End-to-end Speech Translation via Cross-modal Progressive Training

Rong Ye, Mingxuan Wang, Lei Li

ByteDance AI Lab, Shanghai, China

{yerong,wangmingxuan.89,lileilab}@bytedance.com

Abstract

End-to-end speech translation models have become a new trend in research due to their potential of reducing error propagation. However, these models still suffer from the challenge of data scarcity. How to effectively use unlabeled or other parallel corpora from machine translation is promising but still an open problem. In this paper, we propose **Cross S**peech-**T**ext **Net**work (**XSTNet**), an end-to-end model for speech-to-text translation. XSTNet takes both speech and text as input and outputs both transcription and translation text. The model benefits from its three key design aspects: a self-supervised pretrained sub-network as the audio encoder, a multi-task training objective to exploit additional parallel bilingual text, and a progressive training procedure. We evaluate the performance of XSTNet and baselines on the MuST-C En-X and LibriSpeech En-Fr datasets. In particular, XSTNet achieves state-of-the-art results on all language directions with an average BLEU of 28.8, outperforming the previous best method by 3.2 BLEU. Code, models, cases, and more detailed analysis are available at https://github.com/ReneeYe/XSTNet.

1. Introduction

Speech-to-text translation (ST) has found increasing applications. It takes speech audio signals as input and outputs text translations in the target language. Recent work on ST has focused on unified end-to-end neural models with the aim to supersede pipeline approaches combining automatic speech recognition (ASR) and machine translation (MT). However, training end-to-end ST models is challenging - there is limited parallel *speech-text* data. For example, there are only a few hundreds hours for English-German in MuST-C corpus. Existing approaches to this problem can be grouped into two categories: *a)* multi-task supervision with the *speech-transcript-translation* triple data [1, 2]; *b)* pretraining with external large-scale MT parallel text data [3, 4, 5, 6]. We notice that the triple data can decompose into three sub-tasks with parallel supervision, ST, ASR, and MT. This motivates us to design a multi-task model.

In this paper, we designed **Cross S**peech-**T**ext **Net**work (**XSTNet**) for end-to-end ST to joint train ST, ASR and MT tasks. XSTNet supports either audio or text input and shares a Transformer [7] module. To bridge the gap between the audio and text modality, we use a self-supervised trained Wav2vec2.0 representation of the audio [8], which provides a more compressed and contextual representation than the log-Mel filter bank feature. Furthermore, our method is able to incorporate external large-scale MT data. Finally, to support the training of XSTNet, we carefully devise the progressive multi-task learning strategy, a multi-stage procedure following the popular pre-training and fine-tuning paradigm.

Despite the model's simplicity, the experimental results on MuST-C and Augment LibriSpeech datasets improved by a big margin (+3.2 BLEU) against the previous SOTA method.

2. Related Work

End-to-end ST [9] gave the first proof of the potential for end-to-end ST without using the intermediate transcription. And recent Seq2Seq models have received impressive results [10, 11, 12, 13]. Techniques, such as pre-training [4, 5, 14, 3], multi-task learning [15, 2], self-training [16] and meta-learning [1] have further improved the performance of end-to-end ST models. Very recently, [6] proposed a Fused Acoustic and Text Masked Language Model to pre-train and improved ST by fine-tuning.

Self-supervised Pretraining This work is partially motivated by the recent success of self-supervised contrastive learning for speech [17, 18, 8]. These representations have been shown to be effective in low-resource ASR [8], ST [19, 20] and multilingual ST [21]. Recently, many audio-related tasks have demonstrated the feasibility and outstanding performance by using the wav2vec2.0 representation [22, 23]. Maybe the most related work is [21]. However, our work focuses more on the training strategy, while theirs focused on incorporating two pre-trained modules to strengthen the multi-lingual ST.

3. Proposed Method: XSTNet

3.1. Problem Formulation

The speech translation corpus contains *speech-transcript-translation* triples $\mathcal{D} = \{(\mathbf{s}, \mathbf{x}, \mathbf{y})\}$, where $\mathbf{s} = (s_1, ..., s_{|\mathbf{s}|})$ is the input sequence of the audio wave (or the acoustic features), $\mathbf{x} = (x_1, ..., x_{|\mathbf{x}|})$ is the transcript from the source language, and $\mathbf{y} = (y_1, ..., y_{|\mathbf{y}|})$ represents the corresponding translation in the target language. Despite the fact that transcripts are provided during training, the end-to-end speech translation models directly produce the translation *vecy* without producing the transcript *vecx* as an intermediate output. As a result, understanding how to leverage ancillary transcript supervision and make the most of the triple-supervised dataset is critical. The triple-supervised dataset can be pairwise combined into three parallel-supervised sub-datasets, $\mathcal{D}_{\text{ASR}} = \{(\mathbf{s}, \mathbf{x})\}$, $\mathcal{D}_{\text{ST}} = \{(\mathbf{s}, \mathbf{y})\}$ and $\mathcal{D}_{\text{MT}} = \{(\mathbf{x}, \mathbf{y})\}$, which solve ASR, end-to-end ST and MT respectively.

We also introduce external MT dataset $\mathcal{D}_{\text{MT-ext}} = \{(\mathbf{x}', \mathbf{y}')\}$. The amount of external MT corpus is much larger than the ST corpus, i.e. $|\mathcal{D}_{\text{MT-ext}}| \gg |\mathcal{D}|$.

3.2. Speech Encoder

The first part of XSTNet is Wav2vec2.0 sub-network to process speech data in waveform. **Wav2vec2.0** [8] is a model to learn the contextualized speech representation from unlabelled audio data. It consists of a multi-layer convolutional feature encoder and a Transformer-based context encoder. The multi-layer convolutional encoder takes the raw audio signal as input and outputs the latent speech representation, which is then used by the Transformer encoder to output the contextual representation. The self-supervised pre-trained contextual audio
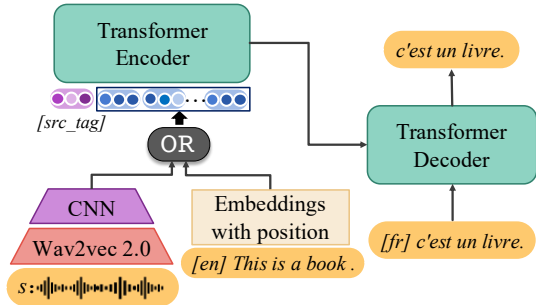
Figure 1: *The model structure of XSTNet. It accepts either audio or text input. For the audio input, we deploy wav2vec2.0 followed by two convolution layers to get the audio embedding. The Transformer encoder and decoder are shared for both modalities.*

representation $\boldsymbol{c} = [c_1, ...c_T]$ offers a good initialization for the model's audio presentation. In the experiment, we use the raw 16-bit 16kHz mono-channel audio as the audio input and follow a base configuration of wav2vec2.0, trained on audio data from Librispeech [24] but without fine-tuning[1].

Convolution layers The sequence of Wav2vec2.0 output for a voice utterance typically has a length of a few hundred, larger than that of a sentence. To further match the lengths of the audio representation and text sequences, we add two additional layers of 2-stride 1-dimensional convolutional layers with GELU activation after Wav2vec2.0, reducing the time dimension by a factor of 4. Hence, we have $\boldsymbol{e}_s = \mathrm{CNN}(\boldsymbol{c})$, $\boldsymbol{e}_s \in \mathbb{R}^{d \times T/4}$, where d is the same as the hidden size of Transformer. We set the kernel size of CNN 5, and the hidden size $d = 512$.

3.3. Encoder-Decoder with Modality and Language Indicators

Since XSTNet supports both speech and text as input, we add an additional lookup table for text tokens to map the embeddings. We adopt the standard Transformer-base model [7] to accomplish the specific generation tasks, ST, ASR, and MT. We use pre-layer normalization for stable training.

We use different indicators `[src_tag]` to distinguish the three tasks and audio/text inputs. The input embedding $\boldsymbol{e}$ (either audio or text) is fed into the Transformer encoder. Specifically,

- For the audio input, we add extra `[audio]` token with embedding $\boldsymbol{e}_{[audio]} \in \mathbb{R}^d$, and the embedding of the audio $\boldsymbol{e} \in \mathbb{R}^{d \times (T/4+1)}$ is the concatenation of $\boldsymbol{e}_{[audio]}$ and $\boldsymbol{e}_s$ in terms of the sequence length.

- For the text input, we put the language id symbol before the sentence. For example, the embedding of English sentence "This is a book." is the embedding of "[en] This is a book.".

When decoding, the language id symbol serves as the initial token to predict the output text. For example, if the audio input for sentence "This is a book." is in English, to do ASR, we use [en] as the BOS and decode "[en] This is a book.", while to translate into French, we use [fr] as the BOS and decode "[fr] c'est un livre."

3.4. Progressive Multi-task Training

The progressive multi-task training strategy consists of two stages, large-scale MT pre-training with external parallel text and multi-task fine-tuning.

---

[1] https://dl.fbaipublicfiles.com/fairseq/wav2vec/wav2vec_small.pt

---

Algorithm 1 Progressive Multi-task Training for XSTNet

1: **Input**: Tasks T={ST, ASR, MT, MT-ext}
2: Initialize the speech module parameters in θ using wav2vec2.0-base, and the rest at random.
3: MT pre-training: optimize Eq (1) on $\mathcal{D}_{\text{MT-ext}}$.
4: **while** not converged **do**
5:     **Step 1**: random select a task τ from T.
6:     **Step 2**: sample a batch of $(\mathbf{x}, \mathbf{y})$ from $\mathcal{D}_\tau$, and optimize the cross entropy loss defined in Eq (2).
7: **end while**

Large-scale MT Pre-training We first pre-train the transformer encoder-decoder module using external MT data $\mathcal{D}'_{\text{MT}}$.

$$\mathcal{L}(\theta) = -\mathbb{E}_{\mathbf{x},\mathbf{y} \in \mathcal{D}_{\text{MT-ext}}} \log P(\mathbf{y}|\mathbf{x}; \theta) \qquad (1)$$

, where θ is the model parameters. Experimental results show that the MT pre-training provide a good warm-up for the shared transformer module.

Multi-task Fine-tuning During the fine-tuning, we combine external MT, ST, ASR, and MT parallel data from the in-domain speech translation dataset and jointly optimize the negative log-likelihood loss.

$$\mathcal{L}(\theta) = -\mathbb{E}_{\mathbf{x},\mathbf{y} \in \mathcal{D} \cup \mathcal{D}_{\text{MT-ext}}} \log P(\mathbf{y}|\mathbf{x}; \theta) \qquad (2)$$

,where $\mathcal{D} = \mathcal{D}_{\text{ST}} \cup \mathcal{D}_{\text{ASR}} \cup \mathcal{D}_{\text{MT}}$ is the union set of all the parallel subsets (the same notation hereinafter).

The overall training process is shown in Algorithm 1. It is ***progressive***, because the external MT data, used in the pre-training stage, is continuously used in the fine-tuning stage. In Section 5.1, we will show that the training procedure largely influences the translation performance, and the progressive multi-task training process is the most effective. In the experiment, we use Adam optimizer [25] with learning rate = 2×10^{-4} and warm-up 25k steps.

4. Experiments

4.1. Datasets

ST datasets We conduct experiments on **MuST-C** [26] and **Augmented LibriSpeech** En-Fr (LibriTrans) [27] datasets. MuST-C contains Engish speech to 8 languages: German (De), Spanish (Es), French (Fr), Italian (It), Dutch (Nl), Portuguese (Pt), Romanian (Ro), and Russian (Ru). We evaluate the BLEU scores on tst-COMMON of MuST-C and the test set of Libri-Trans.

MT datasets We use external WMT machine translation datasets[2] [28] for En-De/Es/Fr/Ro/Ru directions, and OPUS100[3] [29] for En-It/Nl/Pt directions. We also introduce OpenSubtitle[4] [30] for En-De.

4.2. Experimental Setups

We jointly tokenize the bilingual text (En and X) using sub-word units with a vocabulary size of 10k, learned from SentencePiece[5] [34]. The model configurations have been stated in Section 3.2 and 3.3. We save the checkpoint with the best BLEU

---

[2] https://www.statmt.org/wmt16/translation-task.html
[3] http://opus.nlpl.eu/opus-100.php
[4] https://opus.nlpl.eu/OpenSubtitles-v2018.php
[5] https://github.com/google/sentencepiece

Models	External Data	Pre-train Tasks	De	Es	Fr	It	Nl	Pt	Ro	Ru	Avg.
Transformer ST [13]	×	ASR	22.8	27.4	33.3	22.9	27.2	28.7	22.2	15.1	24.9
AFS [31]	×	×	22.4	26.9	31.6	23.0	24.9	26.3	21.0	14.7	23.9
Dual-Decoder Transf. [15]	×	×	23.6	28.1	33.5	24.2	27.6	30.0	22.9	15.2	25.6
Tang et al. [2]	MT	ASR, MT	23.9	28.6	33.1	-	-	-	-	-	-
FAT-ST (Big) [6]	ASR, MT, mono-data[†]	FAT-MLM	25.5	30.8	-	-	30.1	-	-	-	-
W-Transf.	audio-only*	SSL*	23.6	28.4	34.6	24.0	29.0	29.6	22.4	14.4	25.7
XSTNet (Base)	audio-only*	SSL*	25.5	29.6	36.0	25.5	30.0	31.3	25.1	16.9	27.5
XSTNet (Expand)	MT, audio-only*	SSL*, MT	27.8[§]	30.8	38.0	26.4	31.2	32.4	25.7	18.5	28.8

Table 1: *Performance (case-sensitive detokenized BLEU) on MuST-C test sets.* [†]*: "Mono-data" means audio-only data from Librispeech, Libri-Light, and text-only data from Europarl/Wiki Text; *: "Audio-only" data from LibriSpeech is used in the pre-training of wav2vec2.0-base module, and "SSL" means the self-supervised learning from unlabeled audio data.* [§] *uses OpenSubtitles as external MT data.*

Models	Uncased-tok	Cased-detok
Transformer ST [13]	18.7	16.3
AFS [31]	18.6	17.2
Curriculum [4]	18.0	-
LUT [32]	18.3	-
STAST [33]	18.7	-
W-Transf.	14.6	13.0
XSTNet (Base)	21.0	18.8
XSTNet (Expand)	**21.5**	**19.5**

Table 2: *Performance (case-insensitive tokenized BLEU and case-sensitive detokenized BLEU) on LibriTrans En-Fr test set.*

on dev-set and average the last 10 checkpoints. We use a beam size of 10 for decoding. We report the case-sensitive detokenized BLEU using sacreBLEU[6] [35] for MuST-C dataset. We also list both case-insensitive tokenized (Uncased-tok) BLEU and case-sensitive detokenized (Cased-detok) BLEU for LibriTrans for a fair comparison.

4.3. Main Results

We compared our method with other strong end-to-end baseline models, including: Transformer ST [13], AFS [31], Dual-decoder Transformer [15], STAST [33], Tang et al. [2], Curriculum pretrain [4], LUT [32], and FAT-ST [6]. These baselines all take the 80-channel log Mel-filter bank feature (fbank80) as the audio inputs. We also implement wav2vec2+transformer model (abbreviated **W-Transf.**) with the same configuration as XSTNet. **XSTNet (Base)** is only trained based on MuST-C by optimizing $\mathcal{L} = -\mathbb{E}_{\mathbf{x},\mathbf{y} \in \mathcal{D}} \log P(\mathbf{y}|\mathbf{x})$. **XSTNet (Expand)** uses external WMT data and follows the progressive multi-task training method described in Section 3.4.

Table 1 and Table 2 respectively show the BLEU scores of MuST-C and LibriTrans datasets. Compared to the current speech transformer, which acts as a solid baseline for ST, our method achieves a remarkable +3.9 BLEU gain on average. We attribute the gain to the following factors:

Wav2vec2.0 vs. Fbank Comparing the models using the fbank80 feature with the ones applying wav2vec2, we find that even the simplest structure (W-Transf.) without ASR pre-training outperforms the Transformer ST baseline. This indicates the potential of wav2vec2.0 representation.

Multi-task vs. ST-only Comparing XSTNet (Base) and W-Transf., we find that the translation performance improved by around +2 BLEU in all directions. This demonstrates that our model can make the most of the triple-supervised data by applying the multi-task paradigm.

Additional MT data Since XSTNet accepts text input, it is

Footnote
[6]https://github.com/mjpost/sacrebleu

easy to introduce additional MT parallel data. XSTNet (Expand) attributes an average of +1.3 BLEU improvement against XSTNet-Base from the additional MT data and the training procedure described in Section 3.4.

4.4. Results on Auxiliary MT and ASR Tasks

XSTNet can perform MT and ASR tasks provided different input modalities and target language indicators. The performance of auxiliary MT and ASR (measured in WER) tasks are shown in Table 3 and Table 4. Our XSTNet with progressive training is better than other training strategies in both tasks. Comparing Row "I" and "IV" in Table 3, it is worth noting that adding speech-to-text data improves MT performance even further. We attributed the improvement to the information from the speech.

Pretrain	Finetune	En-De	En-Fr	En-Ru
-	$\mathcal{D}_{MT} \cup \mathcal{D}_{MT\text{-}ext}$	31.65	43.46	21.06
-	$\mathcal{D}$	30.74	42.91	19.54
$\mathcal{D}_{MT\text{-}ext}$	$\mathcal{D}$	32.82	44.82	21.29
$\mathcal{D}_{MT\text{-}ext}$	$\mathcal{D} \cup \mathcal{D}_{MT\text{-}ext}$	**33.20**	**45.25**	**21.51**

Table 3: *Comparisons of the auxiliary MT tasks among different multi-task training strategies. The model in the first row is trained based on the external MT data $\mathcal{D}_{MT\text{-}ext}$ and the transcript-translation parallel data in MuST-C.*

Pretrain	Finetune	En-De	En-Fr	En-Ru
-	$\mathcal{D}_{ASR}$	12.99	12.27	12.05
-	$\mathcal{D}$	11.08	10.74	10.94
$\mathcal{D}_{MT\text{-}ext}$	$\mathcal{D}$	10.98	10.44	10.90
$\mathcal{D}_{MT\text{-}ext}$	$\mathcal{D} \cup \mathcal{D}_{MT\text{-}ext}$	**10.80**	**10.38**	**10.86**

Table 4: *Comparisons of the auxiliary ASR tasks, measured by word error rate (WER↓) versus the reference English transcript.*

4.5. Comparsion with Cascaded Baselines

We also compare XSTNet to the cascaded models, i.e. Transformer ASR →Transformer MT in Table 6. We also implemented a strong cascaded model whose ASR part is trained based on W-Transf., and the MT part is Transformer-base model trained based on $\mathcal{D}_{MT} \cup \mathcal{D}_{MT\text{-}ext}$. The results show that XSTNet-expand translates better, meaning that XSTNet can avoid the error propagation problem that plagues the cascaded models.

5. Analysis

5.1. The Influence of Training Procedure

In this section, we investigate the impact of the training strategy. We perform six groups of experiments with different pre-train

	#Exp.	Pretrain	Finetune	En-De	En-Fr	En-Ru	Avg.
MLT	I	$\mathcal{D}_{\text{MT-ext}}$	$\mathcal{D} \cup \mathcal{D}_{\text{MT-ext}}$	**27.12**	**38.01**	**18.36**	**27.8**
	II	$\mathcal{D}_{\text{MT-ext}}$	$\mathcal{D}$	26.99	37.37	18.03	27.5 (-0.3)
	III	-	$\mathcal{D} \cup \mathcal{D}_{\text{MT-ext}}$	26.28	36.15	17.33	26.6 (-1.2)
ST-only	IV	$\mathcal{D}_{\text{MT-ext}} \to \mathcal{D}_{\text{ASR}} \cup \mathcal{D}_{\text{MT}} \cup \mathcal{D}_{\text{MT-ext}}$	$\mathcal{D}_{\text{ST}}$	26.96	35.42	17.87	26.6 (-1.2)
	V	$\mathcal{D}_{\text{MT-ext}} \to \mathcal{D}_{\text{ASR}} \cup \mathcal{D}_{\text{MT}}$	$\mathcal{D}_{\text{ST}}$	25.86	34.44	16.94	25.6 (-2.2)
	VI	$\mathcal{D}_{\text{MT-ext}} \cup \mathcal{D}_{\text{MT}} \to \mathcal{D}_{\text{ASR}}$	$\mathcal{D}_{\text{ST}}$	24.32	35.30	17.50	25.7 (-2.1)

Table 5: *The ablation study results on the pre-train and fine-tuning strategies, where $\mathcal{D} = \mathcal{D}_{ST} \cup \mathcal{D}_{ASR} \cup \mathcal{D}_{MT}$, and Exp.I uses the training strategy described in Section 3.4. Experiments are preformed on MuST-C En-De, En-Fr and En-Ru datasets.*

	Models	En-De	En-Fr	En-Ru
Cascaded	Espnet[12]	23.6	33.8	16.4
	Our implement*	25.2	34.9	17.0
End-to-end	XSTNet	**27.1**	**38.0**	**18.4**

Table 6: *XSTNet versus the cascaded models on MuST-C En-De, En-Fr and En-Ru test sets. **Our implement*** is a strong cascaded model composed of W-Transf. as ASR and Transformer-base trained on $\mathcal{D}_{MT} \cup \mathcal{D}_{MT\text{-}ext}$ as MT.*

and fine-tune strategies, categorizing into two groups according to the different fine-tuning tasks - **multi-task** fine-tuning (Exp. I, II and III) and **ST-only** fine-tuning (Exp. IV, V and VI). The multi-task fine-tuning incorporates ST, ASR and MT tasks using MuST-C dataset $\mathcal{D}(= \mathcal{D}_{ST} \cup \mathcal{D}_{ASR} \cup \mathcal{D}_{MT})$ and optional $\mathcal{D}_{MT\text{-}ext}$. In the ST-only fine-tuning group, we perform a two-stage pre-training. For example, Exp. IV first pre-trains with external MT data, then pre-trains with ASR and all the MT parallel (both MuST-C and WMT) data, following the idea of "***progressive***". Table 5 illustrates the detailed training process and their BLEU scores, with the following empirical conclusions.

MT pretraining is effective. Comparing Exp.I and III, we find that canceling the pre-training using external WMT reduces the average performance by 1.2 BLEU. Furthermore, the results of Exp. III are inferior to those of Exp. II. These findings suggest that MT pre-training provides a strong initialization for the model, leading it to perform better.

Don't stop training the data in the previous stage. An interesting discovery is that data used in the previous training stage can also be helpful in the subsequent training stage. In other words, **progressive training works.** Concretely, we can see from Exp. I vs. III and Exp. IV vs. V that the BLEUs decrease as we stop using the WMT data at the fine-tuning period.

Multi-task fine-tuning is preferred. Comparing the BLEU results in the multi-task and ST-only groups, the models trained from multi-task perform better than the models trained by ST-only tasks.

5.2. Convergence Analysis

Figure 2 depicts the evolution of the BLEU score over time steps for different training methods on the MuST-C En-De dev-set.

Progressive multi-task training converges faster. With WMT pre-trained parameters for the Transformer module, XSTNet-Expand with progressive multi-task training (red line) is found to converge faster than the model without pre-training.

Multi-task training generalizes better. Due to the small data scale, the model is tend to overfit the training set if it is trained only based on audio-translation parallel data. However, by adopting a multi-task framework, the model is more robust and has a better generalization ability.
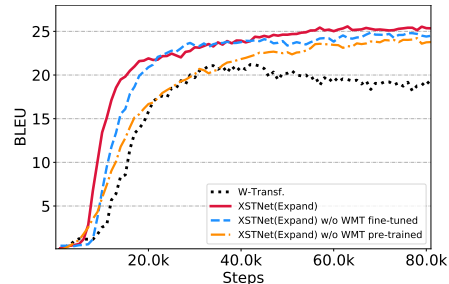

Figure 2: *The evolution of BLEU on MuST-C En-De dev set. "XSTNet (Expand) w/o WMT pre-trained" is trained directly on $\mathcal{D} \cup \mathcal{D}_{MT\text{-}ext}$. "XSTNet (Expand) w/o WMT fine-tuned" means fine-tuning only on $\mathcal{D}$, i.e. not progressively. The actual training has been performed until full convergence.*

5.3. Influence of Additional MT Data

We gradually increase the dataset size to assess the impact of the amount of external WMT data and test BLEU scores (Figure 3). There are increases in the pretrained MT BLEU and ST BLEU as the increase of the external MT data size increases. Substituting WMT with larger OpenSubtitles dataset (Table 7), the ST BLEU score of MuST-C En-De is even higher, achieving **27.8**.
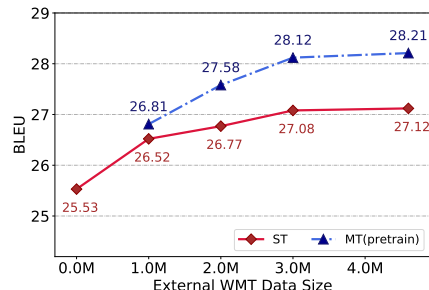

Figure 3: *BLEU curve for MuST-C En-De against the amount of external MT data used. The red curve is the evolution of the ST BLEU. The blue curve is the evolution of the MT BLEU pre-trained based on WMT data.*

Corpora	Size	MT(Pretrain)	ST
WMT	4.6M	28.2	27.1
OpenSubtitles	18M	28.5	**27.8**

Table 7: *BLEU scores for different external MT dataset (WMT v.s. OpenSubtitles)*

6. Conclusion

We propose Cross Speech-Text Network (XSTNet), an extremely concise model which can accept bi-modal inputs and jointly train ST, ASR and MT tasks. We also devise progressive multi-task training algorithm for the model. As compared to the SOTA models, XSTNet can achieve a significant improvement on the speech-to-text translation task.

7. References

[1] S. Indurthi, H. Han, N. K. Lakumarapu, B. Lee, I. Chung, S. Kim, and C. Kim, "Data efficient direct speech-to-text translation with modality agnostic meta-learning," in *Proc. of ICASSP*. IEEE, 2020.

[2] Y. Tang, J. Pino, C. Wang, X. Ma, and D. Genzel, "A general multi-task learning framework to leverage text data for speech to text tasks," in *Proc. of ICASSP*. IEEE, 2021, pp. 6209–6213.

[3] Q. Dong, M. Wang, H. Zhou, S. Xu, B. Xu, and L. Li, "Consecutive decoding for speech-to-text translation," in *Proc. of AAAI*, 2021.

[4] C. Wang, Y. Wu, S. Liu, M. Zhou, and Z. Yang, "Curriculum pre-training for end-to-end speech translation," in *Proc. of ACL*, 2020, pp. 3728–3738.

[5] S. Bansal, H. Kamper, K. Livescu, A. Lopez, and S. Goldwater, "Pre-training on high-resource speech recognition improves low-resource speech-to-text translation," in *Proc. of NAACL-HLT*, 2019, pp. 58–68.

[6] R. Zheng, J. Chen, M. Ma, and L. Huang, "Fused acoustic and text encoding for multimodal bilingual pretraining and speech translation," in *Proc. of ICML*, 2021.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of NeurIPS*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008.

[8] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. of NeurIPS*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[9] A. Bérard, O. Pietquin, C. Servan, and L. Besacier, "Listen and translate: A proof of concept for end-to-end speech-to-text translation," in *NIPS workshop on End-to-end Learning for Speech and Audio Processing*, 2016.

[10] T. Kano, S. Sakti, and S. Nakamura, "Structured-based curriculum learning for end-to-end english-japanese speech translation," in *Proc. of INTERSPEECH*, 2017, pp. 2630–2634.

[11] A. Berard, L. Besacier, A. C. Kocabiyikoglu, and O. Pietquin, "End-to-end automatic speech translation of audiobooks," in *Proc. of ICASSP*, 2018, pp. 6224–6228.

[12] H. Inaguma, S. Kiyono, K. Duh, S. Karita, N. Yalta, T. Hayashi, and S. Watanabe, "ESPnet-ST: All-in-one speech translation toolkit," in *Proc. of ACL*, 2020, pp. 302–311.

[13] C. Zhao, M. Wang, Q. Dong, R. Ye, and L. Li, "NeurST: Neural speech translation toolkit," in *Proc. of ACL - System Demonstrations*, Aug. 2021.

[14] A. Alinejad and A. Sarkar, "Effectively pretraining a speech translation decoder with machine translation data," in *Proc. of EMNLP*, 2020, pp. 8014–8020.

[15] H. Le, J. Pino, C. Wang, J. Gu, D. Schwab, and L. Besacier, "Dual-decoder transformer for joint automatic speech recognition and multilingual speech translation," in *Proc. of COLING*, 2020, pp. 3520–3533.

[16] J. Pino, Q. Xu, X. Ma, M. J. Dousti, and Y. Tang, "Self-training for end-to-end speech translation," in *Proc. of INTERSPEECH*, 2020, pp. 1476–1480.

[17] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition." in *Proc. of INTERSPEECH*, 2019.

[18] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," in *Proc. of ICLR*, 2020.

[19] A. Wu, C. Wang, J. Pino, and J. Gu, "Self-supervised representations improve end-to-end speech translation," in *Proc. of INTER-SPEECH*, 2020.

[20] H. Nguyen, F. Bougares, N. Tomashenko, Y. Estève, and L. Besacier, "Investigating self-supervised pre-training for end-to-end speech translation," in *Proc. of INTERSPEECH*, 2020.

[21] C. Tran, Y. Tang, X. Li, and J. Gu, "Cross-lingual retrieval for iterative self-supervised training," in *Proc. of NeurIPS*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[22] C. Wang, Y. Wu, Y. Qian, K. Kumatani, S. Liu, F. Wei, M. Zeng, and X. Huang, "Unispeech: Unified speech representation learning with labeled and unlabeled data," in *Proc. of ICML*, 2021.

[23] C. Yi, J. Wang, N. Cheng, S. Zhou, and B. Xu, "Applying wav2vec2.0 to speech recognition in various low-resource languages," *arXiv preprint arXiv:2012.12121*, 2021.

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. of ICASSP*, 2015, pp. 5206–5210.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. of ICLR*, Y. Bengio and Y. LeCun, Eds., 2015.

[26] M. A. Di Gangi, R. Cattoni, L. Bentivogli, M. Negri, and M. Turchi, "MuST-C: a Multilingual Speech Translation Corpus," in *Proc. of NAACL-HLT*, 2019, pp. 2012–2017.

[27] A. C. Kocabiyikoglu, L. Besacier, and O. Kraif, "Augmenting librispeech with French translations: A multimodal corpus for direct speech translation evaluation," in *Proc. of LREC*, 2018.

[28] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. Jimeno Yepes, P. Koehn, V. Logacheva, C. Monz, M. Negri, A. Névéol, M. Neves, M. Popel, M. Post, R. Rubino, C. Scarton, L. Specia, M. Turchi, K. Verspoor, and M. Zampieri, "Findings of the 2016 conference on machine translation," in *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, 2016, pp. 131–198.

[29] B. Zhang, P. Williams, I. Titov, and R. Sennrich, "Improving massively multilingual neural machine translation and zero-shot translation," in *Proc. of ACL*, 2020, pp. 1628–1639.

[30] P. Lison and J. Tiedemann, "OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles," in *Proc. of LREC*, 2016, pp. 923–929.

[31] B. Zhang, I. Titov, B. Haddow, and R. Sennrich, "Adaptive feature selection for end-to-end speech translation," in *Proc. of EMNLP - FIndings*, 2020, pp. 2533–2544.

[32] Q. Dong, R. Ye, M. Wang, H. Zhou, S. Xu, B. Xu, and L. Li, "Listen, understand and translate: Triple supervision decouples end-to-end speech-to-text translation," in *Proc. of AAAI*, vol. 35, no. 14, 2021, pp. 12 749–12 759.

[33] Y. Liu, J. Zhu, J. Zhang, and C. Zong, "Bridging the modality gap for speech-to-text translation," *arXiv preprint arXiv:2010.14920*, 2020.

[34] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proc. of EMNLP*, 2018, pp. 66–71.

[35] M. Post, "A call for clarity in reporting BLEU scores," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, 2018, pp. 186–191.