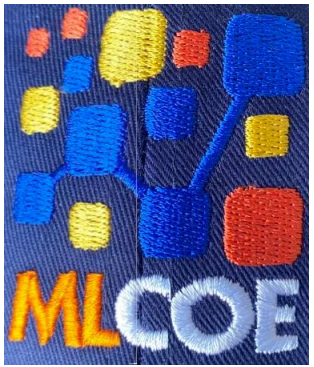
Parameter Averaging is All You Need to Prevent Forgetting

Peter Plantinga, Jaekwon Yoo
Abenezer Girma, Chandra Dhir

JPMorganChase



Acknowledgements — MLCOE Speech Team

Jaekwon Yoo

Abenezer Girma

Chandra Dhir

Strategies for Continual Learning for E2E ASR

Replay [1] — Requires original data, not available for e.g. Whisper

Freezing parameters [2] — Reduced performance vs. full fine-tuning

Loss regularization [3] — Still results in forgetting, if reduced

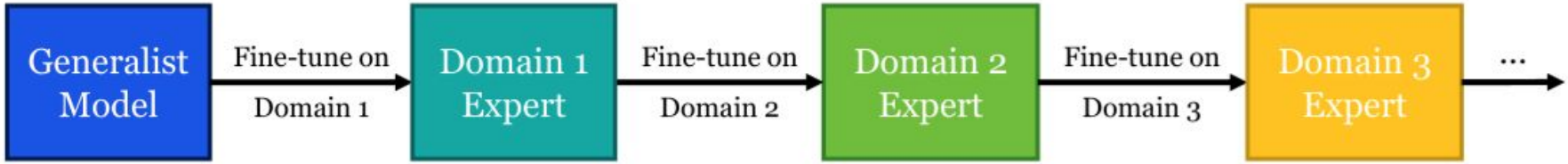
Adapters (e.g. LoRA [4]) — At test time, need to know input domain

Proposed solution

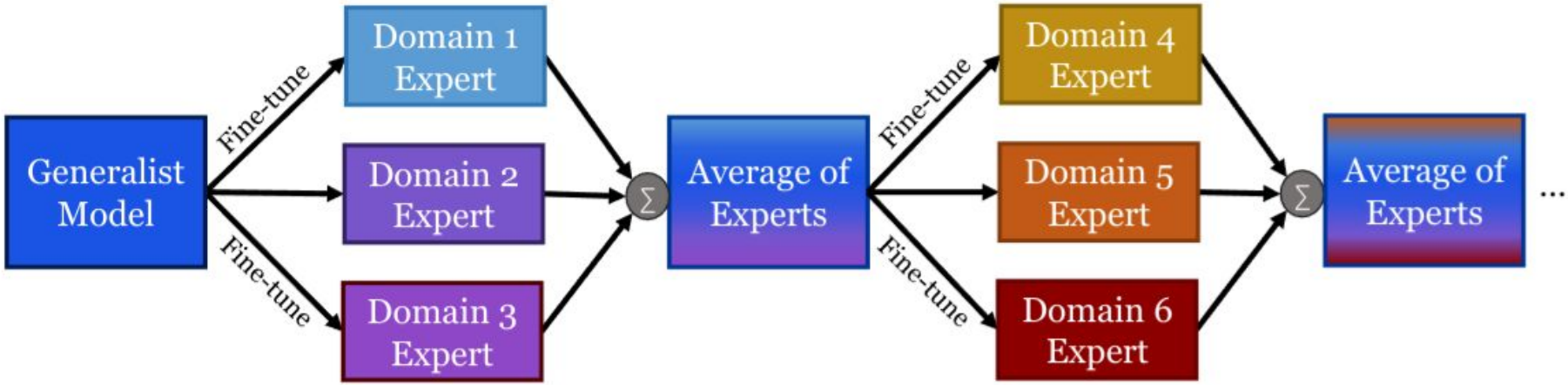
AoDE — Train in parallel on multiple domains, then merge models

Results in single generalized model, almost no forgetting (as low as 0.4%)

Continual Learning



Average of Domain Experts



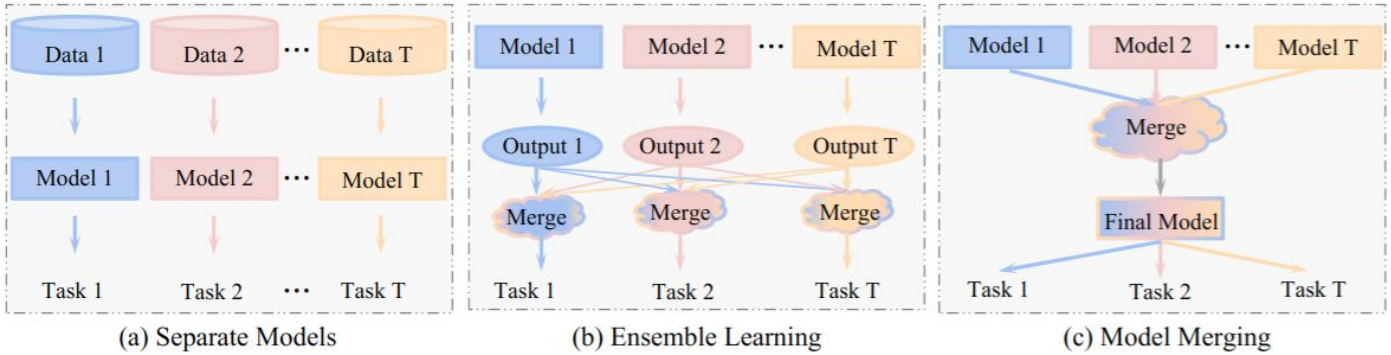
Related Work

Federated learning — distributed model training by averaging [5]

Improving generalization — e.g. stochastic weight averaging [6]

Improving distillation — averaged teachers are better teachers [7]

LLM model merging — SLERP [8], TIES [9], DARE [10], Passthrough [11] …



Yang, Enneng, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. "Model Merging in LLMs, MLLMs, and Beyond: Methods, Theories, Applications and Opportunities." arXiv preprint arXiv:2408.07666 (2024).

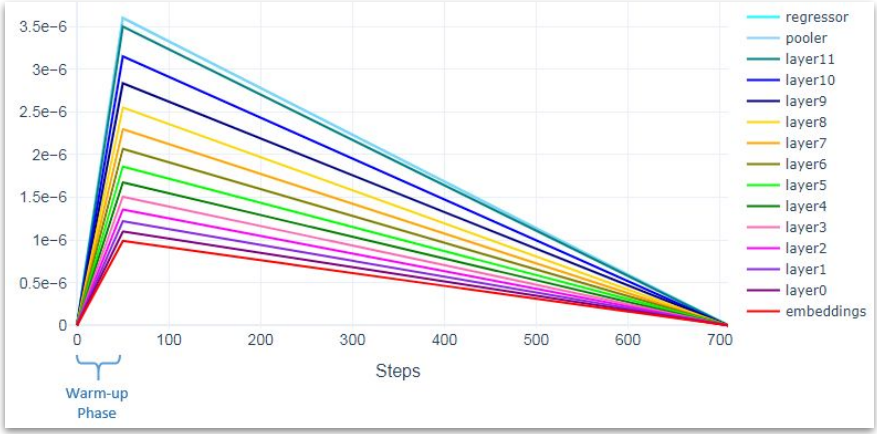
Methods — LLRD

Use layerwise learning rate decay (LLRD), for layer i out of N $\eta_i = \eta_N \cdot \alpha^{(N-i)}$

Matches performance of popular CL techniques: LwF [3], Freezing Dec. [2], etc.

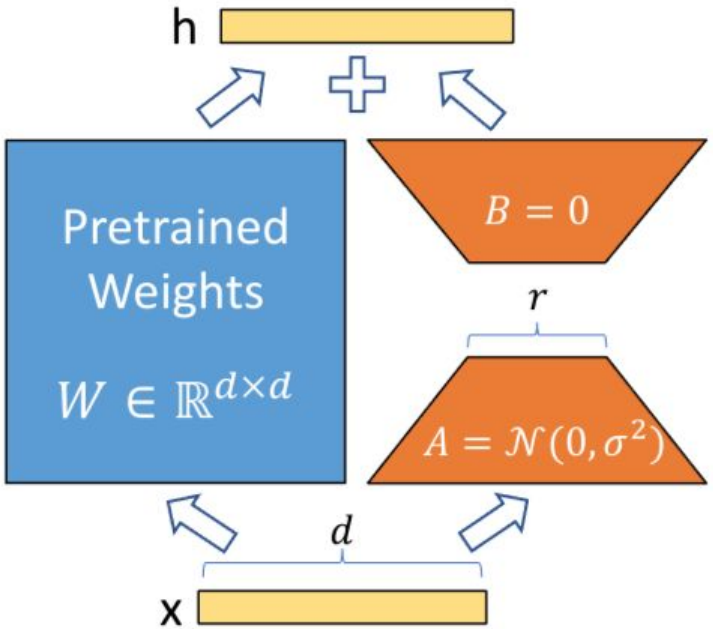
Whisper Small.en

Procedure	CORAAL WER (%)	Forgetting (LS WER ↑)
Pretrained	18.8	0%
FT α = 1.0	14.4	62%
FT α = 0.9	**12.4**	18%
FT α = 0.8	13.2	**6.7%**



Peggy Chang, “Advanced Techniques for Fine-tuning Transformers.” Towards Data Science, 2021.

Methods — LoRA



Hu, Edward J., et al. "Lora: Low-rank adaptation of large language models." arXiv preprint arXiv:2106.09685 (2021).

LoRA can also help to reduce forgetting:

NeMo Parakeet

Procedure	N-VCTK WER (%)	Forgetting (LS WER ↑)
Pretrained	2.26	0%
FT on N-VCTK	**1.62**	120%
LoRA on N-VCTK	1.78	**88%**

Note: this is one example of a general pattern, LoRA has slightly worse performance on target dataset but less forgetting on other data.

Experiments

Fine-tune two generalist E2E ASR models

NeMo Parakeet — 2B params CTC Conformer trained on ~50,000 hours

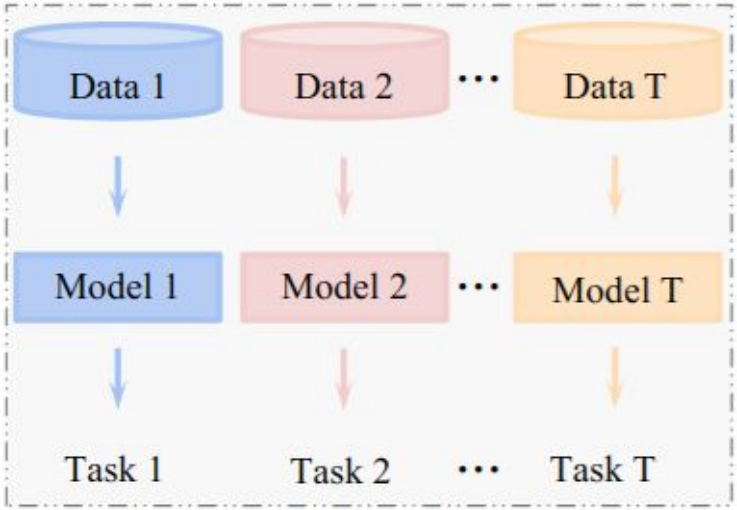
Whisper Small.en — 0.6B params AED Transformer trained on ~500,000 hours

Datasets for fine-tuning:

Dataset	Hours	Domain	Condition
SPGISpeech	5000	Finance	read speech, large vocabulary
CORAAL	140	Everyday	conversational, regional sociolects
Noisy VCTK	20	News	added noise, regional dialects
Google NE	6	News	read speech, regional dialects
DiPCo	6	Everyday	conversational, background noise
JL Corpus	1	Script	emotional speech

Scenario 1 — Fine-tuning

- Direct fine-tuning can sometimes give the best performance on a domain
- If models can be swapped, we can just keep best-performing model
- Leads to high levels of forgetting, even with LLRD



Scenario 1 — Fine-tuning

Procedure	N-VCTK	CORAAL	JL Corpus	Google NE	LS t-clean	LS t-other	Avg.	Forget %
Pretrained	2.3	20.7	4.9	6.7	**2.0**	**3.7**	**6.7**	0%
FT : N-VCTK	**1.6**	33.2	16.8	10.9	4.5	7.8	12.5	120%
FT : CORAAL	3.6	**15.1**	8.0	8.4	2.6	5.2	7.1	38%
FT : JL Corp.	11.4	43.7	**0.3**	15.9	7.3	12.9	15.3	260%
FT : Goog. NE	5.6	25	9.9	**5.3**	3.2	6.8	9.3	79%

Note 1: JL Corpus is smallest, but has largest forgetting rate. We observed that “cleaner” or more narrowly distributed data have more forgetting (e.g. SPGI)

Note 2: None exceed the average performance of the pretrained model

Scenario 2a — Short Retention

- Some laws require regular data deletion e.g. consumer protection laws
- Companies often develop data retention policies — 30 days is common practice
- For privacy, data may not all be available at the same time (i.e. federated learning)

Scenario 2a — Short Retention

Procedure	N-VCTK	CORAAL	JL Corpus	Google NE	LS t-clean	LS t-other	Avg.	Forget %
Pretrained	2.3	20.7	4.9	6.7	2.0	**3.7**	6.7	0%
Sequential →	5.6	18.4	**1.1**	6.2	4.9	8.6	7.5	140%
Sequential ←	**1.5**	26.7	5.9	9.1	4.8	8.7	9.4	140%
AoDE - 4 sets	2.6	**15.3**	3.1	5.9	2.0	4.1	**5.5**	8.9%
AoDE w/ orig	2.0	18.4	3.0	**5.8**	**1.9**	3.9	5.8	3.4%

Note 1: Sequential training tends to do better on more recent datasets

Note 2: Including pre-trained model in average improves WER on LibriSpeech test-clean, without having seen any more data from LibriSpeech domain

Scenario 2b — LoRA + AoDE

```
lora_expert_1.merge_and_unload()
lora_expert_2.merge_and_unload()


avg_of_domain_experts = (
  lora_expert_1.weights
  + lora_expert_2.weights
) / 2
```

- LoRA can be combined with AoDE by averaging models trained with LoRA
- We merge LoRA weights back into model before averaging to avoid dimension mismatch
- We use rank = 16 or 32, scale parameter is 2 x rank

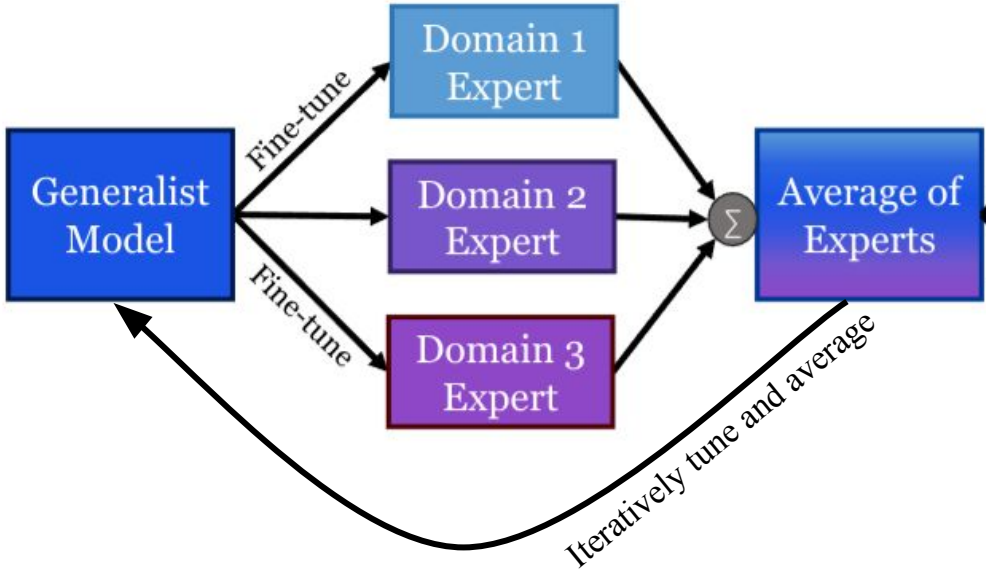
Scenario 2b — LoRA + AoDE

Procedure	N-VCTK	CORAAL	Avg.	Forgetting
Average of FT	2.57	15.3	5.51	8.9%
Average of LoRA	2.07	19.7	6.17	8.6%

- LoRA has slightly better forgetting but overall performance is worse
- We ran more extensive experiments but they all showed similar results

Scenario 3a — All Data Available

- Best case scenario (from ML perspective) is all training data is available all the time
- Enables iterative training schemes, where averaged models are further trained
- Baseline is all data combined into giant dataset with/without resampling to balance sizes



Scenario 3a — All Data Available

Procedure	N-VCTK	CORAAL	JL Corpus	Google NE	LS t-clean	LS t-other	Avg.	Forget %
Combined	3.6	17.5	0.5	8.5	4.1	8.2	7.1	120%
Resampled	2.5	16.7	**0.1**	6.2	3.7	7.2	6.1	94%
AoDE - iter x2	1.9	**15.7**	1.4	5.7	2.0	4.2	5.1	10%
AoDE - iter x3	1.7	15.8	1.2	**5.4**	2.0	4.2	5.1	12%
AoDE x3 w/orig	**1.6**	15.8	1.3	5.6	**1.9**	**4.0**	**5.0**	**5%**

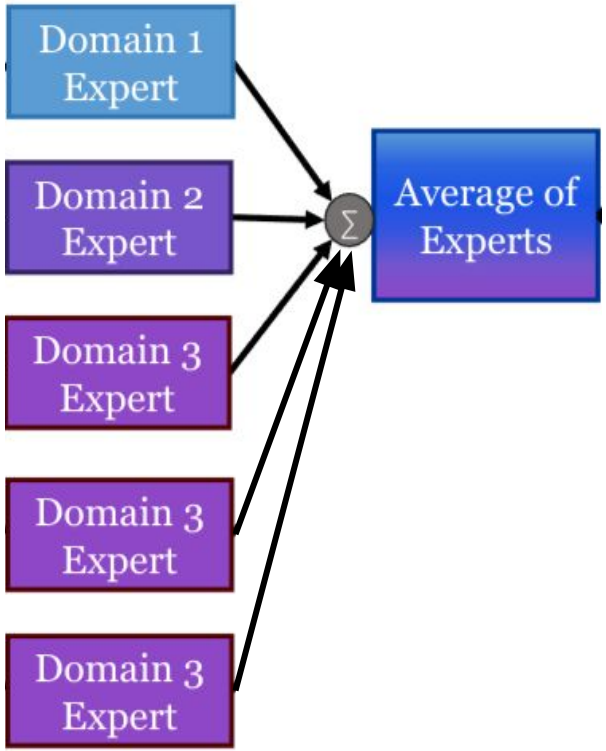
Note 1: 3rd iteration improves on 3 of 4 domains at cost of 2% forgetting

Note 2: Including original in average hurts 2 of 4 domains, 12% → 5% forgetting

Note 3: For “clean” or narrow-domain datasets (e.g. JL Corpus), FT still wins

Scenario 3b — Re-weight Component Models

- We can control degree of expertise in a domain by weighting component models
- For each of the 5 component models (4 experts and original model) we run an experiment weighting the target model x4 compared to the other models, making up ½ of total weight.



Scenario 3b — Re-weight Component Models

Procedure	N-VCTK	CORAAL	JL Corpus	Google NE	LS t-clean	LS t-other	Avg.	Forget %
AoDE balanced	1.6	15.8	1.3	5.6	1.9	4.0	5.0	5.3%
AoDE - N-V x 4	**1.4**	17.5	1.4	6.1	2.1	4.3	5.5	15.0%
AoDE - CO x 4	1.8	**14.7**	1.7	5.5	2.0	4.1	**4.9**	7.1%
AoDE - JL x 4	2.0	16.4	**1.2**	5.8	2.0	4.3	5.3	12.0%
AoDE - NE x 4	1.7	16.2	1.4	**5.3**	2.0	4.1	5.1	9.1%
AoDE - Orig x 4	1.8	17.1	2.4	5.8	**1.8**	**3.8**	5.4	**0.4%**

Note 1: Weighting original model strongly results in **just 0.4% forgetting**

Note 2: Could be dynamically applied, with extra computation and 5x space

Scenario 4 — Oracle Domain

“Interactive Voice Response: IVR Meaning and Benefits” Yeastar.com

- For some use-cases, the domain of the sample may be available at inference time, e.g. a call center uses interactive voice response (IVR) system to redirect calls.
- For such cases, models can be pre-loaded or hot-swapped for a given domain, which is cheaper for adapters (LoRA)

Scenario 4 — Oracle Domain

Procedure	N-VCTK	CORAAL	JL Corpus	Google NE	LS t-clean	LS t-other	Avg.	Forget %
Pretrained	2.3	20.7	4.9	6.7	2.0	3.7	6.7	0%
AoDE - Best	1.6	15.8	1.3	5.6	1.9	4.0	5.0	5%
LoRA hot swap	1.8	15.2	2.5	6.0	2.0	3.7	5.2	0%
FT hot swap	1.6	15.1	**0.3**	**5.3**	2.0	3.7	**4.7**	0%
AoDE hot swap	**1.4**	**14.7**	1.2	**5.3**	**1.8**	3.7	**4.7**	0%

Note 1: AoDE/FT hot swap more costly than LoRA hot swap — need 5x memory

Note 2: AoDE without hot swap outperforms LoRA on average across all data

References

[1] Isele, David, and Akansel Cosgun. "Selective experience replay for lifelong learning." AAAI Artificial Intelligence 2018.

[2] Takashima, Yuki, et al. "Updating only encoders prevents catastrophic forgetting of end-to-end ASR models." arXiv 2022.

[3] Li, Zhizhong, and Derek Hoiem. "Learning without forgetting." Transactions on Pattern Analysis and Machine Intelligence 2017.

[4] Hu, Edward J., et al. "LoRA: Low-rank adaptation of large language models." arXiv 2021.

[5] Wang, Hongyi, et al. "Federated learning with matched averaging." arXiv 2020.

[6] Izmailov, Pavel, et al. "Averaging weights leads to wider optima and better generalization." arXiv 2018.

[7] Tarvainen and Valpola. "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results." NeurIPS 2017.

[8] Shoemake, Ken. "Animating rotation with quaternion curves." Computer Graphics and Interactive Techniques 1985.

[9] Yadav, Prateek, et al. "TIES-merging: Resolving interference when merging models." NeurIPS 2024.

[10] Yu, Le, et al. "Language models are Super Mario: Absorbing abilities from homologous models as a free lunch." ICML 2024.

[11] Kim, Dahyun, et al. "Solar 10.7 B: Scaling large language models with simple yet effective depth up-scaling." arXiv 2023.

Appendix A - Whisper Small.en (Scenario 2)

Procedure	SPGI	CORAAL	DiPCo	Avg.	Forgetting
Pretrained	4.9	18.8	48.5	16.6	-
FT - SPGI	**2.9**	22.6	50.1	18.4	52.3%
FT - CORAAL	4.6	**12.4**	44.3	14.8	17.7%
FT - DiPCo	4.3	18.2	44.0	15.5	**-0.1%**
Sequential FT - SPGI → DiPCo → CORAAL	4.4	13.1	43.3	**14.5**	6.2%
AoDE	3.4	15.7	**43.0**	**14.5**	2.1%

Note 1: Whisper was more prone to forgetting, used LLRD $\alpha = 0.8$

Note 2: Crucially, CORAAL is last in sequential training

Appendix B - Frozen Decoder Baseline

Procedure	SPGI	CORAAL	Forgetting
Pretrained	4.94	18.8	-
Frozen Dec. lr=3e-5	4.17	19.0	10.5%
LLRD=0.9, lr=1e-5	3.14	**18.7**	**9.3%**
LLRD=0.9, lr=3e-5	**2.87**	22.6	52.3%

Note: LLRD beats frozen decoder [2]