GMU Systems for the IWSLT 2025 Low-Resource Speech Translation Shared Task

GEORGE MASON UNIVERSITY

Chutong Meng and Antonios Anastasopoulos
George Mason University

Introduction

- **Background:** Fine-tuning a large pre-trained foundational ST model on a low-resource language pair has been the most prevalent technique. Our objective is to utilize all available data sources to improve model performance under the low-resource setting.

- **Problem:** Simply fine-tuning using the end-to-end (E2E) ST objective has three potential drawbacks:
  - The E2E ST data size is too small;
  - The available ASR and/or MT datasets are not used;
  - The foundation model may not have been pre-trained on this language.

- **Solution:** Besides traditional E2E and cascaded ST approaches, we tried
  - In-domain pre-training with ASR/MT objectives;
  - Multi-task fine-tuning, hoping the stronger MT teacher can help with ST performance.

Method

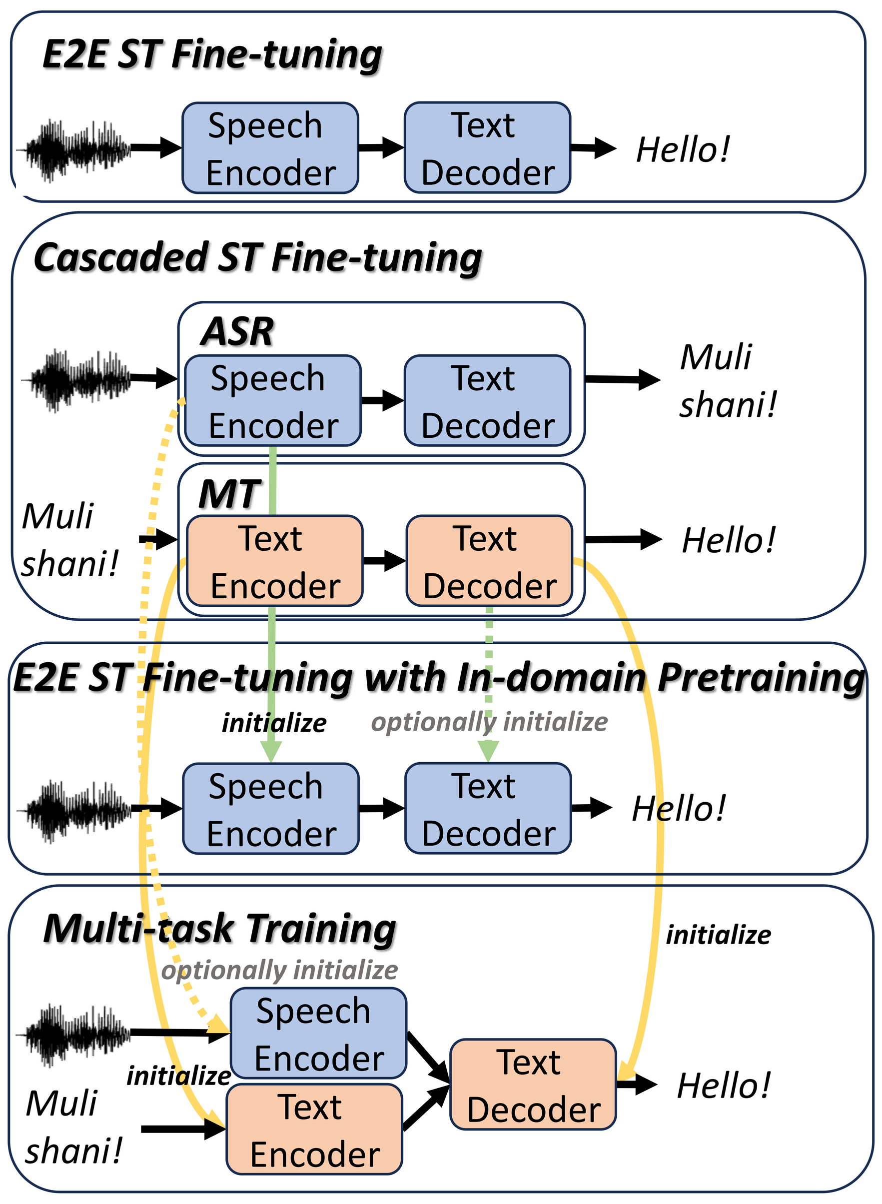

Figure 1: Four fine-tuning strategies. Encoders and decoders refer to the base model components.

- **Base Model**: SeamlessM4T-v2-Large.
- **Terminology**:
  - x^{sp}, x^{text}, y: source speech, source text, target text
  - θ_{se}, θ_{te}, θ_{td}: speech encoder, text encoder, text decoder
  - $\theta_{\text{se}}^{\text{ASR}}$ and $\theta_{\text{td}}^{\text{ASR}}$: speech encoder and text decoder fine-tuned by the ASR objective
  - $\theta_{\text{te}}^{\text{MT}}$ and $\theta_{\text{td}}^{\text{MT}}$: text encoder and decoder fine-tuned by the MT objective
- **Training Objectives**
  - **E2E ST Fine-tuning:** $L_{\text{E2E}} = -\frac{1}{|y|}\log p(y|x^{\text{sp}}; \theta_{\text{se}}, \theta_{\text{td}})$
  - **Cascaded ST Fine-tuning/In-domain pre-training:**
    - ASR objective: $L_{\text{ASR}} = -\frac{1}{|x^{\text{text}}|}\log p(x^{\text{text}}|x^{\text{sp}}; \theta_{\text{se}}, \theta_{\text{td}})$
    - MT objective: $L_{\text{MT}} = -\frac{1}{|y|}\log p(y|x^{\text{text}}; \theta_{\text{te}}, \theta_{\text{td}})$
    - The obtained $\theta_{\text{se}}^{\text{ASR}}$ and $\theta_{\text{td}}^{\text{MT}}$ can be used to init E2E ST fine-tuning
  - **Multi-task Fine-tuning:**
    - The output from the MT teacher: $p_{\text{teacher}}(\cdot|y_{<i}, x^{\text{text}}) = \text{stop-gradient}\left(p(\cdot|y_{<i}, x^{\text{text}}; \theta_{\text{te}}, \theta_{\text{td}})\right)$
    - The knowledge distillation objective: $L_{\text{KD}} = \frac{1}{|y|}\sum_{i=1}^{|y|} D_{\text{KL}}\left[p_{\text{teacher}}(\cdot|y_{<i}, x^{\text{text}}) || p(\cdot|y_{<i}, x^{\text{sp}}; \theta_{\text{se}}, \theta_{\text{td}})\right]$
    - The overall loss: $L = \alpha \cdot L_{\text{E2E}} + \beta \cdot L_{\text{MT}} + \gamma \cdot L_{\text{KD}}$

Main Results

Lang	System	Dev
aeb	E2E	22.73
	E2E-ASR$_{\text{init}}$	**25.48**
	E2E-ASR$_{\text{init}}$-MT$_{\text{init}}$	24.08
	MLT	24.23
	MLT-ASR$_{\text{init}}$	24.64
	Cascaded	24.42
bem	E2E	31.14
	E2E-ASR$_{\text{init}}$	**31.96**
	Cascaded	28.02
fon	E2E	**40.86**
gle	E2E	**24.07**
	E2E-ASR$_{\text{init}}$	23.34
bho	E2E	33.92
	E2E-ASR$_{\text{init}}$	**39.04**

Table 1: BLEU scores on dev sets.

Lang	System	Dev
est	E2E	36.89
	E2E-ASR$_{\text{init}}$	36.97
	Cascaded	**38.00**
mlt	E2E	**57.65**
	E2E-ASR$_{\text{init}}$	57.57
	MLT	57.46
	Cascaded	57.04
mar	E2E	**44.84**
	E2E-ASR$_{\text{init}}$	44.72
que	E2E	12.32
	E2E-ASR$_{\text{init}}$	13.00
	E2E-ASR$_{\text{init}}$-MT$_{\text{init}}$	**13.37**
	MLT-ASR$_{\text{init}}$	13.03
	Cascaded	13.15

Table 2: BLEU scores on dev sets.

- E2E fine-tuning performs best for languages that SeamlessM4T-v2 has ASR support: gle, est, mlt, mar.
- In-domain ASR pre-training improves performance for languages without ASR support: aeb, bem, bho, que.
- In-domain MT pre-training is not so helpful as ASR.
- Multi-task training (MLT) performs better than E2E when MT performance is strong: aeb, mlt, que.
- Cascaded systems are competitive but generally underperform E2E training: aeb, bem.

Codebase Matters

The official and the HuggingFace models have different default behaviors. More details can be found in Appendix A.

Lang	OFF E2E Dev	HF E2E Dev
aeb	**23.76**	22.73
bem	30.69	**31.14**
gle	**29.63**	24.07
bho	**41.96**	33.92
est	38.07	36.89
mlt	57.92	57.65
mar	42.52	44.84

Table 3: Comparison between the official (OFF) and HuggingFace (HF) codebases.

Additional Data Substantially Improves Que

Adding additional ASR/MT/ST data is especially important for que, which has only 1.67 hours of official E2E ST data.

Datasets	Dev ASR CER
IWSLT2025	19.19
+Huqariq	16.97
+Siminchik	**15.54**

Datasets	Dev MT BLEU
IWSLT2025	5.88
+Huqariq+JW300+Hinantin	14.38
+ NLLB	**15.29**

Datasets	System	Dev ST BLEU
IWSLT2025	E2E	3.73
	E2E-ASR$_{\text{init}}$	9.84
	E2E-ASR$_{\text{init}}$-MT$_{\text{init}}$	10.42
+Huqariq	E2E	12.32
	E2E-ASR$_{\text{init}}$	13.00
	E2E-ASR$_{\text{init}}$-MT$_{\text{init}}$	**13.37**

Conclusion

- E2E fine-tuning (with in-domain ASR pre-training) performs best.
- Adding more data is generally bebeficial.