6.35
Self-supervised Speech Encoder					
HuBERT Base	94.70M	768	16.26M	4.43	10.72
WavLM Base	94.38M	768	16.26M	4.14	9.66
HuBERT Large	316.61M	1024	18.88M	4.53	8.74
+ LS-960 Fine-tuning	316.61M	1024	18.88M	2.10	4.26
WavLM Large	315.45M	1024	18.88M	2.13	4.73
+ LS-960 Fine-tuning	315.45M	1024	18.88M	1.96	4.18

Table 5: Explore the performance with different speech encoders for LLM-based ASR. The projector is fixed with linear layers and LLM is fixed with Vicuna-7B-v1.5. LS-960 means the Librispeech 960 hours dataset.

Model	WER(%) ↓ test-clean	WER(%) ↓ test-other
Specialist Models		
ContextNet-large (Han et al. 2020)	2.1	4.6
+ in-domain LM	1.9	4.1
Conformer-large (Gulati et al. 2020)	2.1	4.3
+ in-domain LM	1.9	3.9
Branchformer-large (Peng et al. 2022)	2.4	5.5
+ in-domain LM	2.1	4.5
Zipformer-large (Yao et al. 2024b)	2.0	4.4
+ in-domain LM	1.9	3.9
Ours	**1.8**	**3.4**
Universal Models		
Whisper-large-v2 (Radford et al. 2023)	2.7	5.2
OWSM-v3.1 (Peng et al. 2024)	2.4	5.0

Table 6: Compared with previous NN-based models. ***Specialist Models*** means models trained on Librispeech-960, and ***in-domain LM*** means language models trained on the LibriSpeech language model corpus along with LibriSpeech-960 transcripts. ***Universal Models*** means general-propose models trained on massive pair data.

Model	Implementation	WER(%) ↓ DEV	WER(%) ↓ TEST
Specialist Models			
Gigaspeech	Kaldi	17.96	17.53
Conformer-Transducer	Fairseq	14.30	14.20
Zipformer-Pruned RNN-T	K2	12.24	12.19
Ours	Ours	**10.6**	**11.1**
Universal Models			
Whisper-large-v2 (Radford et al. 2023)	OpenAI/Whisper	10.5	10.2

Table 7: Compared with SOTA NN-based models from popular code repositories. ***Specialist Models*** means models trained on Gigaspeech-M, and ***Universal Models*** means general-propose models trained on massive pair data.

ability to handle more difficult data with limited training data. All in all, experimental results demonstrate the effectiveness of our exploration and the great potential of LLM-based ASR.

5.2 Compared with Other LLM-based ASR Models

As shown in Table 8, we exhibit different LLM-based ASR models from concurrent work, either ASR-specific or audio-universal. A contemporary work (Yu et al. 2024) employs Whisper-large as the speech encoder and Vicuna-13B as the LLM. The segment-level Q-Former (seg-QF) is utilized as the projector to tackle the compatibility between speech sequences and the LLM. Compared with their method, our LLM-based ASR model with WavLM Large as the encoder yields $13.0/19.2\%$ relative WER reductions on test-clean/other subsets trained with the same 960 hours of Librispeech data, and both encoder and LLM are smaller than their solution. When their model is trained on a larger amount of speech over $4,000$ hours, our model still performs better. Further, we scale the speech encoder to 1B parameters using HuBERT X-Large as the speech encoder, and our model yields $21.7/34.6\%$ relative WER reductions on test-clean/other subsets compared to their solution.

We also compare our model with the latest LLM-based audio-universal models, SALMONN (Tang et al. 2024) and Qwen-Audio (Chu et al. 2023), which provide results on the Librispeech benchmark. Compared with these audio-based multimodal LLMs, our model still achieves better performance despite the large margin in training data. This shows that a concise model combination with limited data can still work well.

5.3 Compared with Different Language Models

LLM-based ASR models can be viewed as connecting large language models with acoustic models and training

Model	Speech Encoder		LLM		Projector		ASR Data(h)	WER(%) ↓	
	Module	Learnable	Module	Learnable	Module	Learnable		test-clean	test-other
LLM-based ASR-specific Models									
Yu et al. (2024)	Whisper-large	✗	Vicuna-13B	✗	seg-QF	✓	960	2.3	5.2
							4,000+	2.1	5.0
Ours	WavLM Large	✗	Vicuna-7B	✗	MLP	✓	960	**2.0**	**4.2**
	HuBERT X-Large							**1.8**	**3.4**
LLM-based Audio-universal Models									
SALMONN (Tang et al. 2024)	Whisper-large, BEATs	✗	Vicuna-13B	LoRA	win-QF	✓	1960	2.1	4.9
Qwen-Audio (Chu et al. 2023)	Whisper-large	✓	Qwen-7B	✗	MLP	✓	30,000+	2.0	4.2

Table 8: Compared with other LLM-based speech models. The specific information of the different modules is given in the table, and all the numbers are obtained from their paper.

them end-to-end. Therefore, we fix the encoder as WavLM Large and test the performance without LM, with the in-domain LM officially provided by Fairseq and speechcolab , and with LLM. We present results on the Librispeech and Gigaspeech-M datasets, respectively. Experimental results show that the LLM-based ASR model performs very well on clean data while losing some performance on noisy data on the Librispeech dataset. Therefore, there is great potential for robust LLM-based ASR.

Model	Librispeech		Gigaspeech	
	test-clean	test-other	DEV	TEST
WavLM Large (pre-trained)				
w/ LLM	2.13	4.73	11.14	11.88
WavLM Large (fine-tuned)				
w/o LM (CTC)	2.66	4.97	12.77	12.77
w/ in-domain LM	2.14	**4.00**	11.01	11.45
w/ LLM	**1.96**	4.18	**10.63**	**11.05**

Table 9: Ablation on Librispeech and Gigaspeech.

Figure 1: Training accuracy with the LLM fixed.

6 Capability Emergence

We observe that there is capability emergence for LLM-based ASR during training within 1 epoch (around 12k steps).

Figure 1 demonstrates the training accuracy of the next token prediction with the training steps, where the LLM is kept as Vicuna-7B and the speech encoders vary. As can be seen from the figure, the speech encoders with better performance, in this case, Whisper Large and WavLM Large, will

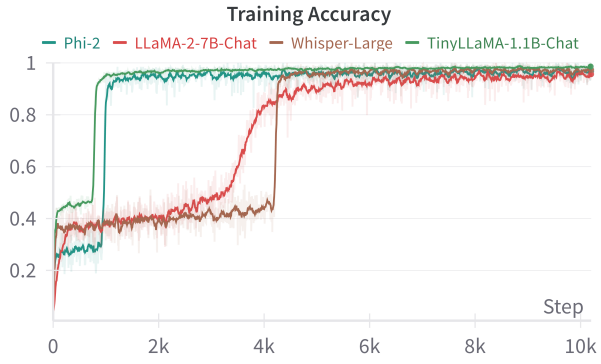

Figure 2: Training accuracy with the speech encoder fixed.

emerge earlier. A possible explanation is that our task is essentially to align speech representations with LLMs, while a powerful speech encoder can provide representations that are easier for the projector to align with LLMs.

We keep the speech encoder as Whisper Large, change different LLMs, and plot the training accuracy, as shown in Figure 2. Experiments show that LLM-based ASR models with smaller LLMs such as TinyLLaMA-Chat and Phi-2 emerge earlier, however, they are not as effective as larger LLMs such as LLaMA-2-7B-Chat and Vicuna-7B. This shows that the larger language models are harder to align with speech features than the smaller ones.

Less training costs will be spent if the model can emerge early, which is yet to be explored. More explorations can be found in supplementary materials.

7 Conclusion and Limitation

In this paper, we systematically explore LLM-based ASR systems with a clean framework. A bunch of conclusions are drawn from benchmarking and optimal configurations are used to train the models with prominent performance. Exploratory experiments show that there is a capability emergence in LLM-based ASR systems. Although there is some progress made in LLM-based ASR, the inference speed is still a bottleneck problem that needs to be solved urgently. We aspire for our research to serve as a step forward in the exploration of LLM-based ASR, offering assistance and insights to the broader community.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 62206171 and No. U23B2018), Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0102, and the Science and Technology Innovation (STI) 2030-Major Projects under Grant 2022ZD0208700.

References

Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; Lenc, K.; Mensch, A.; Millican, K.; Reynolds, M.; et al. 2022. Flamingo: a visual language model for few-shot learning. In *Proc. NeurIPS*.

Baevski, A.; Zhou, Y.; Mohamed, A.; and Auli, M. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. NeurIPS*.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Bavishi, R.; Elsen, E.; Hawthorne, C.; Nye, M.; Odena, A.; Somani, A.; and Taşırlar, S. 2024. Fuyu-8B: A Multimodal Architecture for AI Agents. https://www.adept.ai/blog/fuyu-8b.

Chan, W.; Jaitly, N.; Le, Q.; and Vinyals, O. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Proc. ICASSP*.

Chen, C.; Hu, Y.; Yang, C.-H. H.; Siniscalchi, S. M.; Chen, P.-Y.; and Chng, E.-S. 2024. Hyporadise: An open baseline for generative speech recognition with large language models. *Proc. NeurIPS*.

Chen, G.; Chai, S.; Wang, G.; Du, J.; Zhang, W.-Q.; Weng, C.; Su, D.; Povey, D.; Trmal, J.; Zhang, J.; et al. 2021. Gigaspeech: An evolving, multi-domain ASR corpus with 10,000 hours of transcribed audio. In *Proc. Interspeech*.

Chen, S.; Wang, C.; Chen, Z.; Wu, Y.; Liu, S.; Chen, Z.; Li, J.; Kanda, N.; Yoshioka, T.; Xiao, X.; et al. 2022. WavLM: Large-scale self-supervised pre-training for full stack speech processing. In *Proc. JSTSP*.

Chen, S.; Wu, Y.; Wang, C.; Liu, S.; Tompkins, D.; Chen, Z.; and Wei, F. 2023. BEATs: Audio pre-training with acoustic tokenizers. In *Proc. ICML*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; et al. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https://vicuna.lmsys.org.

Chu, Y.; Xu, J.; Zhou, X.; Yang, Q.; Zhang, S.; Yan, Z.; Zhou, C.; and Zhou, J. 2023. Qwen-Audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*.

Dalzell, T. 2008. *The Routledge dictionary of modern American slang and unconventional English*. Routledge.

Dong, L.; and Xu, B. 2020. CIF: Continuous integrate-and-fire for end-to-end speech recognition. In *Proc. ICASSP*.

Fathullah, Y.; Wu, C.; Lakomkin, E.; Jia, J.; Shangguan, Y.; Li, K.; Guo, J.; Xiong, W.; Mahadeokar, J.; Kalinli, O.; et al. 2024. Prompting large language models with speech recognition abilities. *Proc. ICASSP*.

Gao, Z.; Zhang, S.; McLoughlin, I.; and Yan, Z. 2022. Paraformer: Fast and accurate parallel Transformer for non-autoregressive end-to-end speech recognition. In *Proc. Interspeech*.

Graves, A.; Fernández, S.; Gomez, F. J.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. ICML*.

Graves, A.; Mohamed, A.; and Hinton, G. E. 2013. Speech recognition with deep recurrent neural networks. In *Proc. ICASSP*.

Gulati, A.; Qin, J.; Chiu, C.-C.; Parmar, N.; Zhang, Y.; Yu, J.; Han, W.; Wang, S.; Zhang, Z.; Wu, Y.; et al. 2020. Conformer: Convolution-augmented Transformer for speech recognition. In *Proc. Interspeech*.

Han, W.; Zhang, Z.; Zhang, Y.; Yu, J.; Chiu, C.-C.; Qin, J.; Gulati, A.; Pang, R.; and Wu, Y. 2020. Contextnet: Improving convolutional neural networks for automatic speech recognition with global context. *arXiv preprint arXiv:2005.03191*.

Hsu, W.-N.; Bolte, B.; Tsai, Y.-H. H.; Lakhotia, K.; Salakhutdinov, R.; and Mohamed, A. 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. In *Proc. TASLP*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-rank adaptation of large language models. In *Proc. ICLR*.

Kahn, J.; Rivière, M.; Zheng, W.; Kharitonov, E.; Xu, Q.; Mazaré, P.-E.; Karadayi, J.; Liptchinsky, V.; Collobert, R.; Fuegen, C.; et al. 2020. Libri-light: A benchmark for asr with limited or no supervision. In *Proc. ICASSP*.

Kriman, S.; Beliaev, S.; Ginsburg, B.; Huang, J.; Kuchaiev, O.; Lavrukhin, V.; Leary, R.; Li, J.; and Zhang, Y. 2020. Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions. In *Proc. ICASSP*.

Kuang, F.; Guo, L.; Kang, W.; Lin, L.; Luo, M.; Yao, Z.; and Povey, D. 2022. Pruned RNN-T for fast, memory-efficient ASR training. In *Proc. Interspeech*.

Li, J. 2022. Recent advances in end-to-end automatic speech recognition. In *Proc. APSIPA*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proc. ICML*.

Li, Y.; Chen, P.; Bell, P.; and Lai, C. 2024. Crossmodal asr error correction with discrete speech units. *Proc. SLT*.

Li, Y.; Wu, Y.; Li, J.; and Liu, S. 2023b. Prompting large language models for zero-shot domain adaptation in speech recognition. In *Proc. ASRU*.

Loshchilov, I.; and Hutter, F. 2019. Decoupled weight decay regularization. In *Proc. ICLR*.

Ma, Z.; Zheng, Z.; Tang, C.; Wang, Y.; and Chen, X. 2023. MT4SSL: Boosting Self-Supervised Speech Representation Learning by Integrating Multiple Targets. In *Proc. Interspeech*.

McKinzie, B.; Gan, Z.; Fauconnier, J.-P.; Dodge, S.; Zhang, B.; Dufter, P.; Shah, D.; Du, X.; Peng, F.; Weers, F.; et al. 2024. MM1: Methods, analysis & insights from multimodal llm pre-training. In *Proc. ECCV*.

Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: An ASR corpus based on public domain audio books. In *Proc. of ICASSP*.

Peng, Y.; Dalmia, S.; Lane, I.; and Watanabe, S. 2022. Branchformer: Parallel mlp-attention architectures to capture local and global context for speech recognition and understanding. In *Proc. ICML*.

Peng, Y.; Tian, J.; Chen, W.; Arora, S.; Yan, B.; Sudo, Y.; Shakeel, M.; Choi, K.; Shi, J.; Chang, X.; et al. 2024. OWSM v3. 1: Better and Faster Open Whisper-style Speech Models based on E-Branchformer. *Proc. Interspeech*.

Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *Proc. ICML*.

Radhakrishnan, S.; Yang, C.-H. H.; Khan, S. A.; Kumar, R.; Kiani, N. A.; Gomez-Cabrero, D.; and Tegner, J. N. 2023. Whispering LLaMA: A cross-modal generative error correction framework for speech recognition. *Proc. EMNLP*.

Tang, C.; Yu, W.; Sun, G.; Chen, X.; Tan, T.; Li, W.; Lu, L.; Ma, Z.; and Zhang, C. 2024. SALMONN: Towards generic hearing abilities for large language models. In *Proc. ICLR*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; et al. 2023a. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; et al. 2023b. LLaMA 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, C.; Riviere, M.; Lee, A.; Wu, A.; Talnikar, C.; Haziza, D.; Williamson, M.; Pino, J.; and Dupoux, E. 2021. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In *Proc. ACL*.

Wang, J.; Du, Z.; Chen, Q.; Chu, Y.; Gao, Z.; Li, Z.; Hu, K.; Zhou, X.; Xu, J.; Ma, Z.; et al. 2023. LauraGPT: Listen, Attend, Understand, and Regenerate Audio with GPT. *arXiv preprint arXiv:2310.04673*.

Wu, J.; Gaur, Y.; Chen, Z.; Zhou, L.; Zhu, Y.; Wang, T.; Li, J.; Liu, S.; Ren, B.; Liu, L.; et al. 2023. On decoder-only architecture for speech-to-text and large language model integration. In *Proc. ASRU*.

Yang, Y.; Zhuo, J.; Jin, Z.; Ma, Z.; Yang, X.; Yao, Z.; Guo, L.; Kang, W.; Kuang, F.; Lin, L.; et al. 2024. k2SSL: A Faster and Better Framework for Self-Supervised Speech Representation Learning. *arXiv preprint arXiv:2411.17100*.

Yao, L.; Li, L.; Ren, S.; Wang, L.; Liu, Y.; Sun, X.; and Hou, L. 2024a. DeCo: Decoupling Token Compression from Semantic Abstraction in Multimodal Large Language Models. In *arXiv preprint arXiv:2405.20985*.

Yao, Z.; Guo, L.; Yang, X.; Kang, W.; Kuang, F.; Yang, Y.; Jin, Z.; Lin, L.; and Povey, D. 2024b. Zipformer: A faster and better encoder for automatic speech recognition. In *Proc. ICLR*.

Yu, W.; Tang, C.; Sun, G.; Chen, X.; Tan, T.; Li, W.; Lu, L.; Ma, Z.; and Zhang, C. 2024. Connecting speech encoder and large language model for ASR. In *Proc. ICASSP*.

Zhang, D.; Li, S.; Zhang, X.; Zhan, J.; Wang, P.; Zhou, Y.; and Qiu, X. 2023. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Proc. EMNLP*.

Zhang, P.; Zeng, G.; Wang, T.; and Lu, W. 2024. TinyLLaMA: An open-source small language model. *arXiv preprint arXiv:2401.02385*.