INTERSPEECH2025

MOPSA: Mixture of Prompt-Experts Based Speaker Adaptation for Elderly Speech Recognition

Chengxi Deng1 , Xurong Xie2∗ , Shujie Hu1 , Mengzhe Geng3 , Yicong Jiang2 , Jiankun Zhao1 , Jiajun Deng1 , Guinan Li1 , Youjun Chen1 , Huimeng Wang1 , Haoning Xu1 , Mingyu Cui1 , Xunying Liu1∗

1The Chinese University of Hong Kong, Hong Kong SAR, China 2 Institute of Software, Chinese Academy of Sciences, China 3National Research Council Canada, Canada

2025.10.10

HyorinJung

Index

- **Introduction**
- **Background**
- **Analyze & Delineate**
  - **Experiments**
    - Setting
    - Result
- **Conclusion**
- **Proposal**

Introduction

Why Elderly Speech Recognition Matters?

- Elderly speech is often unclear due to:
  - Neuromotor weakening → imprecise articulation
  - Cognitive decline → repetitive or incomplete phrasing
- Current ASR foundation models (e.g., Whisper, WavLM) are trained mainly on normal speech

👉 Need for **adaptive, low-latency ASR systems** specialized for elderly speakers

Introduction

Key Challenges

- Data Sparsity : Few labeled elderly speech samples
- Speaker Heterogeneity : High variability in accent, gender, and cognitive level
- Real-Time Adaptation : Batch adaptation introduces latency
- Language Degradation : Existing models adapt acoustically, not linguistically

Background

Prior adaptation methods: Elderly speech recognition.

Specially designed speaker-dependent (SD) parameters → Adapter-based speaker parameter estimation approaches

Background

Prior adaptation methods: Normal speech adaptation.

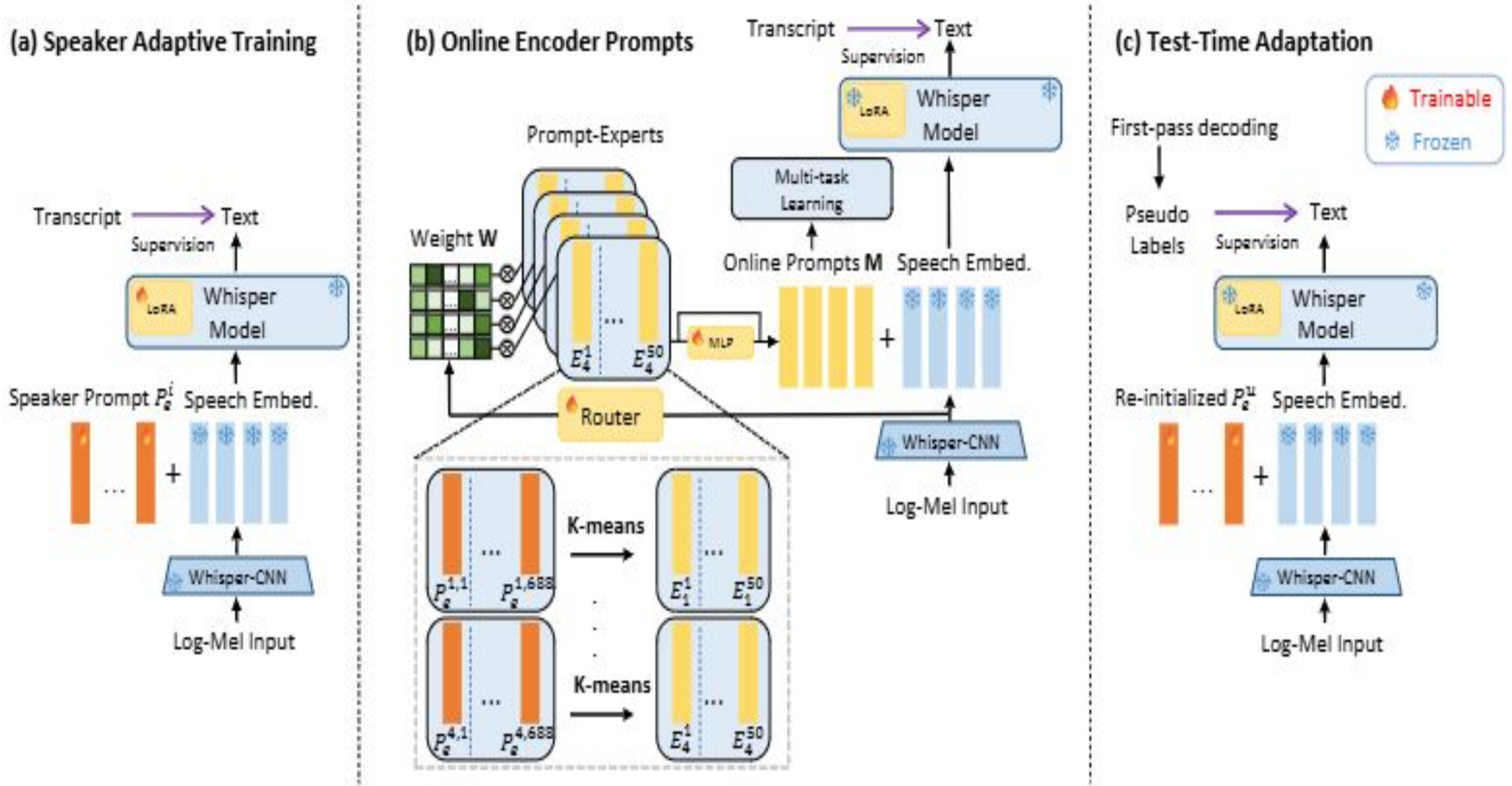
Mixture-of-Experts (MoE)	Prompt-based adaptation
• Multiple LoRA or Adapter modules act as experts • Each capturing distinct speaker-related information • The experts and their weights serve as speaker-dependent (SD) parameters 👉Combine multiple specialized sub-models → dynamically select experts per speaker.	• Learns a small set of speaker-dependent prompts or transformations inserted into the model • Preserves the model's general knowledge while minimizing computational overhead • Provides efficient and lightweight adaptation for unseen or new speakers 👉Instead of tuning model weights, tune only "prompt vectors" to guide model behavior

Background

Prior adaptation methods. Limitation

- Limited Generalization Capacity
  - LoRA-based Mixture-of-Experts → limited generalization to unseen speakers
- Latency Issue
- Lack of Linguistic Adaptation
  - Prompt-based methods → often only acoustic, not linguistic

👉 Goal: Combine Mixture-of-Experts (MoE) and Prompt-based learning for zero-shot, real-time elderly adaptation



(a) Speaker Adaptive Training
Transcript
Text
Supervision
LoRA
Whisper Model
Speaker Prompt P_e^i
Speech Embed.
Whisper-CNN
Log-Mel Input
(b) Online Encoder Prompts
Transcript
Text
Supervision
LoRA
Whisper Model
Prompt-Experts
Multi-task Learning
Weight W
Online Prompts M
Speech Embed.
MLP
E_4^1
E_4^{50}
Router
Whisper-CNN
Log-Mel Input
$P_e^{1,1}$
$P_e^{1,688}$
K-means
E_1^1
E_1^{50}
$P_e^{4,1}$
$P_e^{4,688}$
K-means
E_4^1
E_4^{50}
(c) Test-Time Adaptation
Trainable
Frozen
First-pass decoding
Pseudo Labels
Text
Supervision
LoRA
Whisper Model
Re-initialized P_e^u
Speech Embed.
Whisper-CNN
Log-Mel Input

Background

Three Components

- **Speaker Adaptive Training (SAT):** Obtain prompt embeddings per speaker
- **K-means Clustering:** Form prompt-experts capturing shared traits
- **Router Network:** Dynamically mix experts online → real-time adaptation

Background

Speaker Adaptive Training (SAT)

- Log-Mel spectrogram → two CNN layers for down-sampling
- Encoder & Decoder Prompts are initialized for each training speaker
- LoRA modules inserted in attention layers allow parameter-efficient fine-tuning while freezing most of Whisper's weights
- The model learns both speaker-specific prompts and shared LoRA parameters using reference transcripts
- collect speaker-specific knowledge for clustering and adaptation : Prompt Experts

Background

Log-Mel Spectrogram

$$X \in \mathbb{R}^{D \times T}$$

CNN Down-sampling

Encoder

$$H_e = \mathrm{Encoder}(\mathrm{Conv}(X))$$

Decoder

$$\hat{y}_m = \mathrm{Decoder}(s, \hat{y}_{1:m-1}, H_e)$$

- s : special token sequence
  ex<|PREV|>, decoder prompt,
  <|SOT|>, <|LANGUAGE|>,
  <|TRANSCRIBE|>, <|NO-TIMESTAMP|>

Text

Background

Batch-Mode Speaker Prompt Adaptation

- Adaptation Data Accumulation :
  - first-pass decoding : Speaker prompts are estimated using pseudo labels ($\hat{Y}$)
- Requires large test-speaker data → latency and dependence on pseudo label quality

Test-Time Adaptation (TTA)

- During TTA, Whisper model parameters are frozen
- Trainable prompts for each test speaker are initialized in the encoder and decoder, and optimized using pseudo labels.

👉 Goal: Reuse learned speaker knowledge and adapt instantly without retraining

Background

Test-Time Adaptation (TTA)

$$\boldsymbol{H}_e^u = \mathrm{Encoder}(\mathrm{Concat}[\boldsymbol{P}_e^u, \mathrm{Conv}(\boldsymbol{X})])$$

X: Input log-Mel spectrogram of test speaker *u*
Peu: Speaker-dependent **encoder prompts**

$$\hat{y}_m = \mathrm{Decoder}(s(\boldsymbol{P}_d^u), \hat{y}_{1:m-1}, \boldsymbol{H}_e^u)$$

Pdu: Speaker-dependent **decoder prompts** embedded in special token sequence s

$$\hat{\boldsymbol{P}}^u = \arg\min_{\{\boldsymbol{P}^u\}} \{\mathcal{L}_C(\hat{\boldsymbol{Y}}^u | \boldsymbol{X}^u; \boldsymbol{P}^u)\}$$

LC: Cross-entropy loss between model output and **pseudo-labels**

Background

Online Mixture of Prompt-Experts Speaker Adaptation

Step1 - Prompt-Expert Construction

- From SAT, each training speaker *i* has encoder-side prompts $P_e^{i,l} \in \mathbb{R}^{D \times L_e}$ for each prompt position l
- Apply K-means clustering to all speaker prompts
- The cluster centroids become the prompt-experts $E_c^l = K\text{-means}(P_e^l)$
- This creates **C experts** per prompt position, each representing a distinct speaker style
- The same procedure is applied to decoder-side prompts

Background

Online Mixture of Prompt-Experts Speaker Adaptation

Step2 - Router Network

- **Purpose:** Select and combine prompt-experts dynamically for each input
- Architecture:
  - **Global Context Module:** two multi-head attention layers + LayerNorm
  - **Downsampling Network:** 3 sequential blocks (CNN-1d + BN + AvgPool) followed by 3 linear layers (1200 → 1000 → 50 dims)
- The router outputs **weights $W = [W^1,\dots,W^L] \in \mathbb{R}^{C\times L}$,** where each vector corresponds to the C experts for position l

Background

Online Mixture of Prompt-Experts Speaker Adaptation

Step 3 – Online Prompt Generation

- Weighted combination of experts produces speaker-adaptive prompts:

$$Z_l = \sum_{j=1}^{C} w_j^l \cdot E_j^l, \quad M = Z + \varphi(Z)$$

- Z∈ℝ^{D×L}: Weighted sum of expert
- φ(Z) : Linear projection layer for refinement
- M: Final online prompt, fed into Whisper for decoding

Background

Online Mixture of Prompt-Experts Speaker Adaptation

Step 4 - Multi-Task Learning of Router Network

- Total Loss:

$$L_{Router} = L_{ASR} + \alpha L_{Spkr} + \beta L_{MSE}$$

Background

Online Mixture of Prompt-Experts Speaker Adaptation

Step 4 - Multi-Task Learning of Router Network

- The router is optimized with three complementary losses:

Loss	Description	Purpose
$\mathcal{L}_{ASR}$	Cross- entropy loss	Maintain ASR accuracy
$\mathcal{L}_{Spkr}$.	Cross- entropy loss on speaker ID classification	Preserve speaker identity consistency
L_{MSE}	MSE between online prompts and SAT prompts	Align new prompts with trained SAT space

Experiments : Setting

Datasets

- **English DementiaBank Pitt Corpus:**
    a. 33 hours of elderly interview recordings (292 sessions)
    b. Train: 688 speakers
    c. Dev: 119 speakers
    d. Eval: 95 speakers
    e. After silence removal and data augmentation → total 58.9 hours.
- **Cantonese JCCOCC MoCA Corpus**
    a. 256 interviews for cognitive assessment
    b. Train: 369 speakers
    c. Dev/Eval: 49 speakers each (no overlap)

Experiments : Setting

Base Models

- Whisper-medium3 (for strong generalization)

Implementation Details

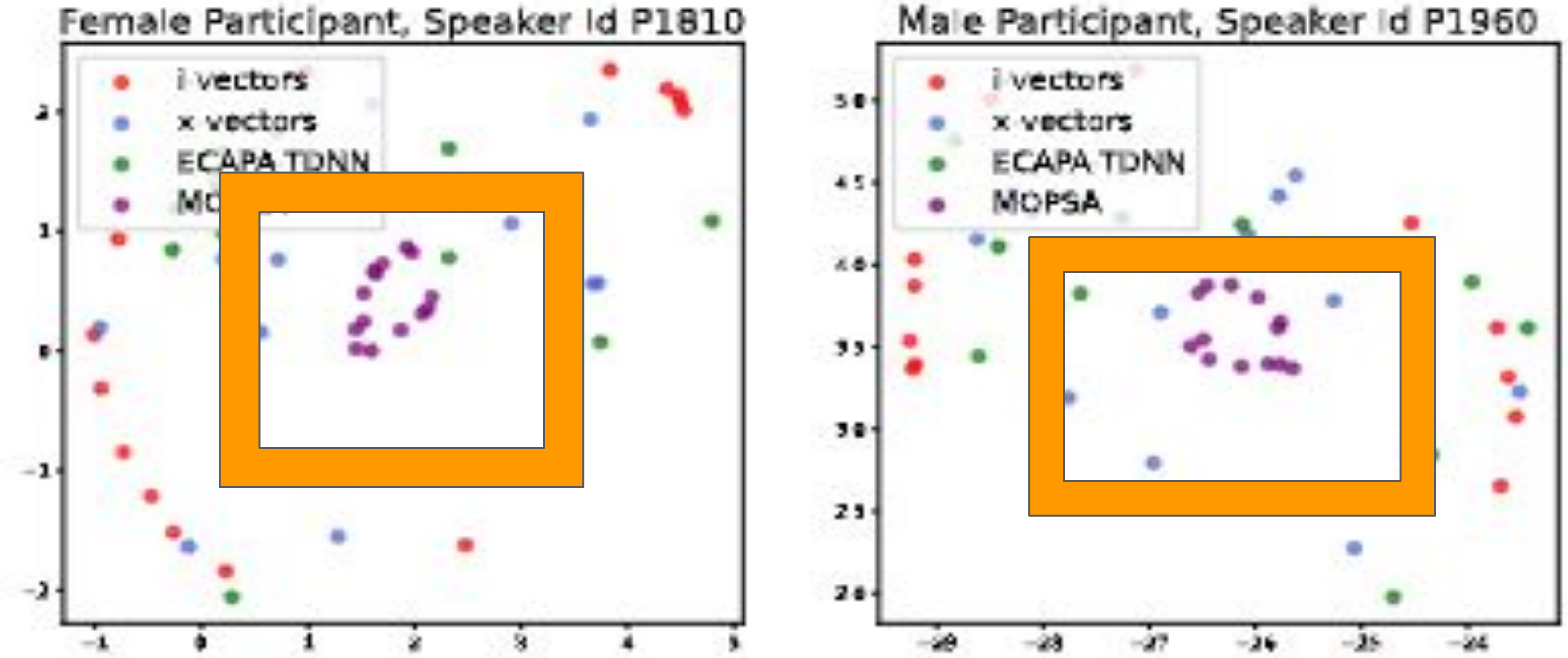
- Applied *LoRA* on attention layers (query, key, value, att.out) with rank = 8
- Router Network:
  - Two multi-head attention layers + LayerNorm
  - Downsampling blocks: (512 → 256 → 128 channels)
  - Linear layers: (1200 → 1000 → 50 dimensions)
  - Dropout after each layer to prevent overfitting.
- Achieved state-of-the-art baseline results on both datasets

Experiments : Ablation Study

Sys.	Multi-task. MSE loss	Multi-task. Spkr loss	Prompt Pos.	Prompt Length	Prompt Cluster	DementiaBank Pitt WER(%) Dev. Par.	Dev. Inv.	Eval. Par.	Eval. Inv.	All
B1	✗	✗	Enc	1	✗	27.83	12.62	19.48	11.88	19.77
B2				2		27.53	12.70	19.36	12.10	19.67
B3				4		**27.53**	**12.44**	**19.53**	**12.43**	**19.60**
B4				8		27.96	11.94	19.82	12.32	19.61
B5	✗	✗	Dec	1	✗	**28.12**	**12.16**	**20.01**	**12.76**	**19.81**
B6				2		28.95	12.47	20.30	12.54	20.31
B7				4		28.21	12.48	20.76	12.76	20.10
B8				8		28.47	12.65	20.76	12.76	20.28
O1	✗	✗	Enc	4	688 (Unclustered)	28.69	12.77	19.95	12.10	20.25
O2	✗	✓				28.78	12.84	20.03	12.54	20.35
O3	✓	✗				28.76	12.57	19.90	11.65	20.18
O4	✓	✓				28.69	12.68	19.74	11.99	20.18
O5	✓	✓			25	28.92	12.40	19.67	12.10	20.15
O6	✗	✗			50	28.69	12.80	19.82	12.65	20.26
O7	✗	✓				29.17	12.45	19.86	12.76	20.32
O8	✓	✗				28.73	12.67	19.61	10.99	20.14
O9	✓	✓				**28.49**	**12.32**	**19.48**	**10.88**	**19.88**
O10	✓	✓			150	28.70	12.57	19.80	11.43	20.13

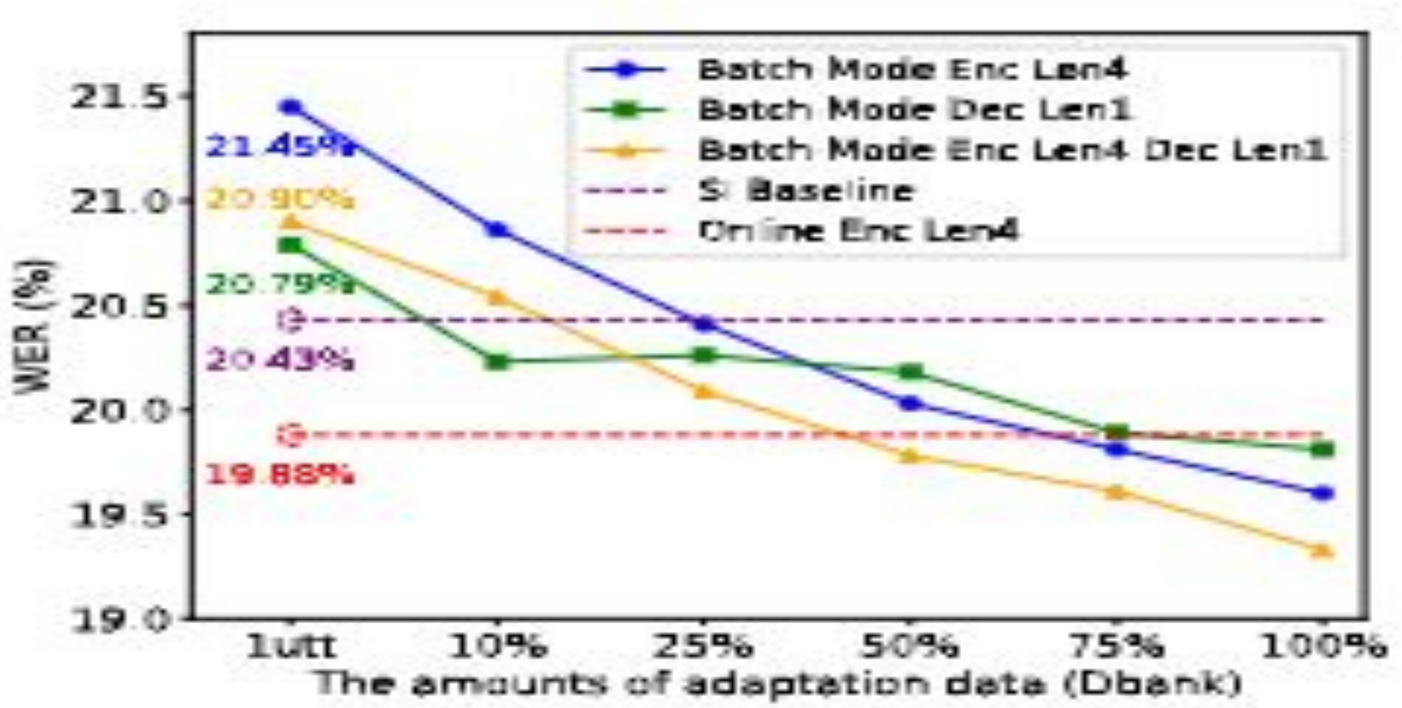
Experiments : Results

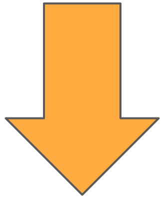
Speaker Representation



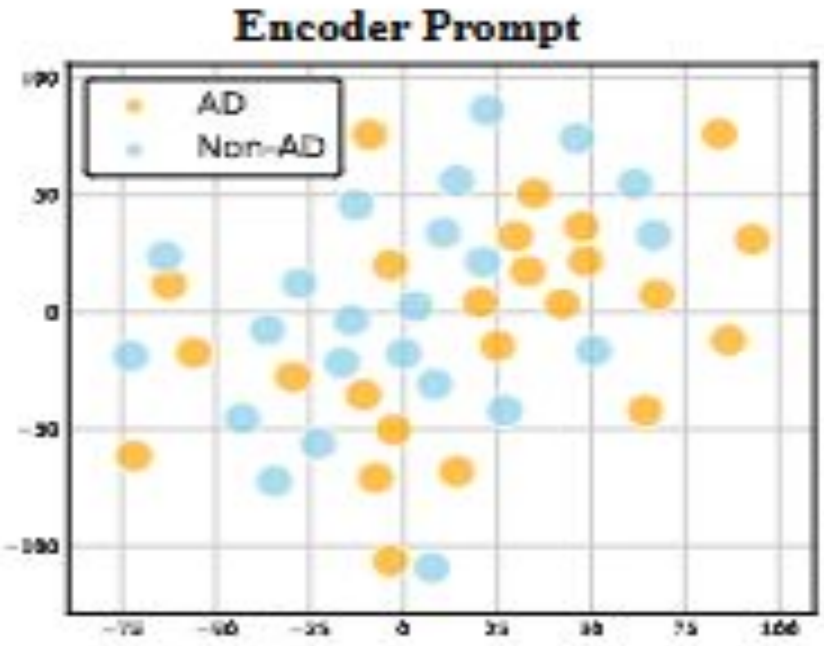
👉more consistent and distinct speaker representations

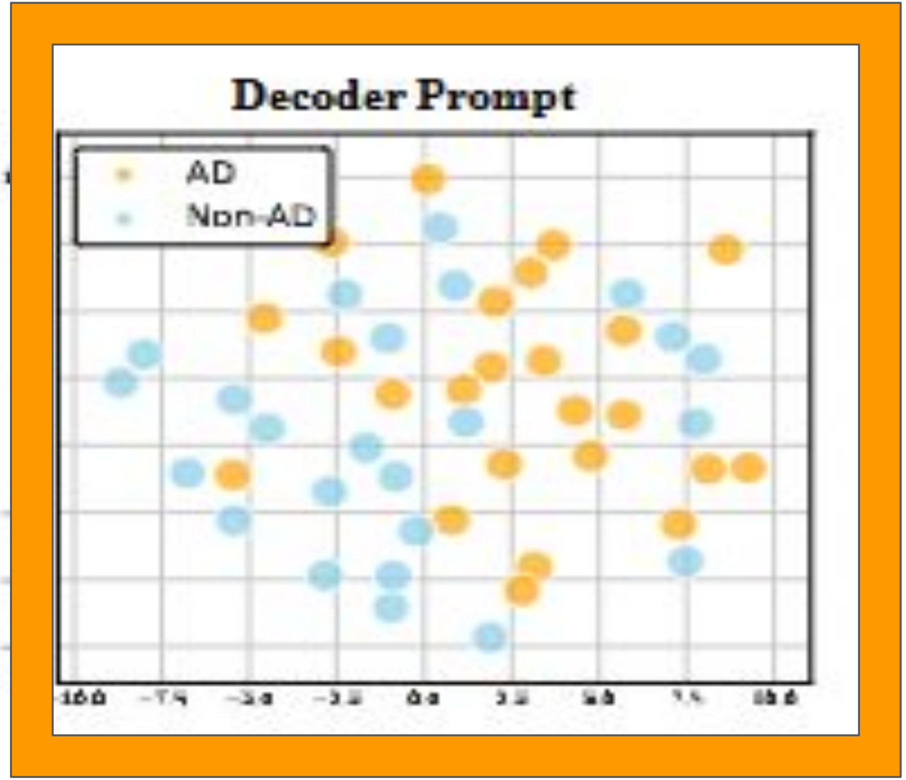
Adaptation Efficiency





Disease Correlation





Sys.	Model	Speaker Modeling	SAT	TTA	Online	DementiaBank Pitt WER(%)							JCCOCC MoCA CER(%)			
						Dev.		Eval.		All	SD Parm.	RTF	Dev.	Eval.	All	SD Parm.
						Par.	Inv.	Par.	Inv.							
1		-	-	-	-	28.79	12.76	20.68	12.65	20.43	-	0.24	28.68	25.79	27.23	-
2		LHUC	✓			28.45	12.40	20.11	11.43	20.02*	0.15M	4.15	30.77	27.63	29.19	0.1M
3		RAB [10]	✗			29.54	13.14	21.27	12.87	20.99	0.53M	2.78	29.35	26.55	27.94	0.53M
4		Prompt Enc	✓	✓	-	27.53*	12.44	19.53*	12.43	**19.60***	0.60M	4.03	27.50*	24.32*	**25.90***	0.4M
5		Prompt Dec	✓			28.12	12.16*	20.01	12.76	**19.81***	0.15M	3.25	27.75*	24.49*	**26.09***	0.1M
6	Whisper (LoRA)	Prompt Enc&Dec	✓			27.41*	12.05*	19.25*	11.76	**19.33***	0.75M	4.13	27.23*	24.15*	**25.69***	0.50M
7		i-vector	✓			29.32	13.13	21.75	10.99	20.92		0.27	39.04	36.16	37.59	
8		x-vector	✓			31.49	14.96	23.37	12.87	22.84		0.27	29.87	27.43	28.64	
9		ECAPA-TDNN [35]	✓	✗	✓	29.01	13.85	21.27	10.54	20.88	-	0.27	33.48	30.19	31.83	-
10		MOPSA Enc	✓			28.49	12.32	19.48*	10.88	**19.88***		0.25	28.10*	25.10*	**26.59***	
11		MOPSA Dec	✓			28.76	12.85	20.22	11.10	20.33		0.25	27.54*	25.20*	**26.36***	
12		MOPSA Enc&Dec	✓			27.64*	12.52	19.08*	11.21	**19.57***		0.27	27.15*	24.39*	**25.76***	

Conclusion

- Proposed a Mixture of Prompt-Experts based Speaker Adaptation (MOPSA) for elderly speech recognition
- Combines cluster-based prompt experts with a dynamic router network
- Enables zero-shot and real-time adaptation to unseen speakers

Proposal

- Router Network Interpretability & Robustness
  - Router decisions are not easily interpretable and may fail under short or noisy speech
  - Apply explainable AI tools (e.g., SHAP, LIME) to visualize feature importance
- Fixed K-means clustering (C=50) may be sensitive to initialization and not optimal for all data
  - explore adaptive clustering (e.g., DBSCAN, GMM) for better robustness

Appendix

- RTF(Real-Time Factor)

$$\text{RTF} = \frac{\text{Processing Time}}{\text{Audio Duration}}$$

- 특정 길이의 오디오를 처리하는 데 걸리는 시간과 실제 오디오 길이의 비율