Enhancing Speech Large Language Models with Prompt-Aware Mixture of Audio Encoders

EMNLP 2025 Main

Weiqiao Shan[1], Yuang Li[2*], Yuhao Zhang[3], Yingfeng Luo[1], Chen Xu[4], Xiaofeng Zhao[2], Long Meng[1], Yunfei Lu[2], Min Zhang[2], Hao Yang[2], Tong Xiao[1,5†], Jingbo Zhu[1,5]

[1]School of Computer Science and Engineering, Northeastern University, Shenyang, China

[2]Huawei Translation Services Center, Beijing, China

[3]The Chinese University of Hong Kong, Shenzhen, China

[4]College of Computer Science and Technology, Harbin Engineering University, Harbin, China

[5]NiuTrans Research, Shenyang, China

* Equal contribution

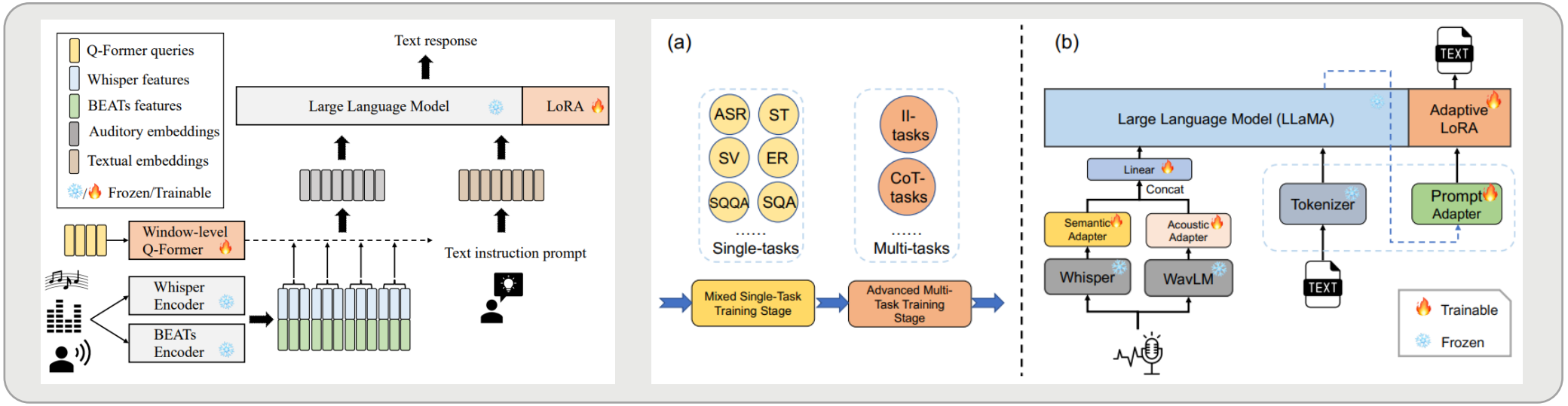
† Corresponding author

Background

- Different tasks may require distinct features
  - > Semantic
  - > Acoustic

The **Whisper model** is trained for speech recognition and translation based on a large amount of weakly supervised data, whose encoder output features **are suitable to model speech and include information about the background noises**. **BEATs** is trained to extract **high-level non-speech audio semantics information** using iterative self-supervised learning.

Speech LLM with multi-encoders



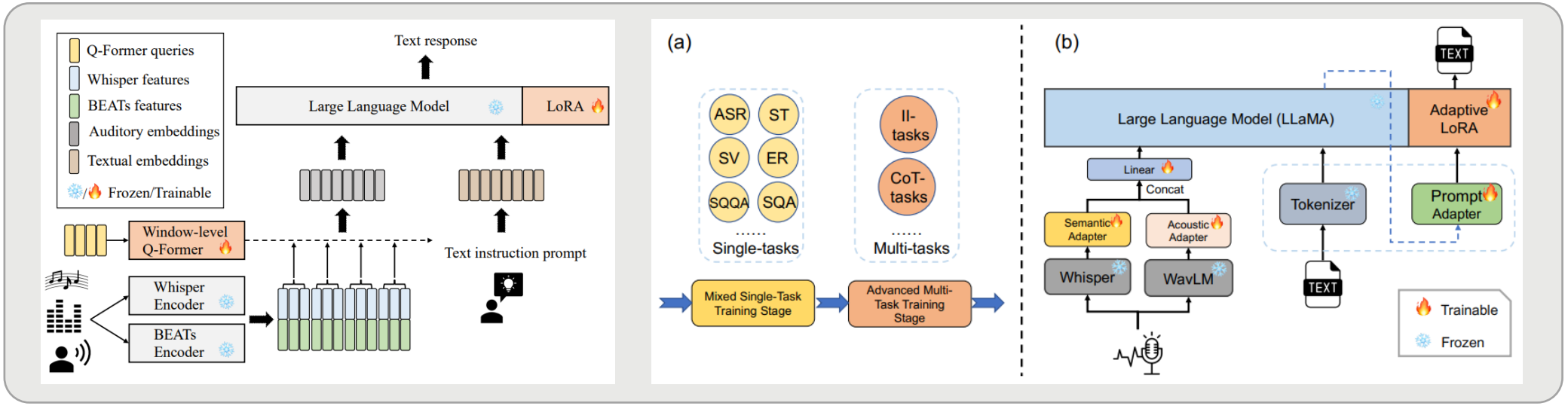
Hu S, Zhou L, Liu S, et al. Wavllm: Towards robust and adaptive speech large language model[J]. arXiv preprint arXiv:2404.00656, 2024.
Tang C, Yu W, Sun G, et al. Salmonn: Towards generic hearing abilities for large language models[J]. arXiv preprint arXiv:2310.13289, 2023.

Background

- Different tasks may require distinct features
  - > Semantic
  - > Acoustic

The **Whisper model** is trained for speech recognition and translation based on a large amount of weakly supervised data, whose encoder output features **are suitable to model speech and include information about the background noises**. **BEATs** is trained to extract **high-level non-speech audio semantics information** using iterative self-supervised learning.

Speech LLM with multi-encoders



Insight 1: A single audio encoder cannot sufficiently capture all the features demanded by multiple downstream tasks.

Hu S, Zhou L, Liu S, et al. Wavllm: Towards robust and adaptive speech large language model[J]. arXiv preprint arXiv:2404.00656, 2024.
Tang C, Yu W, Sun G, et al. Salmonn: Towards generic hearing abilities for large language models[J]. arXiv preprint arXiv:2310.13289, 2023.

Background

- Distinct feature requirements of different audio downstream tasks are manifested across different layers.

The weights of different on the different downstream tasks of the SUPERB bench

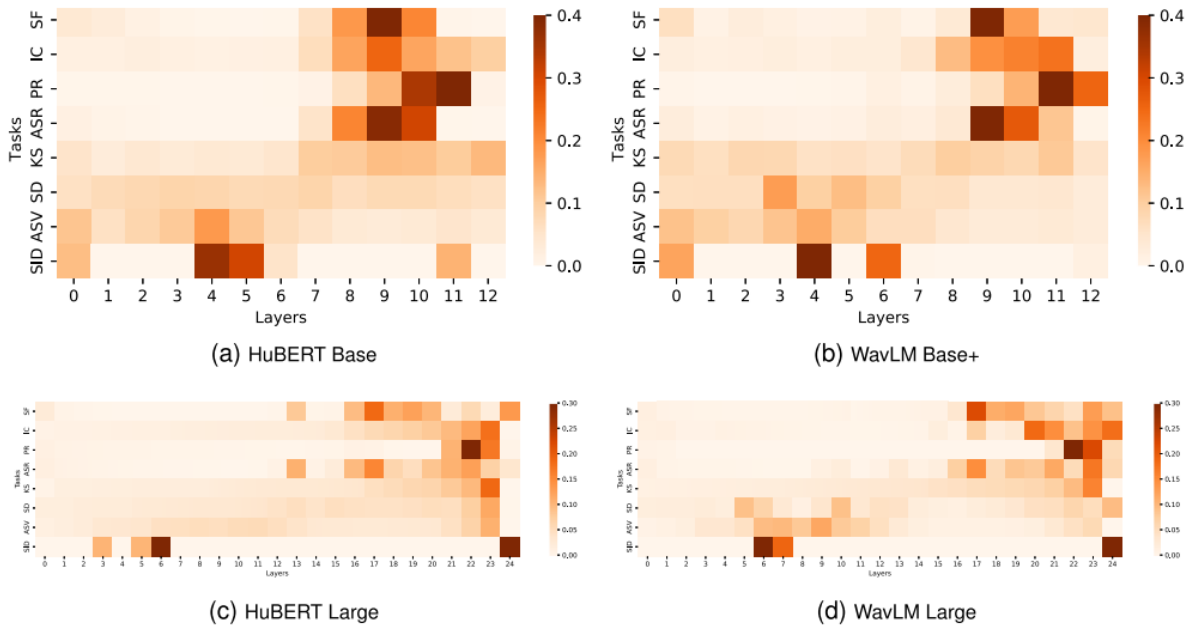


Fig. 2. Weight analysis on the SUPERB Benchmark. Layer 0 corresponds to the input of the first Transformer layer. The y-axis represents different tasks, while the x-axis represents different layers.

Speaker Identification (SID)

Automatic Speaker Verification (ASV)

Speaker Diarization (SD)

Phoneme Recognition (PR)

Automatic Speech Recognition (ASR)

Out-Of-Domain Automatic Speech Recognition (OODASR)

Keyword Spotting (KS)

Intent Classification (IC)

Slot Filling (SF)

Chen S, Wang C, Chen Z, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing[J]. IEEE Journal of Selected Topics in Signal Processing, 2022, 16(6): 1505-1518.

Background

- Distinct feature requirements of different audio downstream tasks are manifested across different layers.

The weights of different on the different downstream tasks of the SUPERB bench

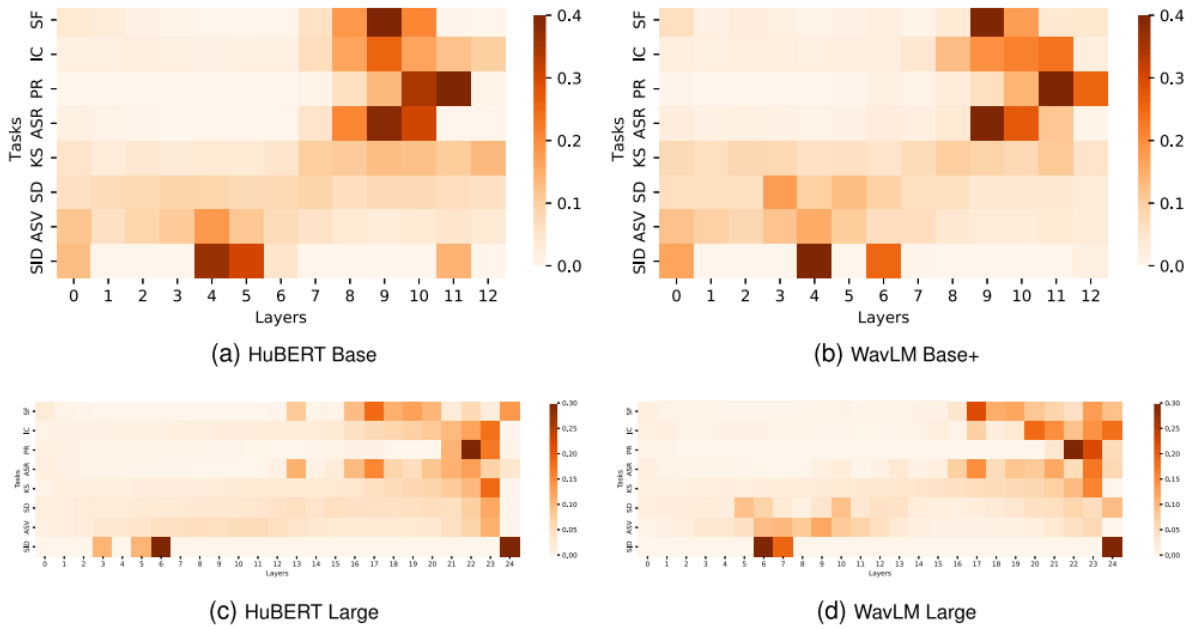


Fig. 2. Weight analysis on the SUPERB Benchmark. Layer 0 corresponds to the input of the first Transformer layer. The y-axis represents different tasks, while the x-axis represents different layers.

Speaker Identification (SID)

Automatic Speaker Verification (ASV)

Speaker Diarization (SD)

Phoneme Recognition (PR)

Automatic Speech Recognition (ASR)

Out-Of-Domain Automatic Speech Recognition (OODASR)

Keyword Spotting (KS)

Intent Classification (IC)

Slot Filling (SF)

Insight 2: Features extracted from different layers of the audio encoder contribute with varying importance to different tasks.

Chen S, Wang C, Chen Z, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing[J]. IEEE Journal of Selected Topics in Signal Processing, 2022, 16(6): 1505-1518.

Background

- Insight 1: A single audio encoder cannot sufficiently capture all the features demanded by multiple downstream tasks.

  > It is necessary to fuse multiple encoders to capture more comprehensive audio information

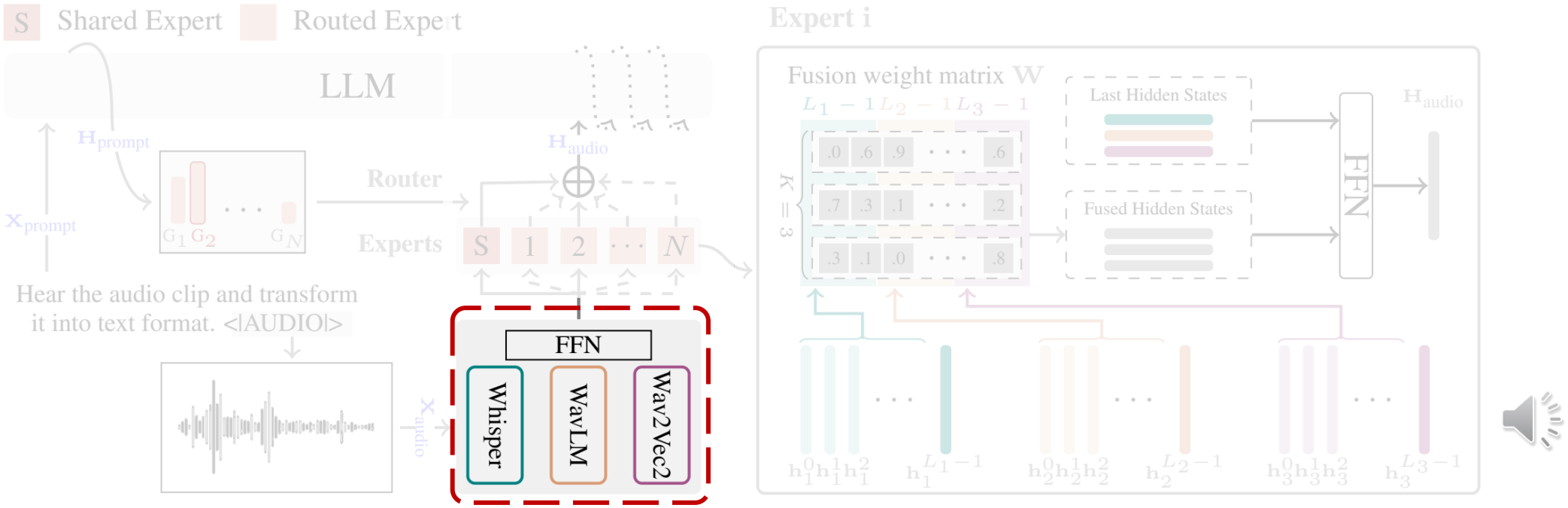
- Insight 2: Features extracted from different layers of the audio encoder contribute with varying importance to different tasks.

  > Need a more effective method to exploit the information from different layers of multiple encoders.

Method - Overview

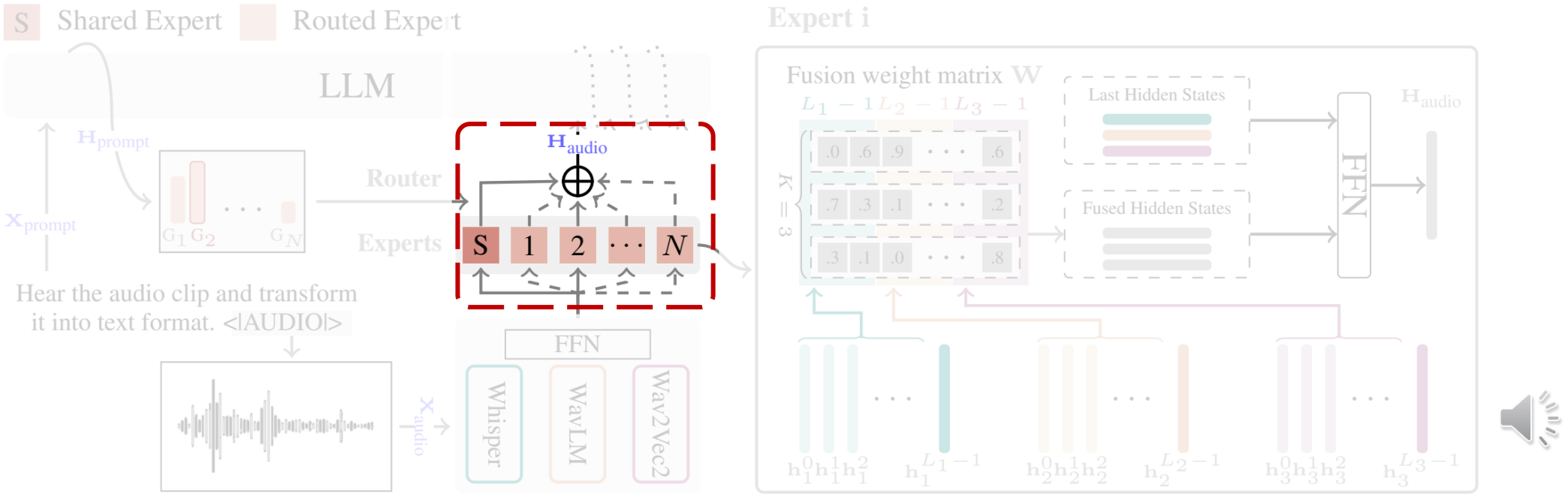
- We propose.

  > a novel multi-encoder SpeechLLM, which effectively leverages features from every layer of each encoder.



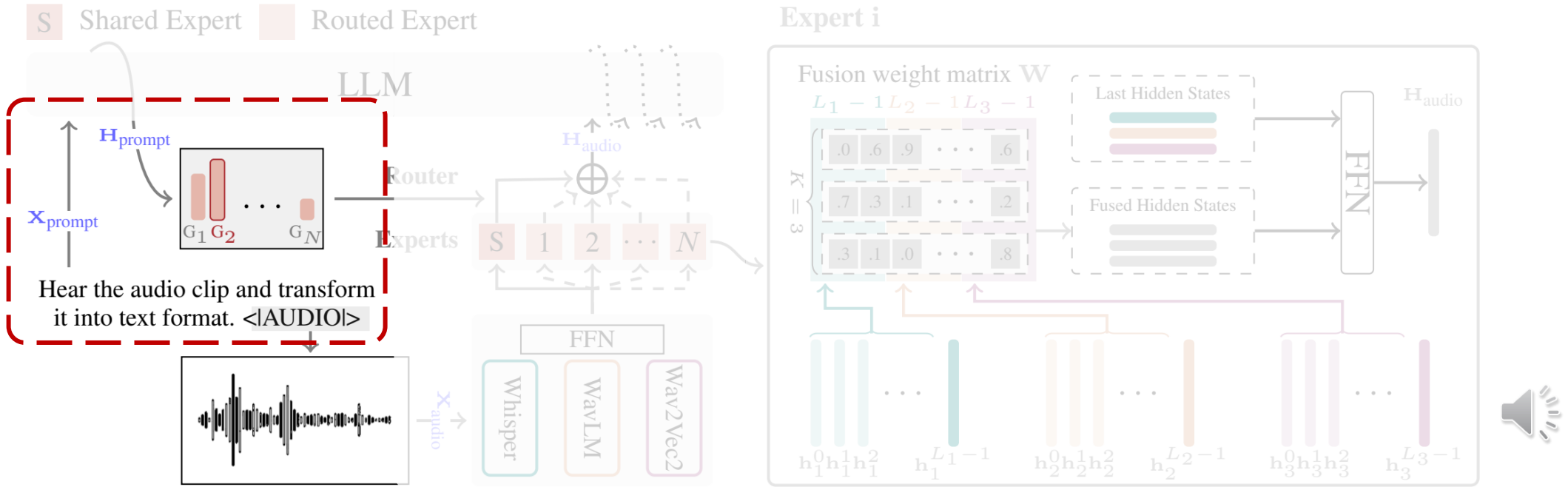
Method - Overview

- We propose.
  > a novel multi-encoder SpeechLLM, which effectively leverages features from every layer of each encoder.
  > an adapter layer for multi-feature fusion, similar to a Mixture-of-Experts (MoE) architecture.



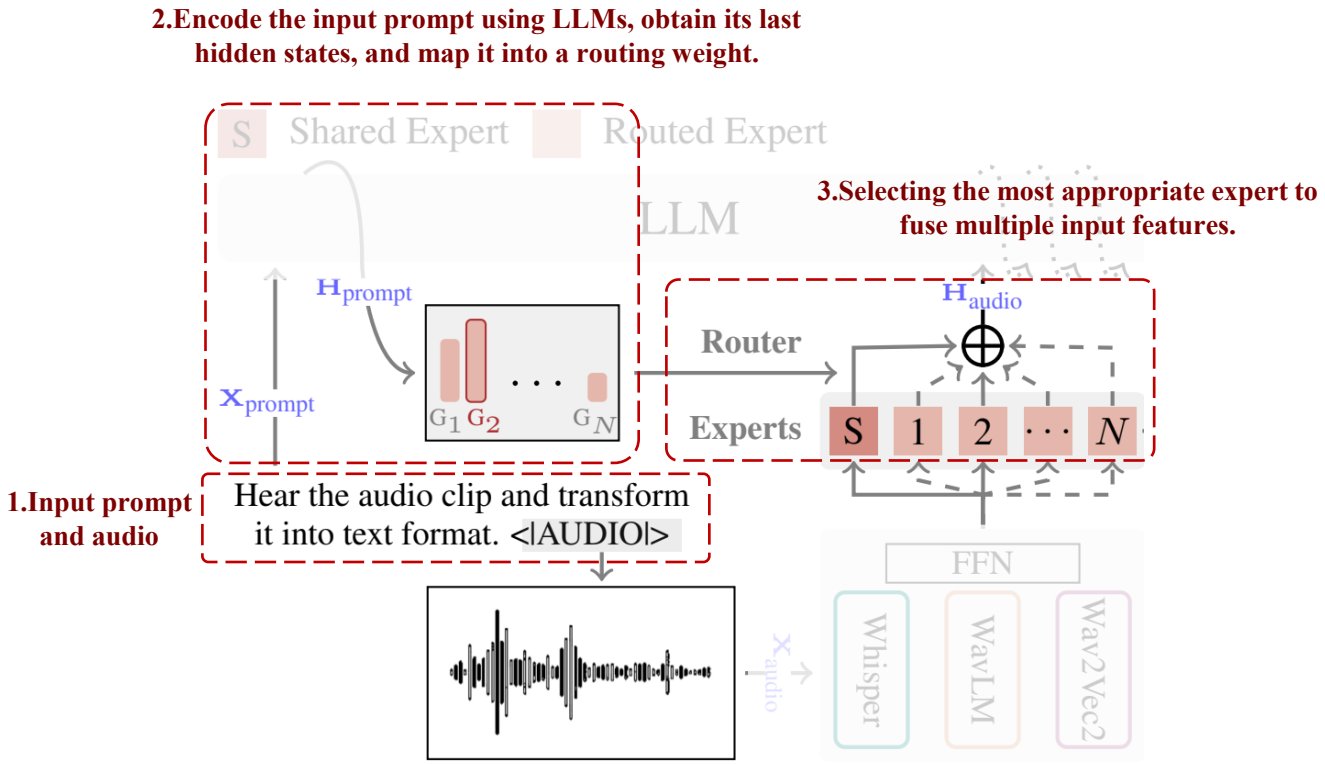
Method - Overview

- We propose.
  - > a novel multi-encoder SpeechLLM, which effectively leverages features from every layer of each encoder.
  - > an adapter layer for multi-feature fusion, similar to a Mixture-of-Experts (MoE) architecture.
  - > a prompt-aware routing mechanism to assign distinct weights to each encoder and its layers.



Method - Detail

- Prompt Aware Routing
  > Task-specific information in speech processing tasks is typically embedded in prompts



$$\mathbf{H}_{\text{prompt}} = \text{LLM}(\mathbf{X}_{\text{prompt}})$$

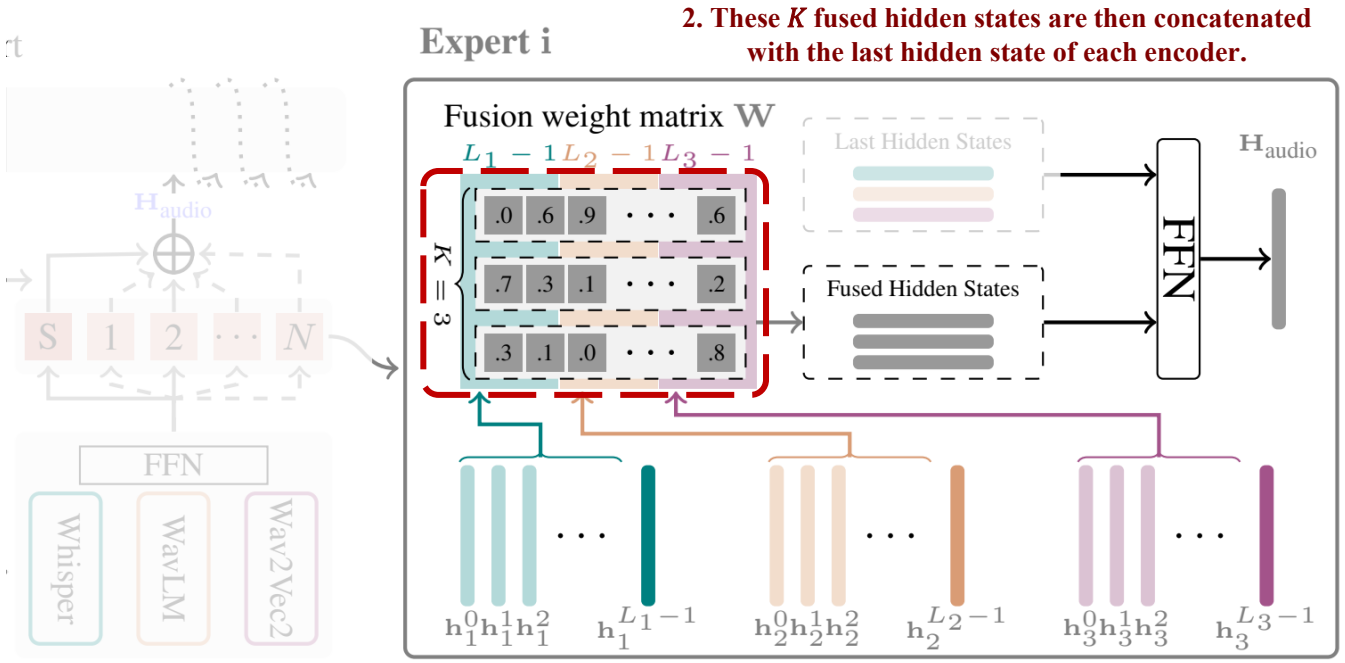
$$\text{P}(\text{Task}|\mathbf{H}_{\text{prompt}}) = \text{Softmax}(\text{FFN}(\mathbf{H}_{\text{prompt}}))$$

$$\mathbf{G}_j = \begin{cases} 1 & \text{if } j \in \text{Top-1}(\text{P}(\text{Task}|\mathbf{H}_{\text{prompt}})) \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{H}_{\text{audio}} = \text{Expert}_{\text{share}}(\mathbf{H}_{\{1,2,3\}}) + \sum_{j=1}^{N} G_j(\mathbf{X}_{\text{prompt}}) \times \text{Expert}_j(\mathbf{H}_{\{1,2,3\}})$$

Method - Overview

- Multi-layer Fusion
  - > The use of fusion weight matrices highlights multi-level feature fusion



$$\mathbf{H}_i = \{\mathbf{h}_i^0, \mathbf{h}_i^1, ...\mathbf{h}_i^{L_i}\}$$

$$\mathbf{h}_k^{\text{fused}} = \sum_{i=1}^{3}\sum_{j=1}^{L_i} w_{i,j}^k \cdot \mathbf{h}_{i,j}$$

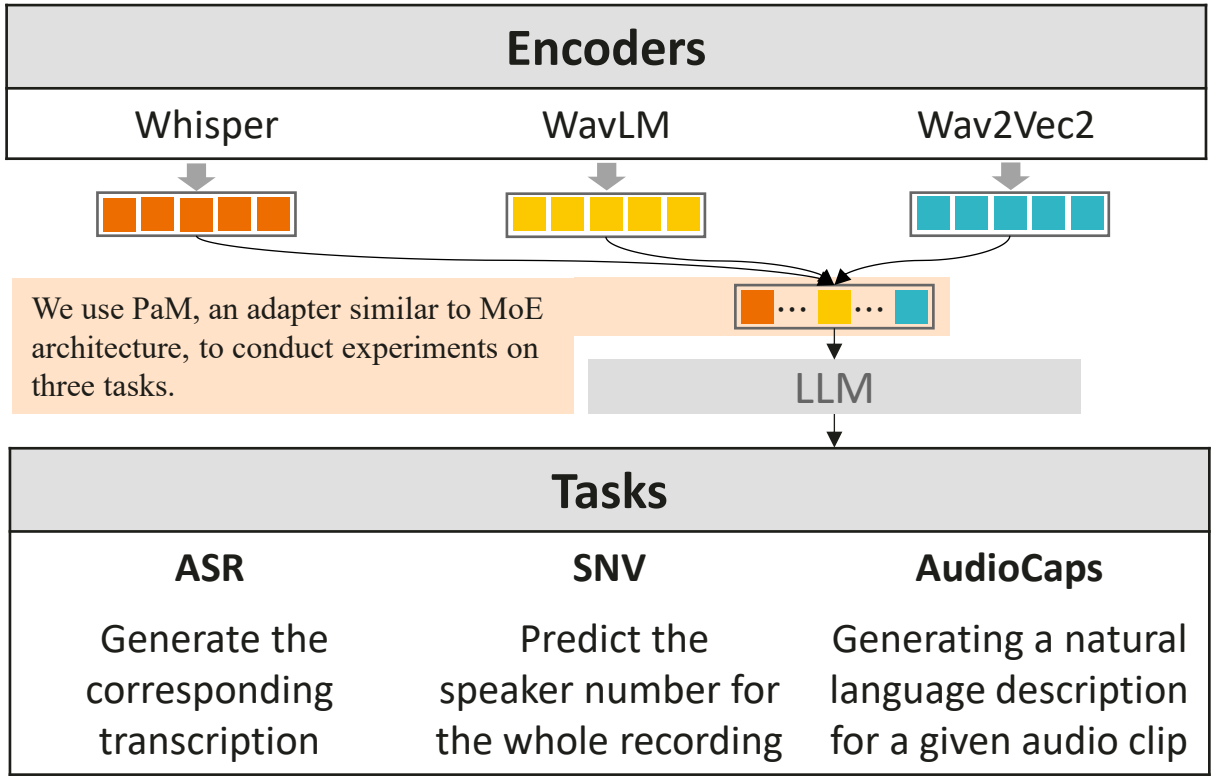
$$\mathbf{h}^{\text{final}} = \text{Concat}(\mathbf{h}_{\{1,2,3\}}^{L_i}, \mathbf{h}_{\{1,...,k\}}^{\text{fused}})$$

$$\text{Expert}(\cdot) = \text{FFN}(\mathbf{h}^{\text{final}})$$

1. In each expert, We use *K* sets of distinct fusion parameters to integrate the hidden states from all layers of all encoders (excluding the last hidden state of each encoder), and yielding *K* fused hidden states.

Experimental Setups

- We conduct experiments with three encoders on three different downstream tasks.



ASR	1. Hear the audio clip and transform it into text format. <|AUDIO|> 2. Listen to the following audio and create a corresponding text transcript. <|AUDIO|>
Speaker Number Verification	1. How many speakers' contributions are in this recording? <|AUDIO|> 2. What is the number of speakers in this spoken content? <|AUDIO|>
AC	1. Listen to this audio and provide a detailed description. <|AUDIO|> 2. Analyze the recording and summarize its contents. <|AUDIO|>

Table 4: Examples of prompts for different tasks.

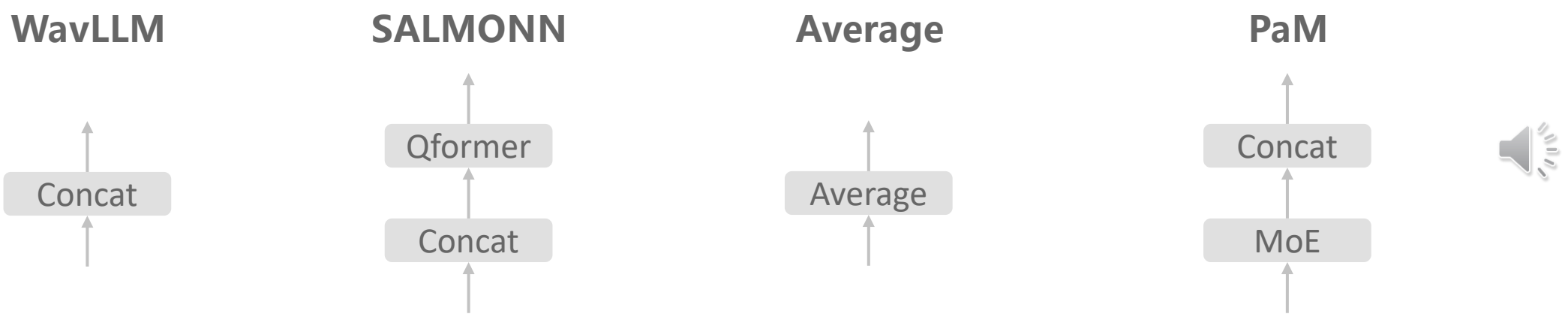
Data Source	Task	Hours	Sample
Librispeech-clean-100 (Panayotov et al., 2015)	ASR	100h	28539
AMI (Kraaij et al., 2005)	ASR	100h	108502
Common Voice V4 (Part) (Ardila et al., 2019)	SNV	~150h	137041
Audio Caption (Kim et al., 2019)	AC	100h	39126

Table 5: The whole training dataset.

Results

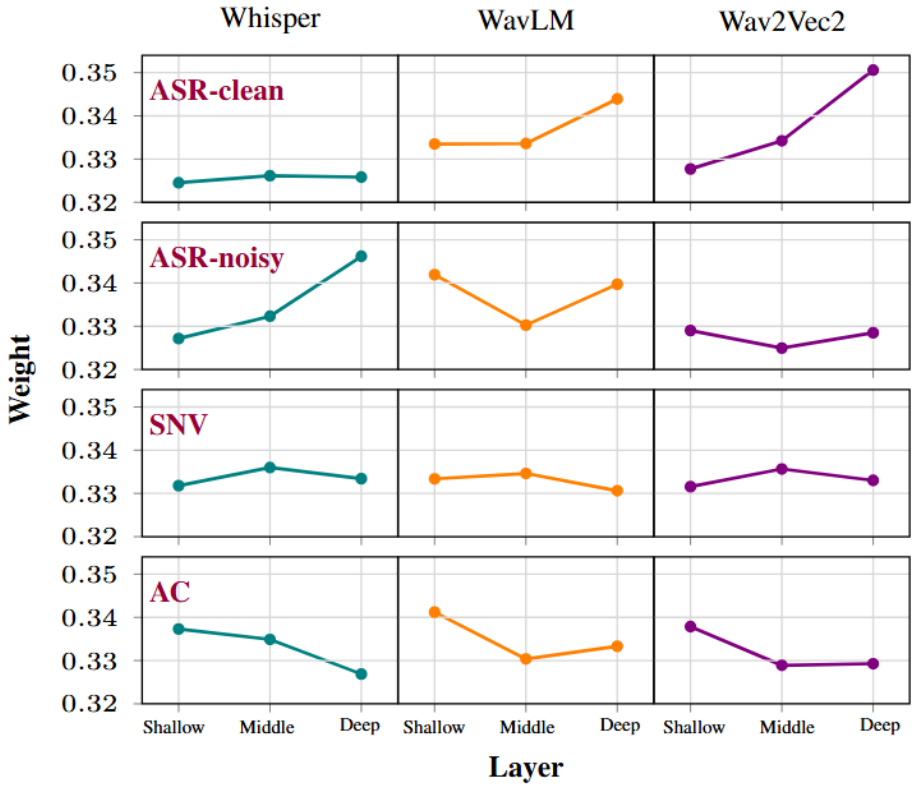
- Single-encoder Baselines, Multi-encoder Baselines and PaM

Model	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA	AVG Rank↓
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑	
Single-encoder Baselines							
- Whisper (Radford et al., 2022)	9.61	16.73	**16.27**	18.80%	**32.96**	**15.04**	5.67
- WavLM (Chen et al., 2022)	5.59	10.57	18.97	**41.40%**	27.14	12.77	5.50
- Wav2Vec2 (Baevski et al., 2020)	**4.30**	**09.46**	26.69	39.00%	23.81	11.20	5.33
Multi-encoder Baselines							
- WavLLM (Hu et al., 2024)	4.95 (- 0.65)	09.19 (+0.27)	15.29 (+0.98)	39.20% (- 2.20)	34.93 (+1.97)	16.35 (+1.31)	2.67
- SALMONN (Tang et al., 2024)	5.04 (- 0.74)	09.70 (- 0.24)	19.04 (- 2.77)	49.40% (+8.00)	34.86 (+1.90)	15.97 (+0.93)	3.50
- Average	4.76 (- 0.46)	10.43 (- 0.97)	17.20 (- 0.93)	45.50% (+4.10)	33.22 (+0.26)	15.53 (+0.49)	3.67
PaM (Ours)	3.65 (+0.65)	07.07 (+2.39)	12.79 (+3.48)	42.80% (+1.40)	35.47 (+2.51)	15.70 (+0.66)	**1.67**



Results

- What PaM Focuses On?
  > We plot the weights assigned by PaM to different layers across various tasks, averaging the weights over every four adjacent layers.



- **ASR-clean: Deep layers + Wav2Vec2**
- **ASR-noisy: Deep layers + Whisper**
- **SNV: Middle layers + Each encoder**
- **AC: Shallow layers + Each encoder**

Model	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑
Single-encoder Baselines						
- Whisper (Radford et al., 2022)	9.61	16.73	**16.27**	18.80%	**32.96**	**15.04**
- WavLM (Chen et al., 2022)	5.59	10.57	18.97	**41.40%**	27.14	12.77
- Wav2Vec2 (Baevski et al., 2020)	**4.30**	**09.46**	26.69	39.00%	23.81	11.20

Results

- More encoders or better encoders?
  > Using an encoder that performs well on a specific task significantly improves the speech LLM's performance on that task.
  > Employing more encoders enhances the performance of the speech LLM on all downstream tasks.

The average (AVG) rank reflects the overall performance across multitasks.

Encoders	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA	AVG Rank↓
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑	
Whisper+X	4.42	8.92	13.96	43.70%	**35.41**	16.11	4.5
WavLM+X	4.07	7.97	18.47	47.47%	32.48	15.01	5.5
Wav2Vec2+X	**3.42**	7.20	18.18	45.13%	32.33	15.02	4.3
HuBERT+X	3.87	8.21	19.49	36.90%	31.82	15.08	6.8
Whisper+X+Y	4.22	8.11	**13.79**	**49.77%**	35.37	**16.31**	**3.0**
WavLM+X+Y	3.72	7.99	16.35	38.73%	34.23	15.82	4.0
Wav2Vec2+X+Y	3.77	**6.90**	16.90	42.63%	34.23	15.82	3.8
HuBERT+X+Y	4.03	7.96	17.13	44.17%	34.18	16.13	4.0

Results

- More encoders or better encoders?
  > Single better encoder does not yield significant improvements.
  > All encoders are improved, the overall performance increases.
  > Employing larger and stronger LLMs does not result in substantial gains.

Models	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑
Base PaM	3.65	07.07	12.79	42.8%	35.47	15.70
PaM with Larger Encoders						
- ① Whisper-Medium	3.93	07.93	12.43	47.5%	35.61	15.58
- ② WavLM-Large	3.75	06.43	12.10	45.6%	35.95	16.71
- ③ Wav2Vec2-Large	3.74	06.49	15.06	47.6%	35.50	**16.74**
- ① + ② + ③	**3.58**	**05.93**	**11.51**	**56.9%**	**36.94**	16.50
PaM with different LLMs						
- Qwen2.5-7B	3.68	08.36	15.26	43.7%	35.46	15.71
- LLaMA3.2-3B	4.98	11.57	15.57	50.5%	35.83	16.34
- LLaMA3.1-8B	4.85	08.87	15.01	50.8%	35.81	15.82

Results

- Parameter Efficiency
  > PaM outperforms baseline methods on most tasks under a similar parameter scale.

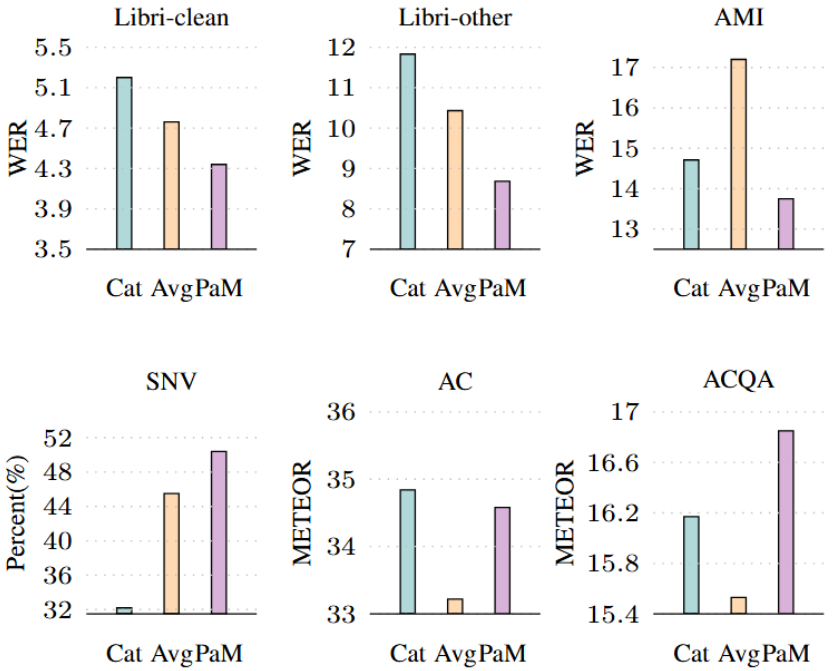


Figure 4: Performance comparison of a smaller PaM (29M parameters) with Concatenation (37M parameters) and Average (24M parameters).

Appendix

- PaM and Concatenation Adapter with larger LLM

Models	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑
Qwen2.5-7B						
- WavLLM	5.13	10.81	13.66	51.1%	36.27	16.55
- PaM	3.68	8.36	15.26	43.7%	35.46	15.71
LLaMA3.1-8B						
- WavLLM	6.38	14.08	13.95	03.1%	34.91	14.76
- PaM	4.85	8.87	15.01	50.8%	35.81	15.82

Table 10: Results based on our PaM adapter and WavLLM with larger LLM.

Appendix

- Ablation study of PaM with different routing methods.

Models	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA	AVG Rank↓
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑	
MoE (audio-based)	4.12	8.32	11.21	26.00%	35.66	15.61	3.17
PaM (with one expert)	4.27	10.04	13.77	45.80%	35.76	16.47	2.83
PaM (without shared)	4.31	7.89	13.43	45.20%	35.27	16.35	3.33
PaM (without task label)	3.97	7.97	13.14	43.50%	35.54	15.42	3.17
PaM (ours)	3.65	7.07	12.79	42.80%	35.47	15.70	2.50

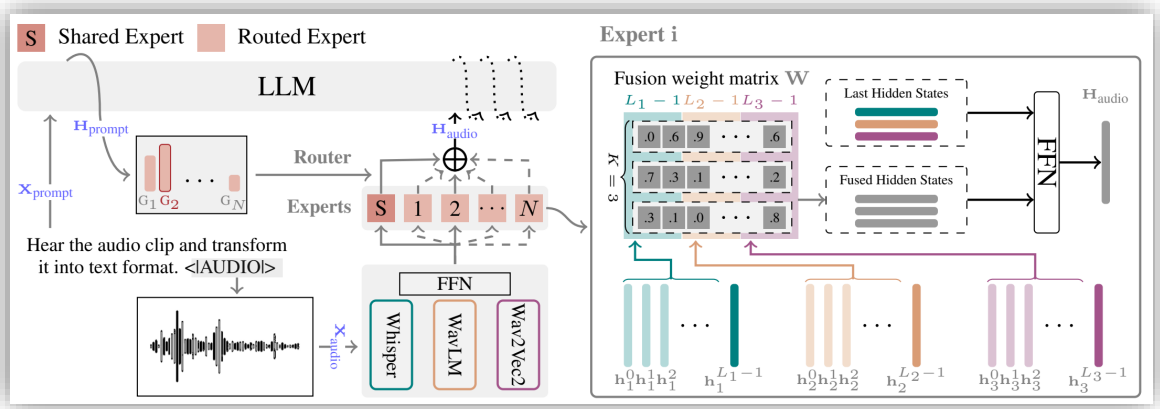
Table 11: Ablation results on our routing method and the result based on the audio-based routing method

- Evaluating the AC task with more metrics.

Models	AudioCaps					AudioCaps QA				
	METEOR↑	FENSE↑	sBERT↑	CIDEr↑	SPICE↑	METEOR↑	FENSE↑	sBERT↑	CIDEr↑	SPICE↑
Single-encoder Baselines										
- Whisper	32.96	0.108	0.596	0.431	0.158	15.04	0.105	0.402	0.205	0.083
- WavLM	27.14	0.106	0.500	0.290	0.122	12.77	0.104	0.337	0.127	0.049
- Wav2Vec2	23.81	0.096	0.414	0.205	0.095	11.20	0.092	0.282	0.073	0.038
Multi-encoder Baselines										
- WavLLM	34.93	0.109	0.640	0.569	0.175	**16.35**	**0.108**	**0.448**	**0.303**	**0.109**
- SALMONN	34.86	0.109	0.631	0.542	0.158	15.97	0.108	0.432	0.254	0.093
- Average	33.22	0.108	0.615	0.471	0.166	15.53	0.108	0.425	0.229	0.093
PaM	**35.47**	**0.111**	**0.644**	**0.581**	**0.183**	15.70	0.108	0.428	0.267	0.087

Table 7: More results based on various metrics on the AC task. The sBERT represents Sentence-BERT.

Thank you.



Model	LibriSpeech		AMI	SNV	AudioCaps	AudioCaps QA	AVG Rank↓
	WER(clean)↓	WER(other)↓	WER↓	Acc↑	METEOR↑	METEOR↑	
Single-encoder Baselines							
- Whisper (Radford et al., 2022)	9.61	16.73	**16.27**	18.80%	**32.96**	**15.04**	5.67
- WavLM (Chen et al., 2022)	5.59	10.57	18.97	**41.40%**	27.14	12.77	5.50
- Wav2Vec2 (Baevski et al., 2020)	**4.30**	**09.46**	26.69	39.00%	23.81	11.20	5.33
Multi-encoder Baselines							
- WavLLM (Hu et al., 2024)	4.95 (- 0.65)	09.19 (+0.27)	15.29 (+0.98)	39.20% (- 2.20)	34.93 (+1.97)	16.35 (+1.31)	2.67
- SALMONN (Tang et al., 2024)	5.04 (- 0.74)	09.70 (- 0.24)	19.04 (- 2.77)	49.40% (+8.00)	34.86 (+1.90)	15.97 (+0.93)	3.50
- Average	4.76 (- 0.46)	10.43 (- 0.97)	17.20 (- 0.93)	45.50% (+4.10)	33.22 (+0.26)	15.53 (+0.49)	3.67
PaM (Ours)	3.65 (+0.65)	07.07 (+2.39)	12.79 (+3.48)	42.80% (+1.40)	35.47 (+2.51)	15.70 (+0.66)	**1.67**

把数字世界带入每个人、每个家庭、每个组织，构建万物互联的智能世界。

Bring digital to every person, home and organization for a fully connected, intelligent world.

NEU NLP SINCE 1973

HUAWEI