KNOWLEDGE DISTILLATION ON JOINT TASK END-TO-END SPEECH TRANSLATION

Khandokar Md Nayem[§], Ran Xue[†], Ching-Yun Chang[†], Akshaya Vishnu Kudlu Shanbhogue[†]

[§]Indiana University, [†]Amazon Alexa AI

OVERVIEW

An End-to-End Speech Translation (E2E-ST) model takes input audio in one language and directly produces output text in another language. The model demands a large architecture to jointly learn speech-to-text modality conversion and translation tasks. We present a pilot work on optimizing model compression for a cross-modality joint-task E2E-ST system with knowledge distillation (KD).

KNOWLEDGE DISTILLATION ON A COMPLEX SYSTEM

Model Architecture Our baseline model is an E2E-ST system which takes both speech and text as input for speech translation model training (Tang et al. 2021).

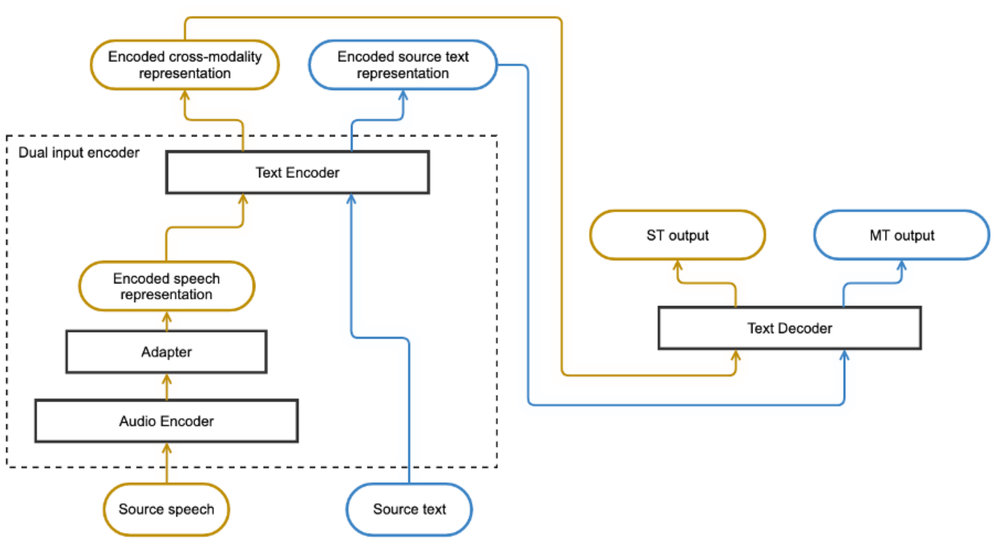

Figure 1. Baseline end to end speech translation system.

The baseline model has 24 transformer layers in the encoder and 12 transformer layers in the decoder (616.33M parameters). To compress the model size, we reduce the total number of transformer layers in the speech encoder, text encoder, and decoder by 50% (364.4M parameters).

KD Loss Function To compress a cross-modality system with KD, we added KD loss terms $\mathcal{L}_{sph\_kd}$ and $\mathcal{L}_{txt\_kd}$ for speech and text modality respectively. And specifically, there are response-based, feature-based, and relation-based KD loss:

$$\mathcal{L} = \overbrace{(\mathcal{L}_{sph} + \mathcal{L}_{t\_guide}) + \mathcal{L}_{sph\_kd}}^{total\ speech\ loss} + \overbrace{\mathcal{L}_{txt} + \mathcal{L}_{txt\_kd}}^{total\ text\ loss} + \mathcal{L}_{cross\_attn} \quad (1)$$

$$\mathcal{L}_{X\_kd} = \mathcal{L}^{X}_{kd\_res} + \mathcal{L}^{X}_{kd\_feat} + \mathcal{L}^{X}_{kd\_rel} \quad (2)$$

$$\mathcal{L}^{X}_{kd\_res} = \sum_{l \in \{L_{enc}, L_{dec}\}} D(P(X^B_l), P(X^M_l)) \quad (3)$$

$$\mathcal{L}^{X}_{kd\_feat} = \sum_{l \in L} (1 - \cos(lay(X^B_l), lay(X^M_l))) \quad (4)$$

$$\mathcal{L}^{X}_{kd\_rel} = \sum_{l \in L} (1 - \cos(attn(X^B_l), attn(X^M_l))) \quad (5)$$

D represents the dynamic dual-skew divergence which measures the gap between teacher and student model output distribution Shanbhogue et al. 2022.

KD Training Scheme Previous study in KD for machine translation found that training layers incrementally with KD loss performs well (Aguilar et al. 2020). Following this insight, we investigate three training schemes where student models are trained in three stages - 1) Sph encoder, 2) Sph encoder+Txt encoder, and 3) Sph encoder+Txt encoder+Decoder - at different pace.

↓Scheme	↓Stage	KD loss			Speech & Guide loss	Text loss
		Sph encoder	Txt encoder	Decoder		
Module	Sph encoder	√	×	×	√	√
	Sph encoder + Txt encoder	√	√	×	√	√
	Sph encoder + Txt encoder + Decoder	√	√	√	√	√
Task	Sph encoder	√	×	×	×	×
	Sph encoder + Txt encoder	√	√	×	×	√
	Sph encoder + Txt encoder + Decoder	√	√	√	√	√
All	Same in all 3 stages	√	√	√	√	√

Table 1. Different training progression schemes. Active and inactive loss terms are denoted using $\surd$ and $\times$, respectively.

RESULTS

Model Structure & Weight Initialization We compared compressed models with different structures and initialization strategies:

- At 50% compression rate, ST model with evenly distributed number of layers (6-6-6) marginally outperforms a shallow decoder model (8-8-2) on speech BLEU by 1.0% relatively.
- Compressed model initialized with pre-trained Wav2Vec 2.0 and mBART model weights yields better performance than model initialized from teacher ST model on speech BLEU by 1.6% relatively.

KD Loss Terms We conduct experiments to evaluate the effectiveness of individual KD loss and overall performance gain in a compressed model. The model $M : \mathcal{L}_{kd\_res}(enc)$ performs the best compared to the other M models and the non-KD compressed model C.

↓ Models	BLEU↑ [sph]	Degradation (%)↓ [sph]	BLEU ↑ [txt]	Degradation (%) ↓ [txt]
Baselines				
B	28.38	-	32.94	-
C	23.34	17.76	30.78	6.56
Ablation study on individual KD loss terms				
M	23.24	18.11	31.03	5.8
$M : \mathcal{L}_{kd\_res}(dec)$	23.26	18.04	31.22	5.22
$M : \mathcal{L}_{kd\_res}(enc)$	**23.78**	**16.21**	**31.31**	**4.95**
$M : \mathcal{L}_{kd\_feat}$	23.69	16.53	31.05	5.74
$M : \mathcal{L}_{kd\_rel}$	21.66	23.68	29.98	8.99
Performance of student KD models				
$M : \mathcal{L}_{kd\_res}$	**24.72**	**12.9**	**32.15**	**2.4**
$M : \mathcal{L}_{kd\_res}(enc) + \mathcal{L}_{kd\_feat}$	24.31	14.34	31.34	4.86
$M : \mathcal{L}_{kd\_res}(enc) + +\mathcal{L}_{kd\_avg(feat)}$	24.14	14.94	31.6	4.07
$M : \mathcal{L}_{kd\_res} + \mathcal{L}_{kd\_feat}$	24.38	14.09	31.58	4.13
Fine-tune the best-performing $M : \mathcal{L}_{kd\_res}$ with S2T and T2T loss				
$M^* : \mathcal{L}_{kd\_res}$	**25.33**	**10.75**	**32.07**	**2.64**

*Table 2. Performance of student models using KD losses. Best shown in **bold**. B denotes baseline full-sized model, C denotes compressed model trained without KD loss, and M denotes student ST models optimized using KD loss terms.*

KD Training Scheme We use the best-performing $M : \mathcal{L}_{kd\_res}$ model configuration to investigate three progressive training schemes. We observe that the model $M : task$ outperforms all the other training schemes before fine-tuning. However, after fine-tuning, the fine-tuned $M^* : task$ model did not outperform the fine-tuned $M^* : \mathcal{L}_{kd\_res}$ model.

↓ Models	BLEU↑ [sph]	Degradation (%)↓ [sph]	BLEU ↑ [txt]	Degradation (%) ↓ [txt]
Baselines				
$M : \mathcal{L}_{kd\_res}$	24.72	12.9	32.15	2.4
$M^* : \mathcal{L}_{kd\_res}$	25.33	10.75	32.07	2.64
Performance of using different training schemes				
$M : module$	24.47	13.78	31.77	3.55
$M : task$	**25.11**	**11.52**	31.69	3.79
$M : all$	24.4	14.02	**31.93**	**3.07**
Fine-tune with S2T and T2T loss				
$M^* : module$	24.78	12.68	31.64	3.95
$M^* : task$	**25.24**	**11.06**	31.67	3.86
$M^* : all$	25.16	11.35	**31.72**	**3.7**

*Table 3. Performance of student models using $\mathcal{L}_{kd\_res}$ loss and trained with different training scheme. Best shown in **bold**.*

CONCLUSION

- We conduct a detailed study on compressing joint E2E speech translation models using knowledge distillation techniques. Our best result presents only **10.75% relative BLEU degradation with 50% model compression rates** on English to German translation.
- Among various KD loss terms, response-based KD (both encoder and decoder KD loss) gives the best performance in both speech and text BLEU scores.
- We observe incremental KD training improves performance only before finetuning. This may be due to the incremental training method placing more emphasis on teacher-student knowledge transfer than on achieving optimal S2T performance.

Contact: *knayem@iu.edu, {ranxue,cychang,ashanbho}@amazon.com*