A Simple Concatenation can Effectively Improve Speech Translation

Linlin Zhang and **Kai Fan**[*] and **Boxing Chen** and **Luo Si**
Alibaba Group
{zll240651, k.fan, boxing.cbx, luo.si}@alibaba-inc.com

Abstract

A triple speech translation data comprises speech, transcription, and translation. In the end-to-end paradigm, text machine translation (MT) usually plays the role of a teacher model for the speech translation (ST) via knowledge distillation. Parameter sharing with the teacher is often adopted to construct the ST model architecture, however, the two modalities are independently fed and trained via different losses. This situation does not match ST's properties across two modalities and also limits the upper bound of the performance. Inspired by the works of video Transformer, we propose a simple unified cross-modal ST method, which concatenates speech and text as the input, and builds a teacher that can utilize both cross-modal information simultaneously. Experimental results show that in our unified ST framework, models can effectively utilize the auxiliary information from speech and text, and achieve compelling results on MuST-C datasets.

1 Introduction

Speech translation (ST) is the task that automatically translates a source acoustic speech signal into a text sequence in a target language. With the advance of Transformer, recent works on end-to-end speech translation (E2E ST) can alleviate many problems usually occurred in the cascade system and achieve comparable performance (Bahar et al., 2021; Bentivogli et al., 2021; Fang et al., 2022).

For the E2E ST model, MT is often used as the teacher of ST, and methods such as knowledge distillation or contrastive learning are used to bridge the modality gap. The MT teacher only uses the source text (transcription) information. The speech and text modalities are consumed individually by ST model. There are two main drawbacks. One is the teacher MT model can not use speech information, which limits the overall model performance. The other is MT uses text input, ST uses the speech input, then close the two individual modalities. There is no unified module can simultaneously use cross-modal information.

Here, we take a further step towards more effective use of both speech and transcription text in ST. Inspired by the related works of video Transformer (Kim et al., 2021), when processing video, concatenating video information and text embedding information can better model the cross-modal information of the video. We concatenate the preprocessed speech and the transcription text jointly, and encode the two-modal information simultaneously. Following the recent popular advance in E2E ST with knowledge distillation (KD) (Tang et al., 2021; Zhao et al., 2021), it provides a practical paradigm for transferring knowledge from rich-resource MT task to limited resource ST task. However, we re-define the role of teacher in our framework, because the information of the two modalities can further improve the upper bound of model performance than the single modality. Our proposed model, a **u**nified **c**ross-modal **c**oncatenate ST structure (**uccST**) introduces the teacher-student learning with Kullback-Leibler divergence (KL) regularization to transfer knowledge from cross-modal translation model to two subtasks – ST and MT.

Our main contributions can be summarized.

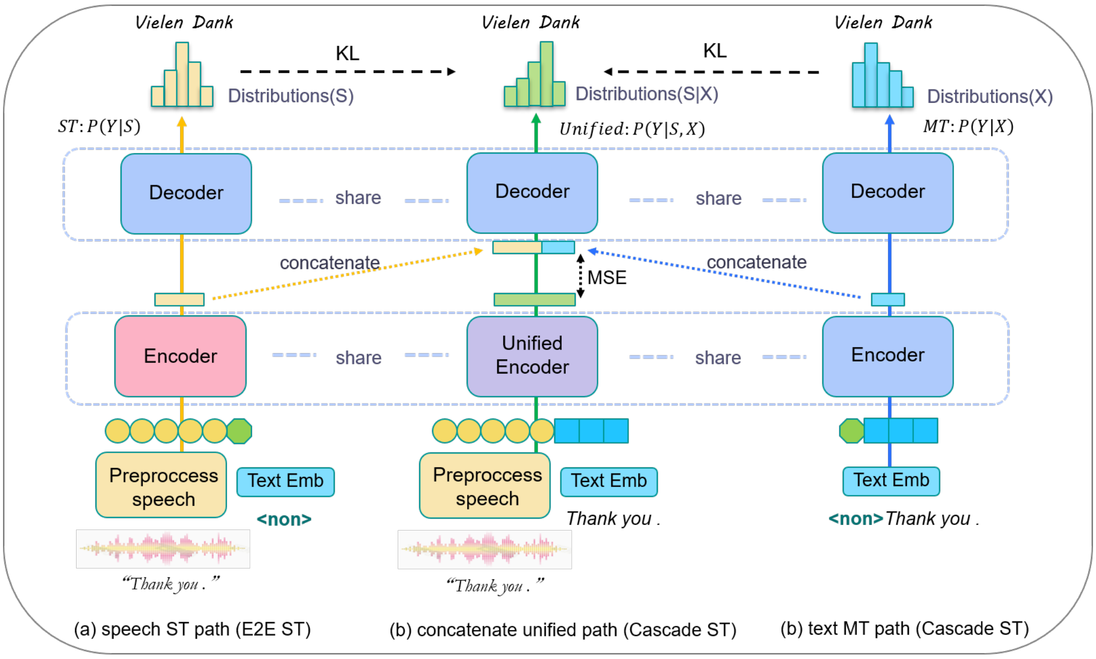
(1) Compared with the previous ST frameworks which can only utilize one single modality text in MT teacher, we design a unified framework that can use both input information of the two modalities simultaneously by concatenating speech and text.

(2) Our cross-modal framework has three diverse inputs when inference, containing three end-to-end and cascade decoding paths. Our multi-task learning framework allows sub-tasks to collaborate, showing promising performance on both end-to-end and cascade ST.

(3) We conduct various experiments on the MuST-C corpus. When using the limited ternary

[*] Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1793–1802
July 9-14, 2023 ©2023 Association for Computational Linguistics



(a) speech ST path (E2E ST) (b) concatenate unified path (Cascade ST) (b) text MT path (Cascade ST)

Figure 1: The overview of the unified cross-modal concatenate framework. The left path (a) is the traditional direct speech translation, input speech, output target text. The middle path (b) is to concatenate the preprocessed speech sequence and the source text sequence. The right part (c) is an MT model that translates the source text (or transcription) into the target text. <non> represents filling with <non> placeholder.

ST data, our E2E ST model can achieve state-of-the-art performance. When adding the external data, our method significantly improves over the strong baselines.

2 Unified Cross-modal Concatenate ST

2.1 Background

Given the source acoustic speech sequence $\mathbf{s}$, the corresponding transcription $\mathbf{x}$ and the text sequence $\mathbf{y}$ in target language, speech translation usually model the conditional distribution as follows.

$$p(\mathbf{y}|\mathbf{s}) = \sum_{\mathbf{x}} p(\mathbf{y}|\mathbf{x},\mathbf{s})p(\mathbf{x}|\mathbf{s}) \qquad (1)$$

In most works, the assumption $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|\mathbf{x},\mathbf{s})$ is usually adopted as the source transcription can deterministicially infer the final translation. However, we prefer to leverage the original conditional probability for our modeling.

2.2 Cross-modal Concatenate Framework

Inspired by video Transformer, the unified model can take as input the concatenation of the features of two modalities along the temporal dimension. As shown in Figure 1(b), the speech preprocessing module usually includes CNN down-sampling and a speech encoder, such as the encoder of the pre-trained ASR or the pre-trained audio encoder wav2vec2.0. For the text sequence, we simply process each token with an embedding layer. After the concatenation, we add the position embedding and segment embedding in the fashion of BERT.

2.2.1 Multi-task Training

Concretely, given a ternary ST example $(\mathbf{s},\mathbf{x},\mathbf{y})$. We optimize three translation tasks in parallel, including MT, ST and our introduced unified cross-modal translation.

$$\mathcal{L}_{MT} = \log p(\mathbf{y}|\mathbf{x}) + \log p(\mathbf{y}|\mathbf{s}) + \log p(\mathbf{y}|[\mathbf{x},\mathbf{s}]) \qquad (2)$$

where $[\cdot,\cdot]$ indicates the concatenation operation.

2.2.2 Regularization

Unlike other ST frameworks, the unified cross-modal decoder output provides the teacher signal, and the ST and MT models are two students. We employ Kullback–Leibler divergence (KL) to minimize the decoding distribution between the student and the teacher model.

$$\mathcal{L}_{KL} = \mathrm{KL}\left(p_{st} \| p_{\text{unified}}\right) + \mathrm{KL}\left(p_{mt} \| p_{\text{unified}}\right) \qquad (3)$$

Further, we impose a representation regularization on the encoder output. Particularly, we apply the MSE loss.

$$\mathcal{L}_{MSE} = \mathrm{MSE}\left(\left[Z_{ST}, Z_{MT}\right], Z_{\text{Unified}}\right) \qquad (4)$$

Model	En-DE			En-Fr			En-Es			Paras
	S	$S\mid X$	X	S	$S\mid X$	X	S	$S\mid X$	X	
E2E baseline	24.5	-	-	34.9	-	-	28.2	-	-	76M
Cascade	-	-	25.4	-	–	35.7	-	-	28.9	-
Dual-Decoder (Le et al., 2020)	23.6	-	-	33.5	-	-	28.1	-	-	-
Adapter Tuning (Le et al., 2021)	24.6	-	-	34.7	-	-	28.7	-	-	78M
Multi-Decoder (Dalmia et al., 2021)	-	26.3	-	-	37.0	-	-	-	-	-
Bi KD (Inaguma et al., 2021)	25.3	-	-	-	-	-	-	-	-	-
mutual KL (Zhao et al., 2021)	-	-	-	36.3	-	-	28.7	-	-	76M
No Uni baseline	24.8	-	25.4	36.4	-	36.8	28.5	-	28.9	76M
Our uccST	25.5†	26.3	25.7	36.6†	37.6	36.9	28.9†	29.7	29.2	76M

Table 1: BLEU scores of the speech translation results on the tst_COMMON sets. The models are trained with the ternary ST data on constrained settings. †: the SOTA performance of all E2E methods. S indicates the ST decoding path. $S|X$ indicates the unified decoding path with both speech and ASR transcribed text. X indicates the MT decoding path with ASR transcribed text. No Uni baseline refers to 4.3.

Model	En-De	En-Fr	Paras
JT-ST* (Tang et al., 2021)	26.8	37.4	74M
E2E-ST-TDA* (Du et al., 2022)	27.1	37.4	76M
Chimera (Han et al., 2021)	26.3	35.6	165M
XSTNet (Ye et al., 2021)	27.8	38.0	155M
SATE (Xu et al., 2021)	28.1	-	-
STEMM (Fang et al., 2022)	28.7	37.4	155M
ConST (Ye et al., 2022)	28.3	38.3	155M
W2V2 baseline	27.3	36.8	155M
Our W2V2-uccST	28.8†	39.1†	158M

Table 2: The ST BLEU results on the tst_COMMON dataset when using the external MT data. * indicates they did not use the pre-trained wav2vec2.0.

Model	En-De			En-Fr		
	S	$S\mid X$	X	S	$S\mid X$	X
E2E	24.53	-	-	34.88	-	-
No Uni	24.83	-	25.36	36.36	-	36.77
Uni sim	25.17	25.74	25.53	36.39	37.12	36.86
Ours	25.54	26.32	25.65	36.61	37.64	36.94

Table 3: Ablation analysis of concatenation in the constrained setting. Uni sim: Unified simple.

where we concatenate the encoder outputs of ST and MT such that it results in the same length as the unified model.

2.2.3 Training and Inference

In summary, the final loss of the proposed uccST can be written as follows.

$$\mathcal{L} = \mathcal{L}_{MT} + \lambda \mathcal{L}_{KL} + \eta \mathcal{L}_{MSE} \tag{5}$$

where λ and θ are hyper-parameters. During inference, we have 3 optional decoding paths. If only audio is available, we can actually choose any decoding path. For the cross-modal unified or MT decoding path, it requires the transcription from an additional ASR, which is commonly a pre-training step for ST.

3 Experiments Settings

3.1 Datasets and Settings

Data For a fair comparison with previous works, we conduct our experiments on the widely used MuST-C V1: English-German (En–De), English-French (En–Fr) and English-Spanish (En–Es) corpus (Gangi et al., 2019).

On En-De and En-Fr, we also verify to what extent the auxiliary MT data can improve our multi-task training. Specifically, we extract about 20M sentence pairs for the WMT14 En-Fr, 4.5M for WMT14 En-De, and 18M for Opensubtitle2018 En-De.

Settings We implement all our experiments on Fairseq[1]. We experiment with two architectures[2]. One is the transformer model with 512 hidden units 2048 feed-forward size, which is same as Tang et al. (2021), in purpose for constrained ST data. The other one is to leverage pre-trained wav2vec2.0(Baevski et al., 2020) as the speech preprocessing module. Since wav2vec2.0 has been already pre-trained with the audio data of Librispeech(Panayotov et al., 2015), we only compare this setup to other works with same architecture. During training, the text input is ground truth transcript of MuST-C. Note that the transcription data in Librispeech is not used in our case. We select the alternative batches between ST and MT with sampling ratios 1.0 and 0.25, respectively.

[1] https://github.com/pytorch/fairseq

[2] https://github.com/pytorch/fairseq/tree/main/examples/speech_to_text

4 Experiments Results

4.1 Results on the Constrained ST Data

As shown in Table 1, our method achieves an appealing performance on the three language pairs in the restricted ternary MuST-C data.

Compared with the direct E2E ST baseline, our method has enhanced 0.7 to 1.7 BLEU on the three language directions, with an average gain of 1.13 BLEU. In a word, our approach can achieve the SOTA translation performance among all end-to-end ST methods.

Compared with the cascade method that we have reproduced, our E2E ST decoding path surpasses the cascade on the language pairs En-Fr, and reaches a comparable level on En-De and En-Es. The results of the MT decoding path with the transcription exceed the cascade method on all language pairs. Our cross-modal unified decoding method has enhanced 0.8 to 1.9 BLEU than the cascade method, with an average gain of 1.17 BLEU. In summary, our E2E ST method has matched or surpassed the cascade method on the constrained triple ST data, and our cross-modal unified decoding method has exceeded the traditional cascade baseline.

4.2 Results on the External Data

Since our model is a multitask learning method that includes the MT subtask, we add additional MT data for comparison experiments. As shown in Table 2, we compare different baselines with similar data usage. Our E2E method (*i.e.*, ST decoding path) and the corresponding baselines are presented in the bottom two rows. The first two rows in the table are the baselines without wav2vec2.0, and the middle part of the table represents the methods with wav2vec2.0 architecture. It is concluded that the pre-trained audio encoder model is indeed helpful for downstream ST task. By introducing more auxiliary MT data, our model with pre-trained wav2vec2.0 improves 1.5 and 2.3 BLEU on the two language pairs En-De and En-Fr, respectively. In shot, our approach outperforms existing state-of-the-art models, especially on En-Fr.

4.3 Ablation Analysis of Concatenation

In order to analyze whether our concatenation is effective, we have done comparative experiments on different input models. As shown in Table 3, E2E baseline indicates Figure 1(a). No Unified baseline means to removing the (b) in Figure 1, and the KL loss is calculated between ST and MT. Unified simple model only concatenates the speech and text sequence from each corresponding encoder output. In accordance to the result, no concatenation or the concatenation method in Unified simple model is inferior to our proposal.

tst_COMMON	ST(BLEU)
Our uccST	25.54
w/o KL	25.12
w/o MSE	24.93
w/o multi-task	24.53

Table 4: Ablation study on the En-De tst_COMMON set in the constrained setting.

4.4 Ablation Study on Loss

To analyze the importance of each component of the overall uccST loss, we conduct an ablation study by removing each loss step by step. Table 4 summarizes the results of the ablation study. We first remove the KL loss but reserve the unified structure. It concludes that the KL terms contribute to an improvement of 0.42 BLEU score. After further removing the MSE loss, the model becomes a standard multi-task ST Transformer. When removing multi-task, it reduces to a standard E2E ST model.

4.5 Comparison with the Cascaded Model

As shown in Table 5, our proposed E2E ST has reached a comparable level to cascaded methods, both in data-constrained and non-constrained cases. As to the two decoding methods that require transcription text, our method can outperform the cascade baseline. Meanwhile, we can observe that with the additional external data, the gap between two inference setups $S|X$ and S is narrowed.

5 Related Works

Cascade ST. Cascade ST system concatenates the individual ASR and MT components (Stentiford and Steer, 1988; Waibel et al., 1991), and represents an intuitive solution to achieve reasonable performance and high intelligibility. At the same time, this cascade method also faces some thorny problems: the traditional cascade method suffers from error propagation and the loss of acoustic information that might be useful to improve final translations. To alleviate the aforementioned problems, some tight integration methods have been proposed (Sperber et al., 2019; Bahar et al., 2020).

Model	En-De					En-Fr				
	ASR	MT	S	$S\|X$	X	ASR	MT	S	$S\|X$	X
Cascade	12.11	29.87	-	-	25.44	11.09	43.21	-	-	35.72
Ours	-	-	25.54	26.32	25.65	-	-	36.61	37.64	36.94
Cascade(ext)	9.85	33.66	-	-	28.97	9.76	46.13			39.16
Ours(ext)	-	-	28.82	29.03	28.95	-	-	39.11	39.32	39.26

Table 5: BLEU scores on the tst_COMMON dataset with cascade method and ours. (ext) is with pre-trained wav2vec2.0 on external MT data. ASR task scores are reported as WER, and others are BLEU.

End-to-end ST. To overcome the weakness of cascade models, Berard et al. (2016) proposed the first direct neural network model of an encoder decoder architecture without the intermediate transcription. Currently, more effective solutions are used in end-to-end ST models (Park et al., 2019; Dong et al., 2021). To alleviate the cross-modal difficulty in end-to-end models, two-pass (Kano et al., 2017; Anastasopoulos and Chiang, 2018) methods are proposed. Curriculum learning (Kano et al., 2017; Wang et al., 2020) is proposed to improve performance of ST models.

6 Conclusion

In this paper, we designed a unified ST framework. Compared with the previous ST frameworks which can only utilize one single modality text in MT teacher, our method can use both information of the two modalities simultaneously by concatenating speech and text. Our ST method can better utilize the cross-modal information. Experiments show that our method can significantly improve ST performance regardless of using the limited ternary data or adding auxiliary external data.

Limitations

A lot of recent work especially in computer vision has leveraged the unsupervised methods or unpaired multi-modality data to pre-trained cross-modal language model. Applying the same idea into speech language model is also discussed in some recent research works. To compare fairly with previous works in ST area, we do not build our model on top of such frameworks and discuss how to utilize the raw audio. In terms of the model training, multi-tasks may affect each other due to uneven data distribution, and we have just scratched the surface of this part of the analysis.

Ethics Statement

This work designs a unified cross-modal concatenate ST structure to take better advantage of the two modalities of speech and text. The datasets and pre-trained models we use are publicly available and are widely used in the research community, whether in a constrained or unconstrained situation.

References

Antonios Anastasopoulos and David Chiang. 2018. Tied multitask learning for neural speech translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 82–91. Association for Computational Linguistics.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Parnia Bahar, Tobias Bieschke, Ralf Schlüter, and Hermann Ney. 2021. Tight integrated end-to-end training for cascaded speech translation. In *IEEE Spoken Language Technology Workshop, SLT 2021, Shenzhen, China, January 19-22, 2021*, pages 950–957. IEEE.

Parnia Bahar, Patrick Wilken, Tamer Alkhouli, Andreas Guta, Pavel Golik, Evgeny Matusov, and Christian Herold. 2020. Start-before-end and end-to-end: Neural speech translation by apptek and RWTH aachen university. In *Proceedings of the 17th International Conference on Spoken Language Translation, IWSLT 2020, Online, July 9 - 10, 2020*, pages 44–54. Association for Computational Linguistics.

Luisa Bentivogli, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus direct speech translation: Do the differences still make a difference? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and* *the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2873–2887. Association for Computational Linguistics.

Alexandre Berard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *CoRR*, abs/1612.01744.

Siddharth Dalmia, Brian Yan, Vikas Raunak, Florian Metze, and Shinji Watanabe. 2021. Searchable hidden intermediates for end-to-end models of decomposable sequence tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 1882–1896. Association for Computational Linguistics.

Qianqian Dong, Rong Ye, Mingxuan Wang, Hao Zhou, Shuang Xu, Bo Xu, and Lei Li. 2021. Listen, understand and translate: Triple supervision decouples end-to-end speech-to-text translation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12749–12759. AAAI Press.

Yichao Du, Zhirui Zhang, Weizhi Wang, Boxing Chen, Jun Xie, and Tong Xu. 2022. Regularizing end-to-end speech translation with triangular decomposition agreement. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 10590–10598. AAAI Press.

Qingkai Fang, Rong Ye, Lei Li, Yang Feng, and Mingxuan Wang. 2022. 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), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 7050–7062. Association for Computational Linguistics.

Mattia Antonino Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2012–2017. Association for Computational Linguistics.

Chi Han, Mingxuan Wang, Heng Ji, and Lei Li. 2021. Learning shared semantic space for speech-to-text translation. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 2214–2225. Association for Computational Linguistics.

Hirofumi Inaguma, Tatsuya Kawahara, and Shinji Watanabe. 2021. Source and target bidirectional knowledge distillation for end-to-end speech translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 1872–1881. Association for Computational Linguistics.

Takatomo Kano, Sakriani Sakti, and Satoshi Nakamura. 2017. Structured-based curriculum learning for end-to-end english-japanese speech translation. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pages 2630–2634. ISCA.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 5583–5594. PMLR.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Hang Le, Juan Miguel Pino, Changhan Wang, Jiatao Gu, Didier Schwab, and Laurent Besacier. 2020. Dual-decoder transformer for joint automatic speech recognition and multilingual speech translation. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3520–3533. International Committee on Computational Linguistics.

Hang Le, Juan Miguel Pino, Changhan Wang, Jiatao Gu, Didier Schwab, and Laurent Besacier. 2021. Lightweight adapter tuning for multilingual speech translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 817–824. Association for Computational Linguistics.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 2613–2617. ISCA.

Matthias Sperber, Graham Neubig, Ngoc-Quan Pham, and Alex Waibel. 2019. Self-attentional models for lattice inputs. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1185–1197. Association for Computational Linguistics.

Fred WM Stentiford and Martin G Steer. 1988. Machine translation of speech. *British Telecom technology journal*, 6(2):116–122.

Yun Tang, Juan Miguel Pino, Xian Li, Changhan Wang, and Dmitriy Genzel. 2021. 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4252–4261. Association for Computational Linguistics.

Alex Waibel, Ajay N. Jain, Arthur E. McNair, Hiroaki Saito, Alexander G. Hauptmann, and Joe Tebelskis. 1991. JANUS: a speech-to-speech translation system using connectionist and symbolic processing strategies. In *1991 International Conference on Acoustics, Speech, and Signal Processing, ICASSP '91, Toronto, Ontario, Canada, May 14-17, 1991*, pages 793–796. IEEE Computer Society.

Chengyi Wang, Yu Wu, Shujie Liu, Ming Zhou, and Zhenglu Yang. 2020. Curriculum pre-training for end-to-end speech translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 3728–3738. Association for Computational Linguistics.

Chen Xu, Bojie Hu, Yanyang Li, Yuhao Zhang, Shen Huang, Qi Ju, Tong Xiao, and Jingbo Zhu. 2021. 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2619–2630. Association for Computational Linguistics.

Rong Ye, Mingxuan Wang, and Lei Li. 2021. End-to-end speech translation via cross-modal progressive training. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, pages 2267–2271. ISCA.

Rong Ye, Mingxuan Wang, and Lei Li. 2022. 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, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 5099–5113. Association for Computational Linguistics.

Biao Zhang, Ivan Titov, Barry Haddow, and Rico Sennrich. 2020. Adaptive feature selection for end-to-end speech translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2533–2544. Association for Computational Linguistics.

Jiawei Zhao, Wei Luo, Boxing Chen, and Andrew Gilman. 2021. Mutual-learning improves end-to-end speech translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3989–3994. Association for Computational Linguistics.

A Appendix

Experience Settings The data statistics are shown in Table 6.

corpus	ST(H/Sents)	MT(Sents)
En-De	408/234K	22.5M(WMT+OS)
En-Fr	492/280K	20M(WMT)
En-Es	504/270K	-

Table 6: The statistics for the three language pairs. H: Hours. Sents: Sentences. OS: OpenSubtiles2018. WMT: WMT14.

We implement all our experiments on Fairseq[3]. We experiment with two architectures[4]: a Transformer model with 512 hidden units 2048 feed-forward size. All ST and ASR models use the same encoder with 12 layers and 6 decoder layers. The corresponding MT model also has 6 encoder and decoder layers. We share parameters of all 6 text encoder Transformer layers with the top 6 Transformer layers in the speech encoder. Hence the preprocessed speech is composed of CNN layers and 6 Transformer layers. The model architecture

[3] This tool can be accessed via https://github.com/pytorch/fairseq

[4] https://github.com/pytorch/fairseq/tree/main/examples/speech_to_text

is same as (Tang et al., 2021), when on constrained ST data.

When using the pre-trained wav2vec2.0 (Baevski et al., 2020) as the preprocessed speech module, we add two additional 1- dimensional convolutional layers to further shrink the audio, with kernel size 5, stride size 2, padding 2, and hidden dimension 1024. Then stack our Unified concatenate model.

For all experiments on limited triple data, we used the Adam optimizer (Kingma and Ba, 2015) with the learning rate $2e-3$. The dropout rate and the label smoothing are both set as 0.1. We choose $\lambda_1 = 1.0$, $\lambda_2 = 1.0$ and $\eta = 0.3$ in the training loss equation through grid search ($[0.2, 1.5]$ for λ and $[0.1, 0.5]$ for η).

For adding external corpus experiments, we fine-tune on the triple data with multi-task learning loss. We select the alternative batches between ST and MT with sample ratios 1.0 and 0.25, respectively. We randomly select 1M WMT14 and 1M OpenSubtitle18 as our fine-tune MT data on En-De. We randomly select 2M WMT14 on En-Fr. For all models at inference, we average 10 checkpoints with a beam size 5.

Limited ST Baselines We compare our method with various baseline models on constrained ST situation:

- E2E ST baseline: The direct ST model translates the speech inputs to the target language text without transcription. The encoder of the E2E ST model is initialized by first training on the ASR data from the triple ST data.

- Cascade baseline: ASR and MT models are independently trained, and then the outputs of the ASR model are taken as the inputs to the MT model. The ASR model uses the same model settings as the corresponding ST model.

- AFS model: AFS model (Zhang et al., 2020) inserts a module between the ST encoder and a pre-trained ASR encoder to filter speech features for translation. AFS model is an end-to-end speech translation.

- Dual-decoder model: Dual-decoder Transformer is an end-to-end ST architecture that jointly performs ASR and ST (Le et al., 2020). The ASR and MT decoders use attention modules to exchange information with each other.

- Bi KD: Source and target bidirectional Knowledge Distillation (Inaguma et al., 2021).

- mutual KL: Bidirectional KL for ST and MT (Zhao et al., 2021).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
section Limitations

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
section abstract and introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

section Appendix

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
table 1 and table 2

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
section Appendix

□ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
No response.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
section Appendix

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.