AHAMask: Reliable Task Specification for Large Audio Language Models Without Instructions

Yiwei Guo[1], Bohan Li[1], Hankun Wang[1], Zhihan Li[1], Shuai Wang[2], Xie Chen[1], Kai Yu[1*]
[1]X-LANCE Lab, School of Computer Science, Shanghai Jiao Tong University, China
MoE Key Lab of Artificial Intelligence, Jiangsu Key Lab of Language Computing, China
[2]School of Intelligence Science and Technology, Nanjing University, China
{yiwei.guo, kai.yu}@sjtu.edu.cn

Abstract

Although current large audio language models (LALMs) extend text large language models (LLMs) with generic acoustic understanding abilities, they usually suffer from prompt sensitivity, where different instructions of the same intention can yield drastically different outcomes. In this work, we propose AHAMask, where we simply mask some of the attention heads in the decoder-only LLM backbone of LALMs, to trigger specific acoustic task functionalities without instructions. These masks are efficiently obtained by training on an LALM, with the number of trainable parameters equal to the attention head count in its LLM backbone. We show by experiments that applying such selective attention head masks achieves comparable or even better performance than using instructions, either on single or composite tasks. Besides achieving reliable acoustic task specification for LALMs, this also reveals that LALMs exhibit certain "functional pathways" in their attention heads.

Introduction

The rise of large language models (LLMs) has brought a revolution in the speech and audio processing domain, resulting in a mature paradigm of large audio language models (LALMs) (Peng et al. 2024; Su et al. 2025; Yang, Ho, and Lee 2025; Arora et al. 2025). In this work, we consider LALMs which are Transformer-based (Vaswani et al. 2017) models that can generate textual response to audio inputs and instructions. They extend text-based LLMs to "hear" and further understand audio input, typically with an audio encoder as the acoustic sensory receptor, and an LLM as the "brain" to process and response according to the audio input.

A prominent advantage of these LALMs is their unified framework of processing all kinds of audio information. Here, instructions play a key role in specifying the task of such LALMs. With a unified surface, LALMs can give answers to different audio traits, e.g. spoken content, speaker gender, identity, emotion, or even complex audio and music events, only by changing the instructions.

However, the flexibility of natural language also brings the risk of **prompt sensitivity**. Even targeted at a specific task, LALMs can still exhibit a large degree of sensitivity to provided instructions, even when these instructions have identical task requirements but only differ in linguistic forms or template specifications. This sensitivity poses a significant challenge for LALMs, impacting both their ability to accurately follow instructions and maintain consistent task performance (Peng et al. 2024). This problem is also a consensus in the broader LLM research field. However, for LALMs, limited effort has been devoted to benchmarking prompt sensitivity, let alone addressing it.

Since instructions are expected to control the functionality of LALMs, we choose to directly control the functionality inside the Transformer (Vaswani et al. 2017) structure of LALMs instead. This relates to the general explainability and functional partitioning of decoder-only Transformers, which have been a trending research topic for text LLMs. Recently, Han et al. (2025) finds that simply masking certain attention heads in LLMs leads to specific task functionalities without instructions. This selective attention head masking mechanism does not apply any modification to the model parameters, revealing the fundamental existence of "functional pathway" among the attention heads. However, this discovery is limited only to narrow applications in the text modality instead of multi-modal LLMs like LALMs. The acoustic functionalities in LALMs are fundamentally distinct from those of text LLMs, requiring a focus on multimodal alignment between continuous signals and discrete textual spaces, together with the comprehension of paralinguistic and non-linguistic information. Therefore, it is worth investigating whether similar properties exist in LALMs.

Inspired by the previous works in text-based LLMs, we apply the method in Han et al. (2025) and verify in this study that the "acoustic functional pathways" also exist in LALMs' attention heads. Thus, we propose AHAMask (Acoustic Attention Head Mask)[1] to achieve reliable task specification for LALMs, simply by masking some attention heads. Fig. 1 depicts the difference between a typical LALM with instructions and with only AHAMask. Importantly, the parameter count of our method is only equal to the number of attention heads in LALM's decoder-only LLM backbone, which is even **magnitudes smaller then previous parameter-efficient fine-tuning methods** (1-2k pa-

*Corresponding author.

[1]Code link: github.com/X-LANCE/SALMONN-AHAMask

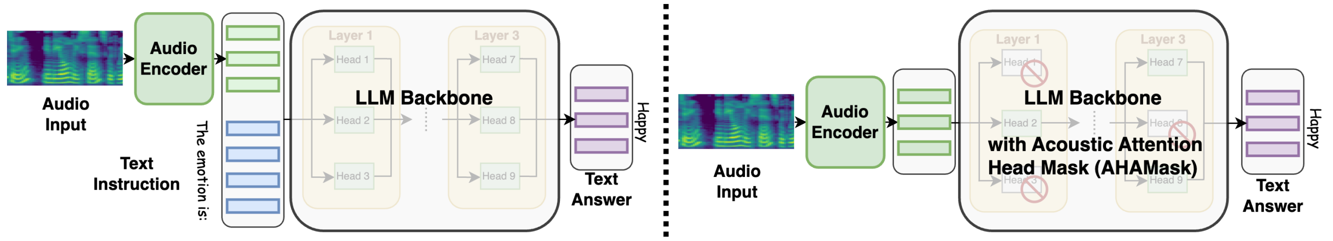

Figure 1: Diagram of a typical large audio language model. **Left**: the original model, which requires a text instruction to perform a specific task, but is sensitive to instructions. **Right**: the large audio language model with AHAMask (acoustic attention head mask), where only a set of attention heads are activated for a specific task. This frees the need for text instructions.

rameters vs. millions of parameters). This means training an AHAMask is highly efficient. As these masks are binary in inference stage, the storage overhead of such masks is also neglectable, e.g. 200 bytes for SALMONN (Tang et al. 2024). AHAMask distinguishes itself from existing fine-tuning approaches by **reducing** the effective parameter count during inference, rather than maintaining or increasing it. We conduct experiments across famous LALMs and classic audio and speech understanding tasks, including single and composite multi-hop tasks. Our findings suggest that:

- Simply masking some of the attention heads in the LLM backbone frees the need of instructions to specify a task in LALMs. These masks are intrinsic in the LALMs, not depending on the inputs.
- On most single auditory tasks, LALMs with AHAMask perform comparably or even better than LALMs with natural language instructions.
- On composite tasks where LALMs typically struggle, AHAMask can guide the model to adhere to task requirements in a much more effective way than natural language instructions.
- The masks across different tasks exhibit varying degrees of correlation; for instance, more similar tasks tend to involve a greater overlap in activated attention heads.
- Acoustic functionalities are formed gradually if we activate attention heads in order based on their importance weights. In other words, the acoustic functional pathways are constructed collectively by the heads.

Generally, AHAMask leverages the fundamental and intrinsic property of acoustic functional pathways in LALMs, and addresses the sensitivity of task specification in LALMs in a highly parameter-efficient way.

Background

Large Audio Language Models

Popular LALMs in the current literature usually make use of an audio encoder and a pretrained backbone text LLM (see Fig. 1 left). The audio encoders are self-supervised or supervised learning models to provide high-dimensional contextual representations of input audio signals. For example, the encoder of Whisper (Radford et al. 2023) and BEATs (Chen et al. 2023) are used and compressed together by Q-former (Li et al. 2023) in SALMONN (Tang et al. 2024) for speech and general audio input, respectively. These extracted embeddings are fed to an LLM backbone, which is typically initialized from a pretrained text LLM and finetuned on audio data, such as by LoRA (Hu et al. 2022). After training, the backbone LLM is able to understand and answer questions on acoustic information in various perspectives, such as spoken content, prosody, speaker traits, and even audio events. Recently, rapid progress has been made in this direction (Hu et al. 2024; Lu et al. 2024; Ding et al. 2025; Geng et al. 2025; Goel et al. 2025; Sakshi et al. 2025), making LALMs a successful and necessary component in artificial general intelligence.

LLM Instruction Following and Prompt Sensitivity

The prompt sensitivity issue in LLMs usually comes in the form of instruction non-following. For text-only LLMs, several benchmarks (Zeng et al. 2024; Zhou et al. 2023; Qin et al. 2024; Lou, Zhang, and Yin 2024; Wen et al. 2024) for evaluating instruction-following capabilities have been established. Optimization methods for improving instruction-following ability include steering vectors (Stolfo et al. 2025; He et al. 2025), supervised finetuning or reinforcement learning with sophisticated data pipelines (Xu et al. 2024; Dong et al. 2025; An et al. 2025; Agrawal et al. 2025). Additionally, prompt sensitivity is widely observed (Chatterjee et al. 2024; Sclar et al. 2024; Cao et al. 2024; Zhuo et al. 2024; Razavi et al. 2025); for instance, minor modifications such as altering punctuation marks alone can degrade LLM performance significantly (Sclar et al. 2024). In contrast, efforts to benchmark instruction-following ability and sensitivity in LALMs are ongoing but comparatively limited (Lu, Kuan, and yi Lee 2025; Gao et al. 2025).

Functional Partitioning of LLMs

In text-based LLMs, researchers have demonstrated that some Transformer modules exhibit certain functionality. For example, some Transformers layers can be swapped for cross lingual transfer (Bandarkar et al. 2025). It is also found that the neurons in feed forward networks (FFNs) store knowledge, such as language (Zeng et al. 2025), concepts (Rai and Yao 2024), and tasks (Xiao et al. 2024). Analysis on the attention heads also show that they encourage instruction following (Wu et al. 2024), transport a compact representation of tasks (Todd et al. 2024), mitigate knowledge conflicts (Jin et al. 2024), etc. Recently, Han et al.

Task	Metric	Training Data	Test Data
Single tasks			
ASR (Automatic Speech Recognition)	WER (Word Error Rate)	LibriSpeech	LibriSpeech test-clean & test-other
GR (Gender Recognition)	ACC (Accuracy)	LibriSpeech train-clean-100	LibriSpeech test-clean
SER (Speech Emotion Recognition)	ACC (Accuracy)	IEMOCAP Session1-4	IEMOCAP Session 5
ASV (Automatic Speaker Verification)	ACC (Accuracy)	VoxCeleb1	VoxCeleb1-O(cleaned)
AAC (Automatic Audio Captioning)	METEOR & ROUGE-L	AudioCaps	AudioCaps test
S2TT (Speech to Text Translation)	BLEU-4	CoVoST2 (en→zh)	CoVoST2 (en→zh) test
OSR (Overlapped Speech Recognition)	WER (Word Error Rate)	Libri2Mix mix-clean train-100 & train-360	Libri2Mix (mix-clean) test
Composite tasks			
GR | ASR, ASR | GR, {"ASR": , "GR":}	IFR (Instruction Following Rate), WER, ACC	LibriSpeech train-clean-100	LibriSpeech test-clean

Table 1: Audio understanding tasks, metrics, training data, and test data considered for LALMs in this paper.

(2025) finds that learning-based attention head selection can form specific task functional pathways in LLMs without instructions. These findings indicate LLMs possibly have some internal modularity. Note that this also has some similarity with studies in functional partitioning of the brain from neuroscience (Bertolero, Yeo, and D'Esposito 2015; Wig 2017). To the best of our knowldge, such partitioning and modularity have not been explored in LALMs so far.

AHAMask: Acoustic Attention Head Mask

Since the backbone LLMs act as "brains" in LALMs, we follow Han et al. (2025) and seek for acoustic functional pathways inside those LLMs. Modern LLMs are decoder-only Transformers, where each layer consists of multi-head attention (MHA), normalization, feed forward network and skip connections. Denote $\mathbf{X} \in \mathbb{R}^{l\times d}$ as the input sequence with length l and dimension d, the attention operation at the i-th layer and j-th attention head is defined as

$$\mathbf{Y}^{(i,j)} = \mathrm{Softmax}\left(\frac{\mathbf{X}\mathbf{W}_Q^{(i,j)}\left(\mathbf{X}\mathbf{W}_K^{(i,j)}\right)^T}{\sqrt{d_{\text{head}}}}\right)\mathbf{X}\mathbf{W}_V^{(i,j)}, \tag{1}$$

where $\mathbf{W}_Q^{(i,j)}, \mathbf{W}_K^{(i,j)}, \mathbf{W}_V^{(i,j)}$ are query, key and value projections with shape $d \times d_{\text{head}}$, respectively. The output of an MHA module is then the weighted sum of all attention head outputs in that layer, i.e. $\mathrm{MHA}_i(\mathbf{X}) = \sum_{j=1}^{h} \mathbf{Y}^{(i,j)}\mathbf{W}_O^{(i,j)}$ where $\mathbf{W}_O^{(i,j)} \in \mathbb{R}^{d_{\text{head}}\times d}$ is the output projection. These MHA modules are the most important part of a Transformer, and have been a long-standing research topic in LLMs (Vaswani et al. 2017; Lin et al. 2022; Zhao et al. 2023, 2024; Zheng et al. 2024).

As a single attention head is the basic granularity in the MHA operation, we try to find out the set of attention heads that can trigger a specific functionality. Let $\mathcal{M} \in \{0,1\}^{n\times h}$ be a set of attention head mask indicators, where n, h are the number of layers in the decoder-only Transformer and the number of heads in each layer, respectively. Denote $m_{i,j} \in \mathcal{M}$ as the binary indicator of the i-th layer and j-th attention head in that layer. We now modify the MHA operation to

$$\mathrm{MHA}_i(\mathbf{X}, \mathcal{M}) = \sum_{j=1}^{h} m_{i,j}\mathbf{Y}^{(i,j)}\mathbf{W}_O^{(i,j)}. \tag{2}$$

In other words, the attention heads are activated when $m_{i,j} \in \mathcal{M}$ is 1, or masked vice versa. Note that due to the existence of skip connections, the computation graph won't be cut off even if all the $m_{i,j}$ in the i-th layer are 0.

With this notation, we can treat $\mathcal{M}$ as the trainable parameters, and use Eq.(2) to train an AHAMask. As $\mathcal{M}$ is a discrete variable, we use Gumbel-Sigmoid (Jang, Gu, and Poole 2017; Geng et al. 2020; Han et al. 2025) for gradient estimation. Specifically, let $\mathbf{M} \in \mathbb{R}^{n\times h}$ be the mask logits of all heads. In the forward process, the discrete mask $\mathcal{M}$ is obtained by

$$\mathbf{S} = \sigma\left(\frac{\mathbf{M}+\mathbf{G}}{\tau}\right),\ \ \mathcal{M} = \mathbb{I}\left(\mathbf{S} \geq 0.5\right) \tag{3}$$

where $\sigma(\cdot)$ is the sigmoid function, $\mathbb{I}(\cdot)$ is the indicator function, and $\mathbf{G} \in \mathbb{R}^{n\times h}$ is Gumbel noise sampled by $g_k = -\log(-\log\epsilon_k), \forall g_k \in \mathbf{G}$ using $\epsilon_k \overset{\text{i.i.d}}{\sim} \mathrm{Uniform}(0,1)$. In the backward pass, gradients on $\mathcal{M}$ (the hard mask) is grafted to $\mathbf{S}$ (the soft mask), using straight-through estimator (Bengio, Léonard, and Courville 2013). Scalar $\tau > 0$ is a temperature hyperparameter used to control the sharpness of Gumbel approximation. Therefore, we only train the mask logits $\mathbf{M}$ using gradient descent. In inference, we obtain the discrete mask as $\mathcal{M} = \mathbb{I}(\mathbf{M} \geq 0)$. Also, $\mathbf{M}$ assigns each attention head an importance weight, which can be utilized to explore the gradual formation of functional pathways. Typically, with approximately 3 to 15 billion parameters, LALMs contain only a few thousand attention heads within their LLM backbone. This means the number of trainable parameters in AHAMask remains minimal, and the training process is highly efficient.

We train AHAMask on specific downstream tasks without instructions. Formally, for a given task $\mathcal{T}$, the training dataset can be represented as $\mathcal{D}_\mathcal{T} = \{..., (\texttt{Audio}_k, \texttt{Text}_k), ...\}$, where $\texttt{Audio}_k$ is an audio clip input, and $\texttt{Text}_k$ represents the corresponding textual re-

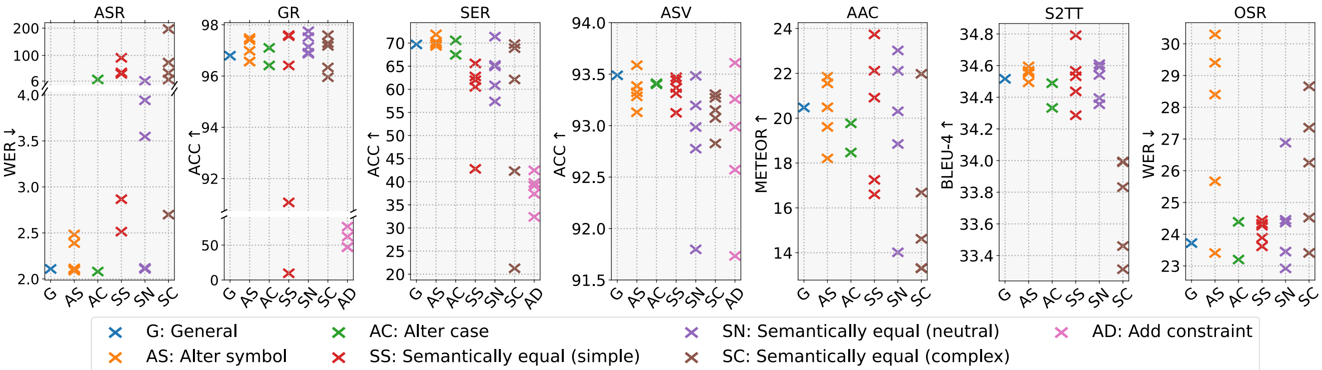

Figure 2: Prompt sensitivity experiments on SALMONN. Different colors and columns denote different types of variations.

sponse for the task. The training objective employs the standard cross entropy loss $\mathcal{L}_{\text{CE}}$ for next-token prediction applied to the tokens within $\texttt{Text}_k$. Through this training paradigm, we anticipate that $\mathcal{M}$ will effectively identify the subset of attention heads associated with specific task functionalities within the LALM, without any instructions.

Experiments

Experimental Setup

LALM Models In this work, we conduct studies on three well-known open-sourced LALMs: SALMONN (Tang et al. 2024), Qwen2Audio-Instruct, and Qwen2Audio-Base (Chu et al. 2024). SALMONN incorporates Whisper and BEATs as acoustic encoders, and vicuna v1.1 (Chiang et al. 2023) 13B as its backbone LLM. Each of the two variants of Qwen2Audio use Whisper-style encoders and Qwen-7B (Bai et al. 2023) as their backbone LLM. Qwen2Audio-Instruct is a supervised fine-tuned outcome of Qwen2Audio-Base for better instruction following ability. All the three models have exhibited good audio understanding abilities in various tasks as general LALMs (Sakshi et al. 2025).

Tasks and Datasets To discover atomic acoustic functionalities, we consider well-established benchmark tasks covering different aspects of audio understanding. These include automatic speech recognition (ASR) and gender recognition (GR) with LibriSpeech (Panayotov et al. 2015), speech emotion recognition (SER) with IEMOCAP (Busso et al. 2008), automatic speaker verification (ASV) with VoxCeleb1 (Nagrani, Chung, and Zisserman 2017), automatic audio captioning (AAC) with AudioCaps (Kim et al. 2019), speech to text translation (S2TT) with CoVoST2 (Wang, Wu, and Pino 2020), and overlapped speech recognition (OSR) with Libri2Mix (Cosentino et al. 2020). In GR, SER and ASV, LALMs are required to perform classification tasks , e.g. *Male* or *Female* in GR, and *Yes* or *No* in ASV. In ASV, two speech utterances are concatenated with a 0.1s silence as a single input. OSR requires LALMs to transcribe two overlapped speech utterances in any order. In S2TT and AAC, LALMs should apply more complex abilities to understand the audio input, and output translations or captions. These tasks span across purely linguistic (as ASR, OSR, S2TT), paralinguistic (SER, GR, ASV) to even general acoustic (AAC) understanding capabilities.

Moreover, we design composite tasks that require LALMs to perform multi-hop tasks in a specific format. In this study, we compose ASR and GR tasks and ask LALMs to perform them in order with separator symbol "`|`", or in json format. We will show that LALMs usually fail on these tasks even with clear natural language instructions, while applying AHAMask can introduce significant performance boosts. Table 1 lists all the tasks, metrics, and datasets used.

Training Details We initialize all the head logits in $\mathbf{M}$ by Gaussian $\mathcal{N}(4, 0.02)$ so that all heads are activated at first. We apply an annealing schedule on Gumbel temperature τ so that it starts from 4.0, linearly decreases to 0.5 at 3k steps, and remains constant thereafter. The learning rate is linearly warmed up from 1e-6 to 1e-2 within the first 3k steps, and decreased to a minimum of 1e-4 using cosine schedule afterwards. All the original parameters from LALM are frozen, and $\mathbf{M}$ is the only trainable parameter. All trainings are done on a single 65G Ascend 910B NPU device.

Prompt Sensitivity in LALMs

Peng et al. (2024) point out that speech LLMs exhibit non-negligible sensitivity to linguistic variations in the prompts. Following Peng et al. (2024) and Sclar et al. (2024), we present more evidence to showcase this issue for general LALMs. For each task, we first write a **general** instruction, and then apply one of the following variations:

- **Alter symbol** and **Alter case**, where we only change punctuation marks or letter casings.
- **Semantically equal (simple, neutral, complex)**, where the general prompt is rewritten with the same semantic meaning but in short, medium, or long sentences. This is done by providing Gemini 2.5 Flash with manual rephrasing examples and asking it to generate more.
- When the task is classification (GR, SER, ASV), we also apply **Add constraint** where the set of possible classes are provided in the prompt. For example, "Happy, Sad, Angry, or Neutral?" for SER.

For **Alter symbol**, **Semantically equal** and **Add constraint** variations, we construct 5 different prompts in each type. We

Task	ASR		GR	SER	ASV	AAC		S2TT	OSR
Metric	**WER↓**		**ACC↑**	**ACC↑**	**ACC↑**	**METEOR↑**	**ROUGE-L↑**	**BLEU-4↑**	**WER↓**
SALMONN, *which has **1600** attention heads in the LLM backbone.*									
w/ Instruction	**2.10**	**4.95**	96.79	69.70	**93.49**	20.60	40.42	**34.48**	**23.72**
w/o Instruction	12.00	17.23	0.00	0.00	0.00	14.90	33.69	15.14	30.95
w/ random mask	21.36	57.62	0.00	0.00	0.18	14.01	32.19	13.64	47.19
w/ AHAMask (no instruction)	**2.10**	5.08	**98.05**	**70.02**	93.24	**24.15**	**48.71**	33.90	23.89
# activated heads in AHAMask	1525	1525	1259	1146	1311	1426		1551	1550
Qwen2Audio-Instruct, *which has **1024** attention heads in the LLM backbone.*									
w/ Instruction	3.65	6.58	91.03	64.46	79.02	16.02	30.79	37.30	46.96
w/o Instruction	217.96	234.78	0.00	0.16	0.00	12.94	26.16	0.49	85.39
w/ random mask	164.67	200.66	0.00	0.00	0.00	13.76	28.23	0.63	87.38
w/ AHAMask (no instruction)	**2.99**	**5.65**	**94.43**	**70.99**	**82.34**	**22.96**	**46.03**	**39.02**	**46.06**
w/ AHAMask of the Base model	156.87	169.32	0.00	10.23	0.02	17.53	33.80	5.89	62.28
# activated heads in AHAMask	653	649	565	735	527	753		741	721
Qwen2Audio-Base, *which has **1024** attention heads in the LLM backbone.*									
w/ Instruction	**1.77**	**3.93**	72.37	56.33	49.24	12.33	27.39	**45.00**	53.66
w/o Instruction	99.86	99.59	0.00	0.00	0.00	1.19	1.79	0.21	94.87
w/ random mask	192.90	127.59	0.00	0.00	0.00	1.70	2.92	0.62	89.50
w/ AHAMask (no instruction)	3.11	6.03	**89.24**	**57.05**	**85.75**	**21.35**	**44.89**	39.27	**38.37**
w/ AHAMask of the Instruct model	252.96	350.77	0.00	0.00	0.00	2.99	5.41	12.73	64.55
# activated heads in AHAMask	726	724	633	679	797	729		769	774

Table 2: Single task performance of SALMONN, Qwen2Audio-Instruct, and Qwen2Audio-Base. WERs in the ASR task are formatted as LibriSpeech "test-clean | test-other". When applying AHAMask, no instruction is given. When applying a random mask, we ensure the number of activated heads is the same with the corresponding AHAMask in that task. Note that the number of trainable parameters in AHAMask is the same as number of attention heads in the LLM backbone.

take SALMONN as a typical LALM here. Notably, all 7 individual tasks here are comprehensively covered within the training data of SALMONN. Our preliminary experiments indicate that SALMONN demonstrates greater robustness to variations in instructions compared to Qwen2Audio. For decoding, we use deterministic beam search with beam size 4, so that the difference in model performance is only caused by different instructions.

The performance with instruction variations are visualized in Fig. 2. These results clearly indicate that SALMONN still has high sensitivity to instructions with the same intention. For example, in ASR task, changing the prompt to all capital letters will cause more hallucinations (repeating or generating upper case phonemes instead of words), thus raising the WER to 12%. In GR task, adding the constraint "Male or Female" makes the model almost always output "Male". Similar phenomenon can also be observed with SER. For SALMONN, ASV and S2TT tasks exhibit better robustness to instruction variations. In almost all cases, using longer and more complex instructions is likely to cause a degradation in performance, sometimes even drastically as in ASR and SER. These results exemplify that prompt sensitivity remains a significant challenge in LALMs.

AHAMask for Single Tasks

We apply AHAMask on SALMONN and Qwen2Audio variants, on all the 7 single tasks. We compare LALM performance with AHAMask against three other methods: ① with clear instructions from the "general" type in the previous section, ② without instructions, and ③ using a random mask with the same number of activated heads as AHAMask for each task. For Qwen2Audio, since the two variants are identical in structure, we also evaluate ④ swapped AHAMask across the two variants.

The results in Table 2 indicate several insights:

- LALMs with only certain attention heads activated can achieve comparable or even better performance than using natural language instructions on all of the 7 single audio understanding tasks. Even though LALMs with natural language instructions sometimes still outperform those with only AHAMask, the difference in performance is very small (e.g. on ASV, S2TT and OSR tasks with SALMONN). Note that AHAMask is only supposed to reliably **manifest** specific functionalities instead of **improving** them, so these are still expected and supporting evidence. Only on ASR and S2TT with Qwen2Audio-Base, which are the pretraining tasks of this model, using instructions achieves noticeably better performance compared to AHAMask.
- Even on tasks where the original LALM struggles to perform well with explicit instructions, applying AHAMask can trigger its functionality to perform much better. This is especially evident for Qwen2Audio-Base, which cannot perform well on tasks that are covered by little or no data in the pretraining phase. Applying AHAMask on

Task	GR+"｜"+ASR			ASR+"｜"+GR			Json style {"ASR": , "GR": }		
Metric	IFR↑	ACC↑	WER↓	IFR↑	WER↓	ACC↑	IFR↑	WER↓	ACC↑
SALMONN, *which has **1600** attention heads in the LLM backbone.*									
w/ Instruction	98.59	68.02	3.52	16.03	29.36	45.95	69.16	6.17	51.05
w/ AHAMask (no instruction)	**99.12**	**97.77**	**2.21**	**97.63**	**2.29**	**97.81**	**98.89**	**2.40**	**97.30**
# activated heads in AHAMask	1315			1252			1124		
Qwen2Audio-Instruct, *which has **1024** attention heads in the LLM backbone.*									
w/ Instruction	47.44	68.79	7.03	79.16	50.34	53.09	0.00	-	-
w/ AHAMask (no instruction)	**94.62**	**91.09**	**3.08**	**89.92**	**3.22**	**89.56**	**58.45**	**3.87**	**89.16**
# activated heads in AHAMask	555			558			522		
Qwen2Audio-Base, *which has **1024** attention heads in the LLM backbone.*									
w/ Instruction	13.74	1.11	58.87	10.92	68.75	4.20	0.00	-	-
w/ AHAMask (no instruction)	**82.40**	**64.94**	**3.25**	**33.93**	**6.51**	**58.83**	**71.19**	**2.79**	**87.46**
# activated heads in AHAMask	641			599			616		

Table 3: Composite task performance of SALMONN, Qwen2Audio-Instruct, and Qwen2Audio-Base. IFR denotes instruction following rate (%). Task-specific metrics are calculated among instruction-following samples. Note that the trainable parameter count in AHAMask is the same as number of attention heads in the LLM backbone.

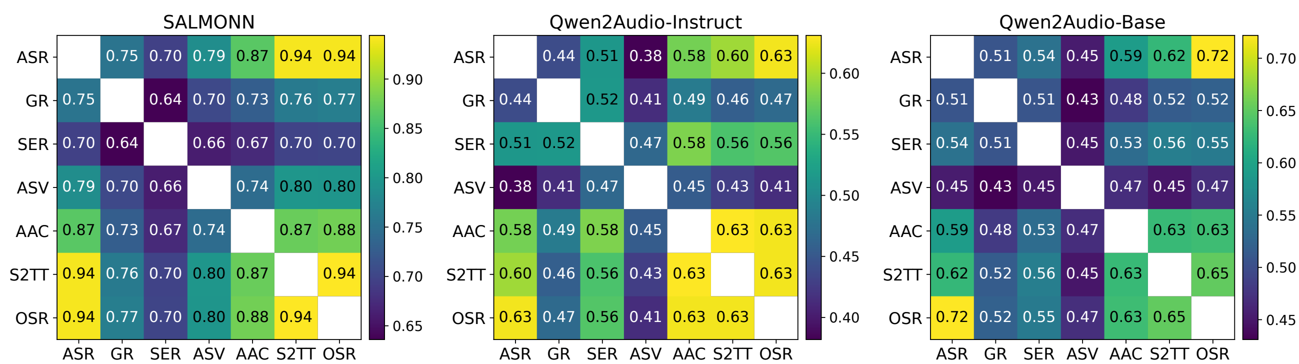

Figure 3: Jaccard similarities of AHAMask between different tasks in each LALM.

this base model significantly improves its performance on GR, ASV, ACC and OSR to match or even surpass the instructed model Qwen2Audio-Instruct.

- An AHAMask is only valid at specific head positions; randomly shuffling it will fail, as indicated by the "w/ random mask" results.
- An AHAMask is only valid for the model it is trained on; swapping from different variants under the same model family will not work, as for Qwen2Audio.
- It does not require dropping many attention heads to achieve reliable task specification. For example, only masking 75 and 49 out of 1600 heads is enough for SALMONN to perform ASR and S2TT without instructions, respectively.
- The number of activated heads in AHAMask seems to correlate with task complexity. For simpler classification tasks like SER, GR and ASV, it usually requires fewer heads; for other sequence generation tasks, more attention heads are activated. However, the base model Qwen2Audio-Base exhibits different characteristics compared to Qwen2Audio-Instruct on this property.

AHAMask for Composite Tasks

From the previous section, it is clear that LALMs can perform single acoustic tasks without instructions once equipped with task-specific AHAMasks. In this section, we evaluate the effect of AHAMask on more complex tasks. We design composite multi-hop tasks that require LALMs to generate answers in a specific format, hence also bring the metric of instruction following rate (IFR). Here, we consider asking the model to perform two tasks in order with a separator "｜", or in json format with keys "ASR" and "GR". IFR is computed as the percentage where the generated answers can be divided into two parts by "｜", or parsed by json. We combine the ASR and GR tasks as they are largely independent of each other, thus require LALMs to attend to different acoustic aspects. We use LibriSpeech for this experiment.

We present the results in Table 3. In all the metrics, applying AHAMask achieves better performance than using instructions, especially for Qwen2Audio-Base where it is hard to instruct the model well. Also, LALMs can be sen-

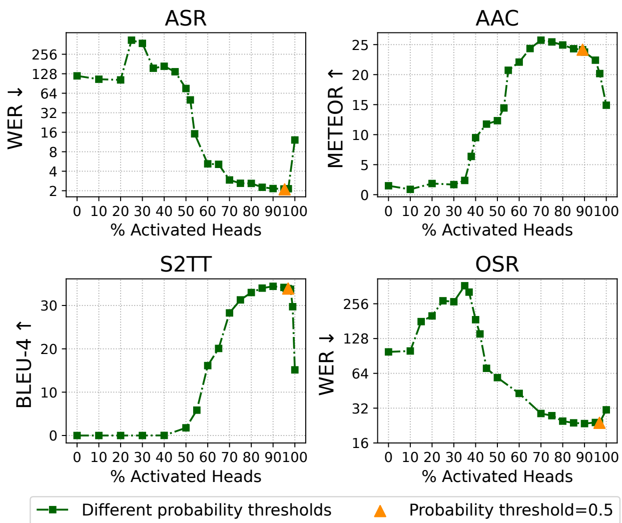

Figure 4: Performance of SALMONN on 4 non-classification tasks with AHAMask in different percentage of activated attention heads. The orange triangle marker denotes the metric in Table 2.

sitive to the ordering of sub-tasks in such composite tasks, e.g. the original SALMONN model can generate answers in the format of "GR | ASR" well, but fails in the opposite ordering. Even if the specific format is satisfied, the performance in each sub-task is not as good as when only doing single tasks. For SALMONN and Qwen2Audio-Instruct, the gap between different task orders is largely reduced using AHAMask, and the performance of each sub-task is closer to the single task scenarios. These results further indicate that AHAMask can control more complex behavior of LALMs in processing multiple aspects of acoustic information.

Analysis of Acoustic Attention Head Masks

The Similarity of Masks for Different Tasks Intuitively, the relation between the sets of activated attention heads should be correlated to the corresponding tasks. More similar tasks should come with more common attention heads. From previous results, we can already observe that the number of attention heads has a correlation with task complexity. In this section, we visualize the similarity of AHAMask between different tasks in all the three LALMs. We consider Jaccard similarity to measure the similarity of two boolean sets: $J(\mathcal{M}_1, \mathcal{M}_2) = |\mathcal{M}_1 \cap \mathcal{M}_2| / |\mathcal{M}_1 \cup \mathcal{M}_2|$.

Fig.3 visualizes the Jaccard similarity matrices. As expected, tasks that are intuitively similar share more attention heads. For example, AHAMasks for OSR and ASR tasks are the most similar in all three LALMs, while paralinguistic and classification tasks like SER and ASV usually have lower similarity with others. Interestingly, the AAC task also has good similarity in AHAMask with ASR, S2TT and OSR tasks. The similarity matrices for different LALMs also exhibit different patterns. These mask similarities very likely stem from LALMs' internal mechanisms, which may inspire future work on interpreting their behaviors.

# Activated Heads	Output	Output Type
320 (20%) 847 (52.9375%)	at 0) - 0 0) 0) 0) 0 & *(empty string)*	Meaningless Strings
848 (53%) 862 (53.875%) 863 (53.9375%) 872 (54.5%)	Happy Happy Neutral Neutral	Incorrect Emotions
873 (54.5625%) 1429 (89.3125%)	Sad Sad	Correct Emotions
1430 (89.375%) 1579 (98.6875%) 1580 (98.75%) 1593 (99.5625%)	Someone is crying. Someone is crying. I know. I know, I know.	Captions or Transcriptions

Table 4: The output of SALMONN on one example in the SER task, with different number of activated heads.

Attention Heads Gradually Form Acoustic Functionalities Since we essentially train a logit matrix $\mathbf{M}$, each attention head is associated with a continuous mask probability in $\sigma(\mathbf{M})$. We can then apply different thresholds to these probabilities to obtain masks with different percentage of activated attention heads. This will indicate if there are gradual or abrupt changes in the performance when we activate the attention heads in order. In other words, here we explore how the acoustic functional pathways of LALMs are formed by the attention heads.

To achieve a desired activation percentage q of heads, we mask the attention heads whose corresponding logits in $\mathbf{M}$ fall within the lowest $1-q$ quantile. We show the performance with different percentage levels on all non-classification tasks for SALMONN in Fig.4, as these metrics are more fine-grained than classification accuracy. It is thus clear that many attention heads have an incremental improvement on the task functionality, and the optimal metrics are obtained based on collective efforts. For classification tasks, we also show in Table 4 an example of SALMONN changing behaviors with different number of activated heads. These results indicate that LALM's acoustic functional pathway is gradually instead of suddenly formed based on the importance of heads, which provides additional insights to studying the explainability of such models.

Conclusion

In this work, we propose AHAMask that simply masks some attention heads in an LALM to address the prompt sensitivity problem and achieve reliable task specification without instructions. These masks are efficiently obtained through a lightweight training process. We verify by experiments that applying AHAMask can achieve comparable or even better performance than using general instructions, in diverse scenarios. Our findings also suggest that acoustic "functional pathways" do exist in the attention heads of LALMs, which can be an important inspiration for the interpretability of such models. Future research includes analyzing the partitioning and composability of AHAMask and constructing a general text-to-mask converter for more use cases.

Acknowledgements

This work is supported by Open Project of the Key Laboratory of Artificial Intelligence, Ministry of Education (No. AI202405), and the China NSFC Project (No. 92370206).

References

Agrawal, A.; Alazraki, L.; Honarvar, S.; Mensink, T.; and Rei, M. 2025. Enhancing LLM Robustness to Perturbed Instructions: An Empirical Study. In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*.

An, K.; Sheng, L.; Cui, G.; Si, S.; Ding, N.; Cheng, Y.; and Chang, B. 2025. UltraIF: Advancing Instruction Following from the Wild. In *Proceedings of EMNLP*, 18722–18737.

Arora, S.; Chang, K.-W.; Chien, C.-M.; Peng, Y.; Wu, H.; Adi, Y.; Dupoux, E.; Lee, H.-Y.; Livescu, K.; and Watanabe, S. 2025. On the landscape of spoken language models: A comprehensive survey. *arXiv preprint arXiv:2504.08528*.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; et al. 2023. Qwen Technical Report. *arXiv preprint arXiv:2309.16609*.

Bandarkar, L.; Muller, B.; Yuvraj, P.; Hou, R.; Singhal, N.; Lv, H.; and Liu, B. 2025. Layer Swapping for Zero-Shot Cross-Lingual Transfer in Large Language Models. In *Proc. ICLR 2025*.

Bengio, Y.; Léonard, N.; and Courville, A. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Bertolero, M. A.; Yeo, B. T.; and D'Esposito, M. 2015. The modular and integrative functional architecture of the human brain. *Proceedings of the National Academy of Sciences*, 112(49): E6798–E6807.

Busso, C.; Bulut, M.; Lee, C.; Kazemzadeh, A.; Mower, E.; Kim, S.; Chang, J. N.; Lee, S.; and Narayanan, S. S. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Lang. Resour. Evaluation*, 42(4): 335–359.

Cao, B.; Cai, D.; Zhang, Z.; Zou, Y.; and Lam, W. 2024. On the Worst Prompt Performance of Large Language Models. In *Advances in Neural Information Processing Systems*, volume 37, 69022–69042.

Chatterjee, A.; Renduchintala, H. S. V. N. S. K.; Bhatia, S.; and Chakraborty, T. 2024. POSIX: A Prompt Sensitivity Index For Large Language Models. In *Findings of EMNLP 2024*, 14550–14565.

Chen, S.; Wu, Y.; Wang, C.; Liu, S.; Tompkins, D.; Chen, Z.; Che, W.; Yu, X.; and Wei, F. 2023. BEATs: Audio Pre-Training with Acoustic Tokenizers. In *Proc. ICML 2023*, volume 202 of *Proceedings of Machine Learning Research*, 5178–5193. PMLR.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; et al. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Chu, Y.; Xu, J.; Yang, Q.; Wei, H.; Wei, X.; Guo, Z.; Leng, Y.; Lv, Y.; He, J.; Lin, J.; et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Cosentino, J.; Pariente, M.; Cornell, S.; Deleforge, A.; and Vincent, E. 2020. Librimix: An open-source dataset for generalizable speech separation. *arXiv preprint arXiv:2005.11262*.

Ding, D.; Ju, Z.; Leng, Y.; Liu, S.; Liu, T.; Shang, Z.; Shen, K.; Song, W.; Tan, X.; Tang, H.; et al. 2025. Kimi-audio technical report. *arXiv preprint arXiv:2504.18425*.

Dong, G.; Lu, K.; Li, C.; Xia, T.; Yu, B.; Zhou, C.; and Zhou, J. 2025. Self-play with Execution Feedback: Improving Instruction-following Capabilities of Large Language Models. In *Proc. ICLR 2025*.

Gao, Y.; Wang, B.; Wei, C.; Sun, S.; and Aw, A. 2025. IFEval-Audio: Benchmarking Instruction-Following Capability in Audio-based Large Language Models. *arXiv preprint arXiv: 2505.16774*.

Geng, X.; Wang, L.; Wang, X.; Qin, B.; Liu, T.; and Tu, Z. 2020. How Does Selective Mechanism Improve Self-Attention Networks? In *Proc. ACL 2020*, 2986–2995.

Geng, X.; Wei, K.; Shao, Q.; Liu, S.; Lin, Z.; Zhao, Z.; Li, G.; Tian, W.; Chen, P.; Li, Y.; et al. 2025. OSUM: Advancing open speech understanding models with limited resources in academia. *arXiv preprint arXiv:2501.13306*.

Goel, A.; Ghosh, S.; Kim, J.; Kumar, S.; Kong, Z.; Lee, S.-g.; et al. 2025. Audio Flamingo 3: Advancing Audio Intelligence with Fully Open Large Audio Language Models. *arXiv preprint arXiv:2507.08128*.

Han, S.; Zeng, H.; Yu, K.; and Chen, L. 2025. Heads up! Large Language Models Can Perform Tasks Without Your Instruction via Selective Attention Head Masking. In *Forty-second International Conference on Machine Learning*.

He, Z.; Zhao, H.; Qiao, Y.; Yang, F.; Payani, A.; Ma, J.; and Du, M. 2025. SAIF: A Sparse Autoencoder Framework for Interpreting and Steering Instruction Following of Language Models. arXiv:2502.11356.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Hu, S.; Zhou, L.; Liu, S.; Chen, S.; Meng, L.; et al. 2024. WavLLM: Towards Robust and Adaptive Speech Large Language Model. In *Findings of EMNLP 2024*, 4552–4572.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. In *5th International Conference on Learning Representations, ICLR 2017*.

Jin, Z.; Cao, P.; Yuan, H.; Chen, Y.; Xu, J.; Li, H.; Jiang, X.; Liu, K.; and Zhao, J. 2024. Cutting Off the Head Ends the Conflict: A Mechanism for Interpreting and Mitigating Knowledge Conflicts in Language Models. In *Findings of ACL 2024*, 1193–1215.

Kim, C. D.; Kim, B.; Lee, H.; and Kim, G. 2019. AudioCaps: Generating captions for audios in the wild. In *Proc. NAACL: Human Language Technologies, Volume 1 (Long and Short Papers)*, 119–132.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Lin, T.; Wang, Y.; Liu, X.; and Qiu, X. 2022. A survey of transformers. *AI open*, 3: 111–132.

Lou, R.; Zhang, K.; and Yin, W. 2024. Large language model instruction following: A survey of progresses and challenges. *Computational Linguistics*, 50(3): 1053–1095.

Lu, K.-H.; Chen, Z.; Fu, S.-W.; Huang, H.; Ginsburg, B.; Wang, Y.-C. F.; and yi Lee, H. 2024. DeSTA: Enhancing Speech Language Models through Descriptive Speech-Text Alignment. In *Interspeech 2024*, 4159–4163.

Lu, K.-H.; Kuan, C.-Y.; and yi Lee, H. 2025. Speech-IFEval: Evaluating Instruction-Following and Quantifying Catastrophic Forgetting in Speech-Aware Language Models. In *Interspeech 2025*, 2078–2082.

Nagrani, A.; Chung, J. S.; and Zisserman, A. 2017. VoxCeleb: A Large-Scale Speaker Identification Dataset. In *Interspeech 2017*, 2616–2620.

Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: An ASR corpus based on public domain audio books. In *Proc. ICASSP 2015*, 5206–5210.

Peng, J.; Wang, Y.; Fang, Y.; Xi, Y.; Li, X.; Zhang, X.; and Yu, K. 2024. A survey on speech large language models. *arXiv preprint arXiv:2410.18908*.

Qin, Y.; Song, K.; Hu, Y.; Yao, W.; Cho, S.; Wang, X.; Wu, X.; Liu, F.; Liu, P.; and Yu, D. 2024. InFoBench: Evaluating Instruction Following Ability in Large Language Models. In *Findings of ACL 2024*, 13025–13048.

Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, 28492–28518. PMLR.

Rai, D.; and Yao, Z. 2024. An Investigation of Neuron Activation as a Unified Lens to Explain Chain-of-Thought Eliciting Arithmetic Reasoning of LLMs. In *Proc. ACL 2024*, 7174–7193.

Razavi, A.; Soltangheis, M.; Arabzadeh, N.; Salamat, S.; Zihayat, M.; and Bagheri, E. 2025. Benchmarking prompt sensitivity in large language models. In *European Conference on Information Retrieval*, 303–313. Springer.

Sakshi, S.; Tyagi, U.; Kumar, S.; Seth, A.; Selvakumar, R.; et al. 2025. MMAU: A Massive Multi-Task Audio Understanding and Reasoning Benchmark. In *Proc. ICLR 2025*.

Sclar, M.; Choi, Y.; Tsvetkov, Y.; and Suhr, A. 2024. Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I learned to start worrying about prompt formatting. In *Proc. ICLR 2024*.

Stolfo, A.; Balachandran, V.; Yousefi, S.; Horvitz, E.; and Nushi, B. 2025. Improving Instruction-Following in Language Models through Activation Steering. In *Proc. ICLR 2025*.

Su, Y.; Bai, J.; Xu, Q.; Xu, K.; and Dou, Y. 2025. Audio-Language Models for Audio-Centric Tasks: A survey. *arXiv preprint arXiv:2501.15177*.

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 2024*.

Todd, E.; Li, M.; Sharma, A. S.; Mueller, A.; Wallace, B. C.; and Bau, D. 2024. Function Vectors in Large Language Models. In *Proc. ICLR 2024*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Proc. NeurIPS*, 30.

Wang, C.; Wu, A.; and Pino, J. 2020. Covost 2 and massively multilingual speech-to-text translation. *arXiv preprint arXiv:2007.10310*.

Wen, B.; Ke, P.; Gu, X.; Wu, L.; et al. 2024. Benchmarking Complex Instruction-Following with Multiple Constraints Composition. In *Proc. NeurIPS 2024*, volume 37, 137610–137645.

Wig, G. S. 2017. Segregated systems of human brain networks. *Trends in cognitive sciences*, 21(12): 981–996.

Wu, X.; Yao, W.; Chen, J.; Pan, X.; Wang, X.; Liu, N.; and Yu, D. 2024. From Language Modeling to Instruction Following: Understanding the Behavior Shift in LLMs after Instruction Tuning. In *Proc. NAACL: Human Language Technologies (Volume 1: Long Papers)*, 2341–2369.

Xiao, C.; Zhang, Z.; Song, C.; Jiang, D.; Yao, F.; et al. 2024. Configurable foundation models: Building LLMs from a modular perspective. *arXiv preprint arXiv:2409.02877*.

Xu, C.; Sun, Q.; Zheng, K.; Geng, X.; Zhao, P.; Feng, J.; Tao, C.; Lin, Q.; and Jiang, D. 2024. WizardLM: Empowering Large Pre-Trained Language Models to Follow Complex Instructions. In *Proc. ICLR 2024*.

Yang, C.-K.; Ho, N. S.; and Lee, H.-y. 2025. Towards holistic evaluation of large audio-language models: A comprehensive survey. *arXiv preprint arXiv:2505.15957*.

Zeng, H.; Han, S.; Chen, L.; and Yu, K. 2025. Converging to a Lingua Franca: Evolution of Linguistic Regions and Semantics Alignment in Multilingual Large Language Models. In *Proc. ACL 2025*, 10602–10617.

Zeng, Z.; Yu, J.; Gao, T.; Meng, Y.; Goyal, T.; and Chen, D. 2024. Evaluating Large Language Models at Evaluating Instruction Following. In *Proc. ICLR 2024*.

Zhao, H.; Chen, H.; Yang, F.; Liu, N.; Deng, H.; Cai, H.; Wang, S.; Yin, D.; and Du, M. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2): 1–38.

Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).

Zheng, Z.; Wang, Y.; Huang, Y.; Song, S.; Yang, M.; Tang, B.; Xiong, F.; and Li, Z. 2024. Attention heads of large language models: A survey. *arXiv preprint arXiv:2409.03752*.

Zhou, J.; Lu, T.; Mishra, S.; Brahma, S.; Basu, S.; Luan, Y.; Zhou, D.; and Hou, L. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Zhuo, J.; Zhang, S.; Fang, X.; et al. 2024. ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs. In *Findings of EMNLP 2024*, 1950–1976.