Improving End-to-End Speech Translation with Progressive Dual Encoding

Runlai Zhang[1], SaiHan Chen[1], Yuhao Zhang[1], Yangfan Du[1], Hao Chen[1], Tong Xiao[1,2] *, and Jingbo Zhu[1,2]

[1] School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2] NiuTrans Research, Shenyang, China
{mehenrychang, christinechensh, yoohao.zhang, dduyangfan}@gmail.com, {xiaotong, zhujingbo}@mail.neu.edu.cn

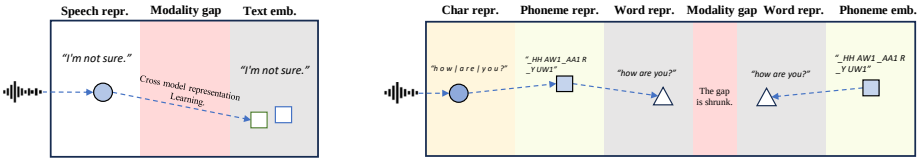
Abstract. In end-to-end speech translation (E2E ST), multi-task learning is often applied due to the scarcity of labeled ST data. However, the modality gap between speech and source text poses a significant challenge for transferring knowledge from machine translation (MT) models to speech translation (ST) models. Currently, one of the main approaches to address this problem focuses on utilizing a powerful self-supervised pretrained speech encoder to learn the text embeddings from the MT model. This method involves cross-modal representation learning and heavily relies on the capability of the speech encoder. However, pretraining such a speech encoder is expensive and difficult to replicate. To this end, we propose the **P**rogressive **D**ual **E**ncoding (**PDE**) method, which aims to provide a lightweight, pretraining-free approach for E2E ST encoding. This is achieved by introducing an additional text encoder that collaborates with the speech encoder to progressively explore a shared representation space under a novel multi-scale constraint. In this way, our method allows the dual-modality encoders to naturally output similar representations, thereby avoiding the challenges of cross-modal representation learning and alleviating the burden on the speech encoder. We evaluate our method on the MuST-C dataset. The experimental results demonstrate that our method achieves competitive performance with other methods while using fewer parameters in the speech encoder. Moreover, it surpasses other methods by 0.6-1.8 average BLEU score when the number of parameters is increased.

Keywords: End-to-End Speech Translation · Progressive Dual Encoding · Multi-scale Constraint.

1 Introduction

End-to-End Speech Translation (E2E ST) is an advanced technology that can directly translate spoken input in one language into text in another language.

* Corresponding author
Code is available at https://github.com/HenryChang666/PDE.



(a) (b)

Fig. 1: (a) illustrates previous approaches to addressing the modality gap, where the speech encoder needs to learn a cross-modal mapping from speech features to text embeddings. (b) demonstrates our method, where we leverage the speech encoder and text encoder to synchronously and progressively extract low-level representations, ultimately aligning features at the word level. This reduces the modality gap and avoids the need for cross-modal representation learning.

So far, the performance of E2E ST can rival or even surpass that of traditional cascaded systems [24]. However, E2E ST faces the scarcity of labeled ST data. To overcome this issue, researchers typically employ techniques such as pretraining [2, 17, 21, 22] and multi-task learning [16] to augment the model with other data sources.

As a cross-modal task, transferring knowledge from MT models to ST models is limited by the modality gap. Cross-modal representation learning [12, 16, 25, 26] is often used to address this problem. Representation learning typically adopts techniques such as contrastive learning [12, 25] and Cross-Attentive Regularization [16, 26]. These techniques significantly alleviate the modality gap problem by enabling the speech encoder to generate speech representations that are similar or correlated to the text input (typically text embeddings) of the auxiliary MT task. However, these methods require cross-modal representation learning, which heavily rely on the capability of the speech encoder and increase its burden. The speech encoder processes audio signals into highly abstract and complex representations, capturing intricate details like tone, pitch, and rhythm, as well as the semantic information of the whole sentence, which are not present in text embeddings. Therefore, speech features and text embeddings inherently have substantial representational differences. These differences make cross-modal representation learning challenging.

On the other hand, Self-supervised learned models, like wav2vec 2.0 [1] and HuBERT [6], have been widely used to initialize ST models' speech encoders [12, 24, 25]. Though these large pretrained speech encoders significantly improve the performance of E2E ST, the cost associated with pretraining these speech encoders is substantial. For example, the wav2vec 2.0 base requires 64 V100 GPUs for 1.6 days of training [1]. Consequently, replicating these models' performance on other source languages is challenging due to the lack of equivalent data and the high computational costs.

For these reasons, we propose **P**rogressive **D**ual **E**ncoding (**PDE**), a method that alleviates the burden on the speech encoder by adding an additional text encoder to produce a text feature consistent with the speech feature. This method

highly unleashes the potential of light-weight parameter encoders by leveraging multi-scale information from the corpus to guide them in learning a unified high-level semantic representation progressively. In particular, we harness three levels of granularity: character, phoneme, and word, to progressively guide the speech encoder to converge at the word-level. Synchronously, we utilize phoneme text as input for an additional text encoder, which also converges at the word level under the supervision of word information.

Our contributions are summarized as follows:

• We propose PDE, a lightweight, simple yet effective method for E2E ST encoding. PDE encourages the speech encoder and text encoder to independently explore similar representations in a progressive manner, avoiding cross-modal representation learning and reducing the burden on the speech encoder.

• Our method, which focuses on modeling encoders, is orthogonal to many other advanced E2E ST modeling methods.

• Results on the MuST-C benchmark demonstrate that our model, with the Base configuration, achieves performance competitive with other models that use self-supervised pretrained speech encoders. With the Large configuration, our model even surpasses them.

2 Method

A speech translation corpus typically comprises triplets of speech, transcription, and translation, denoted as (s, x, y) , where s is the original audio, x is the transcription, and y is the translation. The task of E2E ST is to build a model capable of directly converting the input audio s into the translation y.

2.1 Model Architecture

As shown in Figure.2, our model can be decomposed into multi-scale speech/text encoders, a translation encoder, and a translation decoder. The original audio s is fed into the speech encoder, and the decoder ultimately generates the predicted translation y. Simultaneously, a phoneme text p, constructed from the word-level transcription x, is fed into the text encoder, the predicted translation y is also produced by the decoder.

Multi-Scale Speech Encoder. Inspired by [23], we develop a novel speech feature extraction method that performs progressive compression with the help of the multi-scale constraint. We construct the backbone of the speech encoder with vanilla transformer layers [18]. As shown in Figure.2, we decompose the entire speech encoder into three levels: the char encoder, phoneme encoder, and word encoder. This hierarchical structure enables the gradual extraction of speech feature information, progressing from fine-grained to coarse-grained levels, under the constraint of multi-scale CTC loss. (The multi-scale constraint applied in this process will be explained in detail later.) Multiple 1-D convolutional layers are inserted between different levels of encoders to facilitate efficient progressive

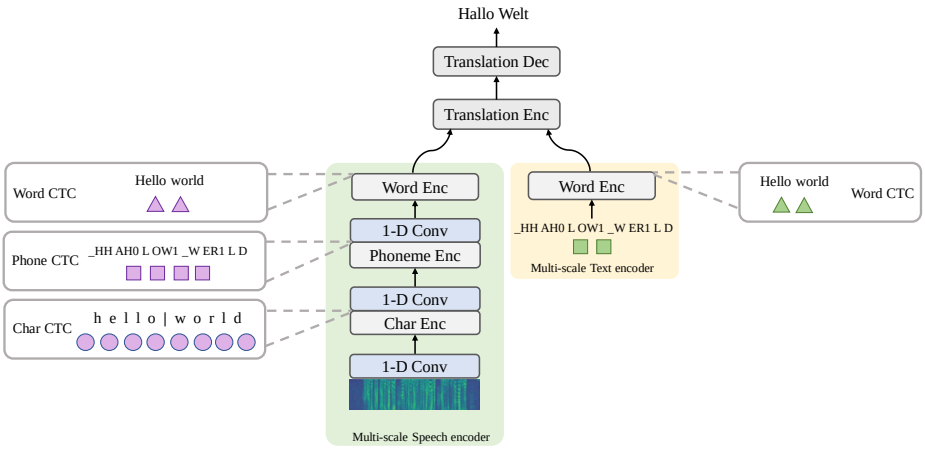


Fig. 2: The overall architecture of our model.

down-sampling of the speech features. In total, we apply three 1-D convolutions, achieving an 8 times compression of the input speech features.

Prior to downsampling at different layers, we propose using CTC loss [5] to align the speech features at each layer with the corresponding scale of transcriptions. To be more specific, we utilize the transcriptions at three different granularities – character c, phoneme p, and word x to guide compression at three granularity levels, respectively. In this case, the triplet (s, x, y) from the ST data is expanded to a quintuplet (s, c, p, x, y). This approach offers two advantages: Firstly, as speech signals are temporal sequences, adjacent samples in time exhibit strong correlation and dependency [15]. Therefore, fine-grain signals in speech are prone to excessive compression [23]. By guiding the compression process with the proposed multi-scale constraint, we effectively mitigate the issue of excessive compression. Secondly, by continuously emphasizing the semantic information throughout multiple layers of feature extraction, we ensure the integrity and consistency of the semantic content within the features. This promotes the speech encoder's ability to output higher-quality word-level features at the top encoder layer, facilitating better alignment with the word-level features produced by the text encoder (which will be discussed later).

Multi-Scale Text Encoder. As we mentioned in the Introduction, the information distribution of vanilla text embeddings is largely different from that of speech features. This means that directly aligning speech features with text embeddings risks losing original information. To address this, our solution is to insert a single-layer transformer encoder between the translation encoder and the input embeddings, constrained by word-level CTC to provide semantic guidance. Additionally, we replace the generic word embeddings with phoneme-level text (converted from word text), which originally has an information distribution closer to spoken speech.

With the aforementioned Progressive Dual Encoding method, the speech and text features generated by the encoders can naturally align in terms of semantics

and information distribution, without the need for cross-modal constraints such as contrastive loss.

Translation Encoder and Decoder. The translation encoder and translation decoder are shared by both the ST task and the auxiliary MT task, and they are also composed of multiple transformer layers. In the ST task, the translation encoder receives speech features from the speech encoder, while in the MT task, it receives text features from the text encoder. After passing through multiple transformer encoder layers, the resulting representation is fed into the translation decoder to finally obtain the translation.

2.2 Training Objective

We adopt a two-stage training approach. First, we train the text encoder, translation encoder, and translation decoder using external MT data. Then, we perform multi-task learning using the quintuplet (s, c, p, x, y) from the ST data. Specifically, we input the phoneme text p into the text encoder for the auxiliary MT task and input the speech s into the speech encoder for the ST task. To prevent catastrophic forgetting, we randomly sample from the external MT data to perform additional MT training simultaneously.

As described in Section 2.1, we introduce a multi-scale constraint to align speech features at different levels with the sequences of varying granularity in the triplet (c, p, x). The multi-scale CTC loss can be calculated as:

$$\begin{aligned}\mathcal{L}_{\mathrm{MS-ASR}} = &-\log \mathrm{P}_{\mathrm{CTC}}\left(c|s\right) - \log \mathrm{P}_{\mathrm{CTC}}\left(p|s\right) - \\ &\log \mathrm{P}_{\mathrm{CTC}}\left(x|s\right) - \log \mathrm{P}_{\mathrm{CTC}}\left(x|p\right)\end{aligned} \tag{1}$$

the CTC probabilities can be calculated as:

$$\mathrm{P}_{\mathrm{CTC}}\left(c|s\right) = \sum_{\pi \epsilon \mathcal{B}^{-1}(c)} \mathrm{P}\left(\pi|s\right) \tag{2}$$

$$\mathrm{P}_{\mathrm{CTC}}\left(p|s\right) = \sum_{\pi \epsilon \mathcal{B}^{-1}(p)} \mathrm{P}\left(\pi|s\right) \tag{3}$$

$$\mathrm{P}_{\mathrm{CTC}}\left(x|s\right) = \sum_{\pi \epsilon \mathcal{B}^{-1}(x)} \mathrm{P}\left(\pi|s\right) \tag{4}$$

$$\mathrm{P}_{\mathrm{CTC}}\left(x|p\right) = \sum_{\pi \epsilon \mathcal{B}^{-1}(x)} \mathrm{P}\left(\pi|p\right) \tag{5}$$

where $\mathcal{B}$ maps an alignment sequence π to c, p or x. (2-4) are applied on the speech encoder, and (5) is applied on the text encoder. The training objectives of MT and ST tasks are auto-regressive training as follow:

$$\mathcal{L}_{\mathrm{MT}} = -\sum_{i=0}^{|y|} \log \mathrm{P}\left(y_i|x, y_{1:i-1}\right) \tag{6}$$

$$\mathcal{L}_{\mathrm{ST}} = -\sum_{i=0}^{|y|} \log \mathrm{P}\left(y_i | s, y_{1:i-1}\right) \tag{7}$$

where the $|y|$ denotes the length of the target sequence. Thus the total training objective can be described as follows:

$$\mathcal{L} = \mathcal{L}_{\mathrm{ST}} + \mathcal{L}_{\mathrm{MT}} + \lambda \mathcal{L}_{\mathrm{MS-ASR}} \tag{8}$$

where the λ is a predefined hyper-parameter.

2.3 Mixing Speech and Text Representations at the Phoneme Level

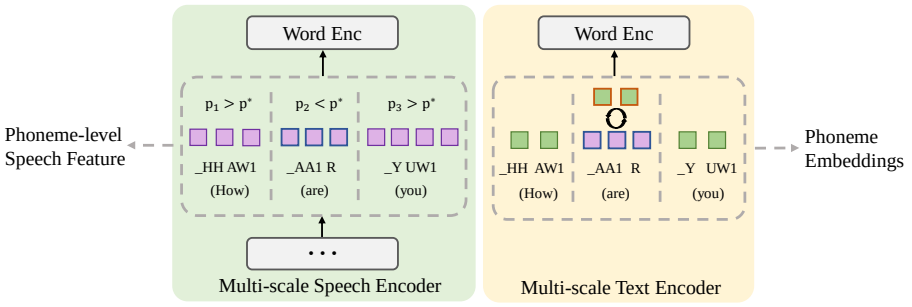


Fig. 3: We mix up phone embeddings with phoneme-level speech representations.

There is an inherent representational difference between speech features and text embeddings. The mixup method [4, 10, 19, 26] has already demonstrated its effectiveness in reducing representational differences between different modalities. Inspired by this, we mix speech features into the embeddings input to the text encoder to further reduce the modality gap between the two modalities.

Specifically, as shown in Figure.3, we use the Montreal Forced Aligner [11] to perform word-level forced alignment between speech representation and text, providing the start and end frames of each word in the speech features. For each word in a sentence, we extract frames from the speech encoder corresponding to the word with a certain probability p^* and replace the phoneme embeddings in the text encoder corresponding to the word.

※ It should be noted that we incorporated a simple knowledge distillation method proposed in JT-S-MT [16] into our model. However, this is not our main contribution, so we do not describe it in detail due to space limitations.

3 Experiment

3.1 Datasets

ST Datasets: We conduct our experiments on two language pairs (En-De,

	ST		MT	
	hours	sentences	name	sentences
En-De v3	440	270k	WMT16	4.6M
En-De v1	408	234k	WMT16	4.6M
En-Fr v1	492	280k	WMT14	40.8M

Table 1: Statistics of datasets

En-Fr) of the MuST-C v1.0 corpus [3]. MuST-C corpus is a multilingual speech translation dataset, which contains audio recordings from TED Talks along with their transcriptions and translations.

We also conduct experiments on the MuST-C v3.0 En-De dataset. The v3.0 En-De dataset includes more training data and is of higher-quality (e.g., audio-text misalignments have been fixed). This allows us to observe how our method performs on higher quality data.

External MT Datasets: We use data from WMT data for the En-De, and En-Fr language pairs as external MT training data to pretrain the text encoder, translation encoder, and translation decoder.

3.2 Experimental settings

Preprocessing settings: Following the setup of [16], we use 80-dimensional log mel-filterbank coefficients computed every 10ms with a 25ms window as audio input. Global channel mean and variance normalization as well as SpecAugment [13] data augmentation are applied. We use the `"g2p_en"` Python package [8] to generate phoneme-level text as input to the text encoder. Character-level text is generated by simply decomposing words into letters and manually inserting the "|" symbol between words. The vocabulary sizes of the phoneme and character texts are 135 and 38, respectively. The vocabulary of word text consists of 10k "unigram" word units learned by SentencePiece [7].

Model settings: We use transformer layers to construct all encoder and decoder layers, following the base configuration. For the translation encoder, we use 6 transformer encoder layers. For the translation decoder, we use 6 transformer decoder layers. For the text encoder, we use a single transformer encoder layer. As for the speech encoder, to investigate the difference in feature extraction capability based on the number of parameters, we provide two configurations: Base and Large, which contain 6 and 18 transformer encoder layers, respectively. All transformer layers consist of 512 hidden units, 8 attention heads, and 2048 feed-forward hidden units.

For the multi-scale CTC constraint on the speech encoder, we introduce char, phoneme, and word-level CTC loss at layers (3,5,6) for the Base configuration and at layers (9,15,18) for the Large configuration of the speech encoder. The hyper-parameter λ is set to 0.2. Each convolution layer of the speech encoder has a stride of 2 and a kernel size of 5. The mixup probability is set to 0.4.

Training and inferencing settings: Our model is trained with FAIRSEQ [20]. We pretrain the MT model on external MT data, using the Adam optimizer with

Models	Params	External Data			BLEU			
		Speech	ASR	MT	De(v3)	De	Fr	Avg
w/ self-supervised pretrained speech encoder								
XSTNet [24]	150M	✓	-	✓	-	27.8	38.0	32.9
ConST [25]	150M	✓	-	✓	-	28.3	38.3	33.3
WACO [12]	150M	✓	-	✓	-	28.1	38.1	33.1
STEMM [4]	150M	✓	-	✓	-	28.7	37.4	33.1
w/o self-supervised pretrained speech encoder								
JT-S-MT [16]	76M	-	-	✓	27.1*	26.8	37.4	32.1
SATE [22]	150M	-	✓	✓	-	28.1†	-	-
PDE-Base	84M	-	-	✓	28.6	28.1	38.1	33.1
PDE-Large	120M	-	-	✓	**29.2**	**28.8**	**39.0**	**33.9**

Table 2: Case-sensitive detokenized BLEU scores on MuST-C tst-COMMON set. "Speech" refers to unlabeled speech data. "ASR" refers to external ASR data. "MT" refers to external MT data. $*$ indicates our reproduced results. † use 40M OpenSubtitles [9] as external MT data. Avg indicates the average score on the MuST-C v1.0 dataset.

a learning rate of 5e-4, 4000 warm-up updates and 1e-4 weight decay. The ST model is trained with ST data, using the Adam optimizer with a learning rate of 2e-3, 20000 warm-up updates. The β_1, β_2 of Adam optimizer are set to (0.9, 0.999). The dropout is set to 0.1. The label smoothing is set to 0.1. Experiments were conducted on NVIDIA RTX 3090 GPUs or NVIDIA TITAN RTX GPUs.

At the inferencing stage, we average the last 10 checkpoints for evaluation. The beam size of beam search is set to 5. We use the sacreBLEU [14] for computing BLEU scores.

Baselines: We introduced several typical E2E ST systems as comparisons to demonstrate the effectiveness of our method, including JT-S-MT [16] and SATE [22], which do not use self-supervised pretraining for the speech encoder, as well as XSTNet [24], ConST [25], and WACO [12], STEMM [4] which do.

It is worth mentioning that JT-S-MT has a similar structure to our Base model, consisting of a standard 12-layer Transformer encoder and a 6-layer Transformer decoder forming the backbone of the E2E ST system. The distinctive feature of JT-S-MT is the use of Cross-Attentive Regularization (CAR). This makes JT-S-MT a good point of comparison for our Base model with limited parameters.

3.3 Results

Table.2 shows the comparison between our model and the baselines. Our Base model achieved an average BLEU score of 33.1 with limited parameters, demonstrating competitiveness compared to baselines, and outperforming JT-S-MT by 1.0 BLEU score. This demonstrates the effectiveness of our proposed PDE method. When further increasing the model parameters to 120M in the Large configuration, our model achieved an average BLEU score of 33.9, surpassing all baselines by a margin of 0.6 to 1.8 BLEU. These experiments illustrate that with an appropriate encoder modeling method, E2E ST can achieve outstanding

performance even without using pre-trained speech encoders. In addition, when facing cleaner datasets such as MuST-C v3.0 En-De, PDE consistently provides improvements relative to the baseline JT-S-MT, with a maximum increase of 2.1 BLEU scores.

4 Analysis

4.1 Ablation Studies

Is using an additional text encoder beneficial? To demonstrate the effectiveness of using an additional text encoder, we conduct experiments on the MuST-C v3.0 En-De dataset, comparing the differences with and without the text encoder, as shown in Table.3. The experimental models are modified based on the PDE-Base. In experiment c, we drop the text encoder, and the word embeddings are directly used as input to the translation encoder. The BLEU score drops from 28.4 to 28.0. This demonstrates the effectiveness of an additional text encoder.

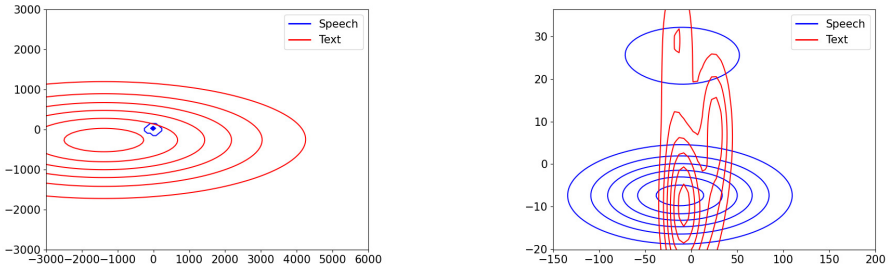
	Configuration	BLEU
a	PDE-Base	28.6
b	−mixup	28.4
c	−text encoder	28.0
d	−char CTC	28.1
e	−phoneme CTC	28.0
f	−phoneme CTC − char CTC	27.6

Table 3: BLEU scores on the MuST-C v3.0 En-De tst-COMMON dataset with different model configurations.

Is the multi-scale constraint on the speech encoder effective? To demonstrate the effectiveness of employing the multi-scale constraint on the speech encoder, we conduct experiments on the MuST-C v3.0 En-De dataset to investigate the impact of the (c, p) constraints. As shown in Experiments d, e, and f in Table.3, the absence of either one or both of the (c, p) constraints hurts the BLEU scores to varying degrees. With the score dropping by 27.6 when both constraints are missing.

4.2 Why is using an additional word encoder beneficial?

We extract representations from the speech and text encoders, and applied Principal Component Analysis (PCA) to reduce the high-dimensional features to two dimensions for visualization. In Figure.4a, the contour lines for speech (blue) and text (red) are more distinct and less overlapping compared to Figure.4b. The red contours representing text features cover a much larger area, while the blue contours representing speech features are concentrated in a smaller region.



(a) (b)

Fig. 4: (a) shows representations extracted from the outputs of the word encoders for both modalities, while (b) shows representations extracted from their inputs.

The centers of the contours for the two modalities are farther apart, indicating a significant difference between the features of two modalities.

In Figure.4b, the contour lines representing the two modalities show a significant overlap. The centers of the contours for both modalities are relatively close to each other, indicating that the features extracted from the outputs of two word encoders are similar. This proximity suggests that the representations of the two modalities have a high degree of alignment. The overlapping regions of the contour lines suggest that there is a common structure or pattern in the high-dimensional features of both modalities.

The analysis above indicates that an additional word encoder effectively alleviated the modality gap between the representations of the two modalities.

4.3 Case Study

Models		
CASE 1		
Ref.	src	And you know what I've learned?
	tgt	Und wissen Sie was ich gelernt habe?
JT-S-MT	tgt	Sie wissen, was ich gelernt habe.
PDE-Base	tgt	Wissen Sie, was ich gelernt habe?
CASE 2		
Ref.	src	Then the next steps, like language and so on, took less than a million years.
	tgt	Dann die nächstens Schritte, wie Sprache und so weiter, dauerten weniger als eine Million Jahre.
JT-S-MT	tgt	Im nächsten Schritt, wie Sprache und so weiter, dauerte es weniger als eine Million ~~Jahre~~.
PDE-Base	tgt	Die nächsten Schritte, wie Sprache und so weiter, dauerten weniger als eine Million Jahre.

Table 4: The cases are generated from JT-S-MT and PDE-Base on the MuST-C v3.0 En-De tst-COMMON dataset. The blue text represents incorrect translation, and the red text represents missing translation.

As shown in Table.4, for Case 1, the baseline model's translation fails to retain the interrogative form of the original sentence, turning it into a statement. However, our model correctly retains the question format and original intent.

For Case 2, the baseline model's translation lacks the plural form of "steps" and incorrectly uses the singular form with "es," which alters the sentence's meaning. Additionally, the semantic element of "years" is missing, further impacting the accuracy. In contrast, our model accurately maintains the correct plural form and meaning, ensuring the sentence is both semantically and syntactically correct.

These cases more intuitively demonstrate the effectiveness of our method in practical applications.

5 Conclusion

To address the modality gap in E2E ST, previous methods often employ cross-modal representation learning. In contrast, we explore an approach that does not require cross-modal learning by progressively aligning the speech encoder and text encoder. This method alleviates the modeling burden on the speech encoder, achieving comparable performance to existing modality alignment methods with fewer parameters and lower computational cost. This approach effectively decouples E2E ST from the reliance on a self-supervised pretrained speech encoder.

6 Acknowledgement

This work was supported in part by the National Science Foundation of China (No.62276056), the Natural Science Foundation of Liaoning Province of China (2022-KF-16-01), the Fundamental Research Funds for the Central Universities (Nos. N2216016 and N2316002), the Yunnan Fundamental Research Projects (No. 202401BC070021), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No.B16009).

References

1. Baevski, A., Zhou, Y., Mohamed, A., Auli, M.: wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in neural information processing systems **33**, 12449–12460 (2020)
2. Bérard, A., Besacier, L., Kocabiyikoglu, A.C., Pietquin, O.: End-to-end automatic speech translation of audiobooks. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)
3. Di Gangi, M.A., Cattoni, R., Bentivogli, L., Negri, M., Turchi, M.: MuST-C: a Multilingual Speech Translation Corpus. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 2012–2017. Minneapolis, Minnesota (Jun 2019)

4. Fang, Q., Ye, R., Li, L., Feng, Y., Wang, M.: STEMM: Self-learning with speech-text manifold mixup for speech translation. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 7050–7062. Dublin, Ireland (May 2022)
5. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd international conference on Machine learning. pp. 369–376 (2006)
6. Hsu, W.N., Bolte, B., Tsai, Y.H.H., Lakhotia, K., Salakhutdinov, R., Mohamed, A.: Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Transactions on Audio, Speech, and Language Processing **29**, 3451–3460 (2021)
7. Kudo, T., Richardson, J.: SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 66–71. Brussels, Belgium (Nov 2018)
8. Lee, Y., Kim, T.: Learning pronunciation from a foreign language in speech synthesis networks. CoRR **abs/1811.09364** (2018)
9. Lison, P., Tiedemann, J.: Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles (2016)
10. Meng, L., Xu, J., Tan, X., Wang, J., Qin, T., Xu, B.: Mixspeech: Data augmentation for low-resource automatic speech recognition. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 7008–7012. IEEE (2021)
11. Michael, M., Michaela, S., Sarah, M., Michael, W., Morgan, S.: Trainable text-speech alignment using kaldi. In: Interspeech. pp. 498–502 (2017)
12. Ouyang, S., Ye, R., Li, L.: WACO: Word-aligned contrastive learning for speech translation. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 3891–3907. Toronto, Canada (Jul 2023)
13. Park, D.S., Chan, W., Zhang, Y., Chiu, C.C., Zoph, B., Cubuk, E.D., Le, Q.V.: Specaugment: A simple data augmentation method for automatic speech recognition. Interspeech 2019 (Sep 2019)
14. Post, M.: A call for clarity in reporting BLEU scores. In: Proceedings of the Third Conference on Machine Translation: Research Papers. pp. 186–191. Brussels, Belgium (Oct 2018)
15. Rabiner, L.: Fundamentals of speech recognition. Prentice Hall google schola **2**, 447–453 (1993)
16. Tang, Y., Pino, J., Li, X., Wang, C., Genzel, D.: Improving speech translation by understanding and learning from the auxiliary text translation task. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 4252–4261. Online (Aug 2021)
17. Tang, Y., Pino, J., Wang, C., Ma, X., Genzel, D.: A general multi-task learning framework to leverage text data for speech to text tasks. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6209–6213 (2021)
18. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

19. Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states. In: International conference on machine learning. pp. 6438–6447. PMLR (2019)
20. Wang, C., Tang, Y., Ma, X., Wu, A., Okhonko, D., Pino, J.: Fairseq S2T: Fast speech-to-text modeling with fairseq. In: Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: System Demonstrations. pp. 33–39. Suzhou, China (Dec 2020)
21. Weiss, R.J., Chorowski, J., Jaitly, N., Wu, Y., Chen, Z.: Sequence-to-sequence models can directly translate foreign speech. arXiv preprint arXiv:1703.08581 (2017)
22. Xu, C., Hu, B., Li, Y., Zhang, Y., Huang, S., Ju, Q., Xiao, T., Zhu, J.: Stacked acoustic-and-textual encoding: Integrating the pre-trained models into speech translation encoders. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 2619–2630. Online (Aug 2021)
23. Xu, C., Zhang, Y., Jiao, C., Liu, X., Hu, C., Zeng, X., Xiao, T., Ma, A., Wang, H., Zhu, J.: Bridging the granularity gap for acoustic modeling. In: Findings of the Association for Computational Linguistics: ACL 2023. pp. 10816–10833. Toronto, Canada (Jul 2023)
24. Ye, R., Wang, M., Li, L.: End-to-end speech translation via cross-modal progressive training. In: Proc. of INTERSPEECH (Aug 2021)
25. Ye, R., Wang, M., Li, L.: Cross-modal contrastive learning for speech translation. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5099–5113. Seattle, United States (Jul 2022)
26. Yin, W., Liu, Z., Zhao, C., Wang, T., Tong, J., Ye, R.: Improving speech translation by fusing speech and text. In: Findings of the Association for Computational Linguistics: EMNLP 2023. pp. 6262–6273. Singapore (Dec 2023)