The NiuTrans's Submission to the IWSLT22 English-to-Chinese Offline Speech Translation Task

Yuhao Zhang[1], **Canan Huang**[1], **Chen Xu**[1], **Xiaoqian Liu**[1], **Bei Li**[1], **Anxiang Ma**[1,2], **Tong Xiao**[1,2] **and Jingbo Zhu**[1,2]
[1]NLP Lab, School of Computer Science and Engineering
Northeastern University, Shenyang, China
[2]NiuTrans Research, Shenyang, China
yoohao.zhang@gmail.com, xuchenneu@outlook.com
{maanxiang, xiaotong, zhujingbo}@mail.neu.edu.cn

Abstract

This paper describes NiuTrans's submission to the IWSLT22 English-to-Chinese (En-Zh) offline speech translation task. The end-to-end and bilingual system is built by constrained English and Chinese data and translates the English speech to Chinese text without intermediate transcription. Our speech translation models are composed of different pre-trained acoustic models and machine translation models by two kinds of adapters. We compared the effect of the standard speech feature (e.g. log Mel-filterbank) and the pre-training speech feature and try to make them interact. The final submission is an ensemble of three potential speech translation models. Our single best and ensemble model achieves 18.66 BLEU and 19.35 BLEU separately on MuST-C En-Zh tst-COMMON set.

1 Introduction

Speech translation is the task that transfers the speech input to the target language text. Comparing the cascade of automatic speech recognition (ASR) and machine translation (MT) systems, recently the end-to-end speech translation (E2E ST, for short ST) model arises more attention for its low latency and avoiding error propagation (Pino et al., 2020; Wang et al., 2020; Xu et al., 2021a; Indurthi et al., 2021). On the IWSLT21 offline speech translation task, the ST has shown its potential ability compared with cascade systems by using ASR and MT labeled data to pre-train modules of the ST model (Bahar et al., 2021). We explore that using different speech features and model architecture for the ST model can further lessen the gap with the cascade system. We design a model which fuses the two speech features to enrich speech information.

In our submission, we pre-train the machine translations model and choose the deep Transformer (Wang et al., 2019), ODE Transformer (Li et al., 2021a) and MBART (Liu et al., 2020) as MT backbone architectures. For the acoustic model, we use a progressive down-sampling method (PDS) and Wav2vec 2.0 (W2V) (Baevski et al., 2020). To integrate the pre-trained acoustic and textual model, we use the SATE method (Xu et al., 2021a) which adds an adapter between the acoustic and textual model. To utilize the model pre-trained by unlabeled data, such as W2V, and MBART, we purpose the multi-stage pre-training method toward ST (MSP) and add the MSP-Adapter to boost the ST performance. Manuscripts for the MSP and PDS are in preparation. We fuse the output feature of the PDS encoder and W2V with the multi-head attention of the decoder. The input of the former is a standard speech feature while the latter is a waveform. We evaluate the relation between the effect of the ensemble model and the diversity of model architecture.

Our best MT model reaches 19.76 BLEU and our ST model reaches 18.66 BLEU on the MuST-C En-ZH tst-COMMON set. While the ensemble model achieves 19.35 which shows the performance of ST can be further improved. The model that fuses two strong encoders does not outperform the model with a single encoder. We show the diversity of models is important during the ensemble stage. We find the bottleneck of our ST model is the de-noising and translating ability of MT modules.

2 Data

2.1 Data pre-processing

MT Due to the WMT21 task aiming at the news domain, we only choose the high-quality ones from WMT21 corpora. We fellow the Zhang et al. (2020) to clean parallel texts. The OpenSubtitle is the in-domain corpus but many translations do not match their source texts. We use the fast-align (Dyer et al.,

Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022), pages 232 - 238

Task	Corpus	Sentence	Hour
MT	CWMT	5.08M	-
	News commentary	0.29M	-
	UN	5.68M	-
	OpenSubtitle	4.14M	-
	Total	15.19M	-
ASR	Europarl-ST	0.03M	77
	Common Voice	0.28M	415
	VoxPopuil	0.18M	496
	LibriSpeech	0.28M	960
	TED LIUM	0.26M	448
	MuST-C V1	0.07M	137
	ST TED	0.16M	234
	MuST-C En-Zh	0.36M	571
	Total	1.61M	3338
ST	MuST-C En-Zh	0.35M	571

Table 1: Detail of labeled data

2013) to score all the sentence. We average the score by the length of the corresponding sentence and filter sentences below the score of -6.0. Since the news translation is always much longer than the spoken translation, we filter sentences with more than 100 words.

ASR Following the previous work (Xu et al., 2021b), we unify all the audio to the 16000 per second sample rate and single channel. The Common voice corpus consists of many noises, so we choose the cleaner part according to the CoVoST corpus. For the MuST-C V1 corpus, we remove repetitive items comparing the MuST-C En-Zh transcriptions. We use the Librispeech set to build the ASR system and then score the Common Voice, TED LIUM, and ST TED three corpora. The sentence that the WER is higher than 75% will be removed. We filter frames with lengths less than 5 or larger than 3000. We remove the utterances with the size of characters exceeding 400.

ST Since ST data is scarce, we only filter the data according to the frame lengths and the standard is the same as ASR. We segment the final test speech by the WebRTC VAD tool[1]. We control the size of the speech slices to make sure the length distribution is similar to the training set.

[1]https://github.com/wiseman/py-webrtcvad

Task	Corpus	Sentence	Hour
MT	TED	0.51M	-
ST	Europarl-ST	0.03M	77
	Common Voice	0.27M	415
	VoxPopuil	0.17M	496
	TED LIUM	0.26M	442
	MuST-C V1	0.06M	137
	ST TED	0.15M	233
	MuST-C En-Zh	0.35M	571
	Perturbation MuST-C En-Zh	0.71M	1142
	Total	2.03M	3513

Table 2: Detail of pseudo data

2.2 Data Augmentation

MT The MT is sensitive to the domain (Chu and Wang, 2018), so we only back-translate the monolingual data in the TED talk corpus as the pseudo parallel data.

ASR We only use the SpecAugment (Park et al., 2019) to mask the speech feature.

ST We use an MT model to translate transcriptions to build the pseudo tuple data. And we transform the MuST-C audio by speed rates of 0.9 and 1.1 to perturb the speech.

The Table 1 and Table 2 show the sizes of training data. We segment the English and Chinese text by Moses (Koehn et al., 2007) and NiuTrans (Xiao et al., 2012) separately. We use sentencepiece (Kudo and Richardson, 2018) to cut them to sub-word and the model is the same as MBART.

3 Model

We explore the performances of different ASR, MT, and adapter architectures. We experiment with three MT models, two ASR models and two adapters that integrate the MT and ASR to the ST model.

3.1 MT Model

The deep Transformer has been successfully used in translation task (Li et al., 2019). It deepens the encoder layer to obtain a stronger ability to model the source language. The ODE Transformer (Li et al., 2021a) also reached the state-of-art performance based on the vanilla deep model due to the efficient use of parameters. Since the output of

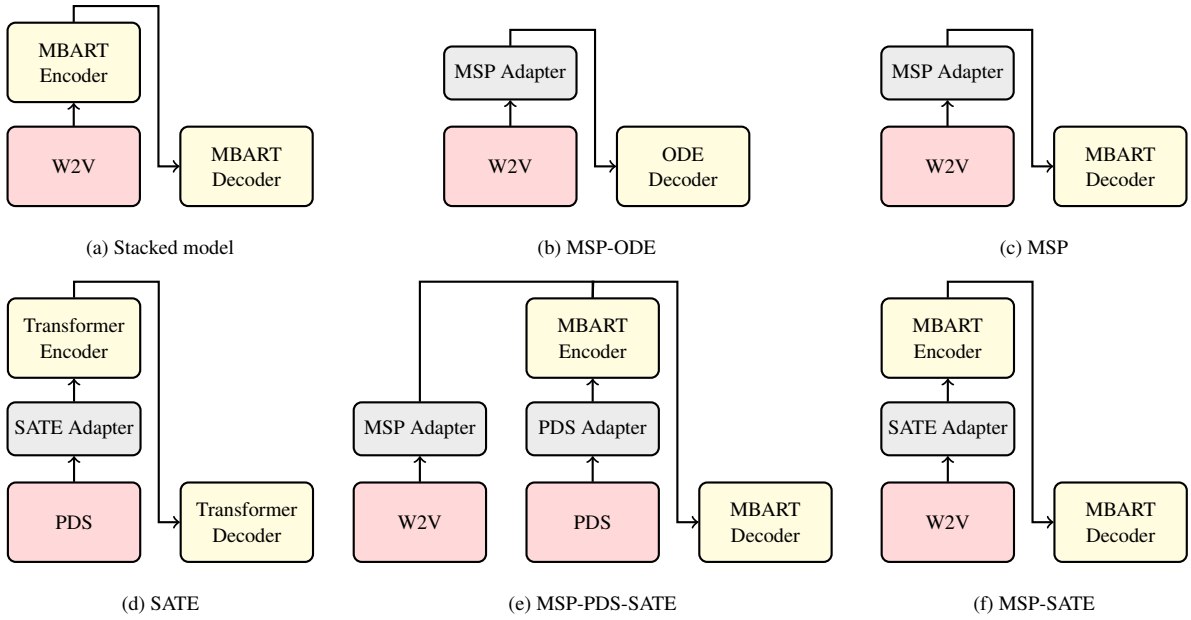

Figure 1: Overview of different ST models

the acoustic model consists of much noise, the Denoising self-encoding (DAE) model (e.g. MBART) can handle well about this situation. Further, the MBART pre-trained by lots of multilingual unlabeled data is helpful for the cross-lingual learning task. So we choose the above three models as our translation backbone models. Considering the output of the acoustic model does not contain the punctuation, we remove the punctuation in the source text before training the MT system. This operation is a little harmful to the MT model but does help the end-to-end system.

3.2 ASR Model

We use a progressive down-sampling method PDS for acoustic encoding based on Conformer which could improve the ASR performance. We also use the MSP method to fine-tune the W2V on the ASR task and can better bridge the gap between ASR and MT model. The input of the PDS model is the log Mel-filterbank feature while the W2V is based on waveform. Besides, acoustic models implement the relative position encoding (Dai et al., 2019).

3.3 ST Model

We combine the pre-trained modules with several adapters then fine-tune them with ST data. Besides the widely used Adapter consisting of a single hidden-layer feed-forward network (Bapna and Firat, 2019), we also use the SATE (Xu et al., 2021a) and MSP adapter. As Figure 1 shows, there are mainly six kinds of combined architecture we trained. Figure 1 (a) shows the W2V and MBART are stacked with the Adapter. The Figure 1 (b) and (c) show the W2V and MSP-adapter combined different MT decoders. The ST models composed with SATE adapter are shown in Figure 1 (d) and (f). As Figure 1 (e) shows, we fuse the output of two encoders which the input is filter-bank and waveform to make the different features interact. We use the cross multi-head attention of the decoder to extract two features and then average them.

4 Fine-tuning and Ensemble

To adjust the composed model to the ST task and a certain domain, we use the whole ST data to fine-tune the model. After coverage, we continue to train the model with only the MuST-C data set for domain adaptation.

We ensemble ST models by averaging distributions of model output. We search different combinations and numbers of models on the MuST-C set to investigate the influence of structural differences on the results of the ensemble model.

Since the final segmentation on the test set is inconsistent with the training set, we re-segment the training set by the same hyper-parameters as the test set. To get the reference of the audio, we implement the ensemble model to decode all the training audios and use the WER to re-cut the gold training paragraph into sentences. We utilize the new re-segment set to fine-tune the models.

Model	#Param	Dev	tst-COMMON
Baseline	54M	14.34	16.92
+parallel data	77M	16.48	18.74
+pseudo data	77M	16.81	18.74
+deep encoder	165M	16.91	19.76
ODE	104M	16.44	18.77
MBART	421M	16.04	18.12
Deep model	165M	16.23	18.96

Table 3: MT model measured by BLEU [%] metric

Model	#Param	Dev	tst-COMMON
PDS	127M	6.89	5.33
W2V	602M	4.89	5.31

Table 4: ASR model measured by WER [%] metric

Model	#Param	Dev	tst-COMMON
Single MT	165M	16.91	19.76
Transformer	30M	11.37	13.27
MSP	607M	14.96	17.19
+Pseudo data	607M	14.62	17.47
+Fine-tuning	607M	15.65	18.54
+Resegmentation	607M	15.26	18.41
+Ensemble	-	16.42	19.35

Table 5: ST model measured by BLEU [%] metric

Model	tst-COMMON	Ref2	Ref1	Both
MSP	26.7	-	-	-
Ensemble	29.1	32.3	33.2	40.5

Table 6: BLEU scores of ST models on MuST-C tst-COMMON and submitted tst2022 set. The scores are measured by the SLT.KIT toolkit.

5 Experiments

5.1 Experiment Settings

For the deep Transformer, we increased the encoder layers to 30 and keep the decoder 6 layers, the hidden size and FFN size is the same as the Transformer-base configuration. The ODE Transformer consisted of 18 encoder layers and 6 decoder layers. The pre-trained MBART consisted of a 12 layers encoder and a 12 layers decoder. All the models were trained with the pre-normalization operation. The size of the shared vocabulary was 44,144.

We used the pre-trained W2V model which does not fine-tune on the ASR task. We added the MSP-Adapter after the W2V and fine-tuned the model following the Baevski et al. (2020) fine-tuning configuration. During training on the ST set, we froze many parameters followed by Li et al. (2021b) to avoid catastrophic forgetting. The learning rate is set 3e-5 and we set drop and label smoothing at 0.2 to avoid over-fitting.

We implemented the early stop if the model does not promote for 8 times. We averaged the weights of the last 5 checkpoints for each training task. The beam size of inference was 8. All the MT and ST scores were calculated by multi-BLEU [2]. The ASR system was evaluated by word error rate (WER).

5.2 Results

MT Table 3 shows the MT results on the MuST-C dev and tst-COMMON set. Adding out-domain massive parallel data can significantly improve the performance. Though we add very few in-domain pseudo data, there is a +0.32 improvement on the dev set. The deep model gains +1.02 BLEU which significantly increases the ability of the MT model. To be consistent with the output of the acoustic model, we lowercase the English text and remove the punctuation. The MT results show a little degradation of performance while it is helpful for the end-to-end system. The MBART does not show its advantage compared with other methods. We conjecture that the exclusive model is better to deal with the Chinese translation task when there are dozen millions of clean parallel texts.

ASR There are two main architectures used for the ASR task. The PDS receives the log Mel-filterbank feature which is pre-processed while the input of W2V is the original sampling point of the waveform. Table 4 shows that W2V has much more parameters and achieves much better performance on the dev set. But the two models are comparable on the tst-COMMON set. This shows the W2V model is easy to over-fit.

ST Table 5 shows the MSP method which integrates pre-trained W2V and MBART modules to gain significant improvement compared with the vanilla Transformer model. We find directly adding pseudo data does not have an obvious effect. But after fine-tuning the MuST-C set, the improvement is significant. This shows the ST model is still

[2]https://github.com/moses-smt/mosesdecoder

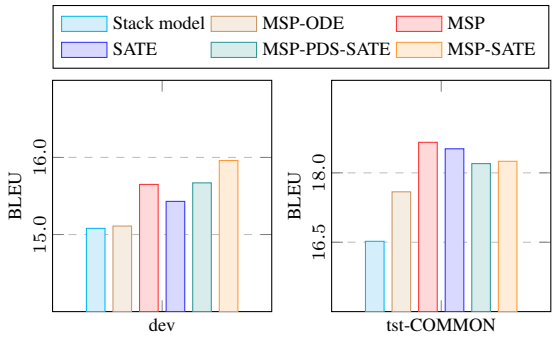

Figure 2: Comparison of the performance of the different models on MuST-C dev and tst-COMMON set

sensitive to the domain.

We compare the six combined architectures in Figure 2. Directly stacking two pre-trained models get the worst performance, this causes by the gap between the ASR and MT model. The ODE model has a stronger translation ability than the MBART, but the MSP-ODE does not outperform MSP on the ST task. We think it is due to the de-noising ability of the MBART since much noise such as silence exists in speech features. The MSP and the SATE get comparable performance on the tst-COMMON set and MSP-SATE which combined two methods gets the highest on the dev set. This proves the effect of MSP and SATE methods. We use the MSP-PDS-SATE to fuse two kinds of speech features and this model has about 900 million parameters. But the performance is not good enough. It needs to further explore how to make the pre-trained and original features interact.

To compare with other work conveniently, we provide some tst-COMMON results measured by official scripts [3] and each hypothesis is resegmented based on the reference by mwerSegmenter. The final results which are supplied by Anastasopoulos et al. (2022) in Table 6.

Ensemble The Table 5 shows the effect of ensemble model is also remarkable. We compared the performance of different combinations in Table 7. The fine-tuned model is likely over-fitting and we find the ensemble of the un-fine-tuned model is useful. We ensemble two models with much different architecture and the resulting gain is +0.56 improvement. We further add another different model but only gain slight improvement. We replace the MSP model with a worse model while the performance does not degenerate. This proves the ensemble model prefers the combination of models with a great difference and when the number of models increases, the performance of a single model does not matter.

[3] https://github.com/isl-mt/SLT.KIT/blob/master/scripts/evaluate/Eval.sh

Combination	tst-COMMON
MSP	18.66
MSP+MSP-UFT	18.99
MSP+SATE	19.22
MSP+SATE+MSP-SATE	19.35
MSP-UFT+SATE+MSP-SATE	19.34

Table 7: Ensemble model results measured by BLEU [%] metric. The MSP-UFT indicates the MSP model is un-fine-tuned.

6 Conclusions

This paper describes our submission to the IWSLT22 English to Chinese offline speech translation task. Our system is end-to-end and constrained. We pre-trained three types of machine translation models and two automatic speech recognition models. We integrate the acoustic and translation model on speech translation tasks by two types of adapters MSP and SATE. We fine-tune models to adapt domain and search for the best ensemble model for our submission. Our final system achieves 19.35 BLEU on MuST-C En-Zh tst-COMMON set.

Acknowledgments

This work was supported in part by the National Science Foundation of China (Nos. 61732005 and 61876035), the China HTRD Center Project (No. 2020AAA0107904) and Yunnan Provincial Major Science and Technology Special Plan Projects (Nos. 201902D08001905 and 202103AA080015). The authors would like to thank anonymous reviewers for their valuable comments. Thank Hao Chen and Jie Wang for processing the data.

References

Antonios Anastasopoulos, Luisa Bentivogli, Marcely Z. Boito, Ondřej Bojar, Roldano Cattoni, Anna Currey, Georgiana Dinu, Kevin Duh, Maha Elbayad, Marcello Federico, Christian Federmann, Hongyu Gong, Roman Grundkiewicz, Barry Haddow, Benjamin Hsu, Dávid Javorský, Věra Kloudová, Surafel M. Lakew, Xutai Ma, Prashant Mathur, Paul McNamee, Kenton Murray, Maria Nădejde, Satoshi Nakamura, Matteo Negri, Jan Niehues, Xing Niu, Juan Pino, Elizabeth Salesky, Jiatong Shi, Sebastian Stüker, Kat-

suhito Sudoh, Marco Turchi, Yogesh Virkar, Alex Waibel, Changhan Wang, and Shinji Watanabe. 2022. FINDINGS OF THE IWSLT 2022 EVALUATION CAMPAIGN. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, Dublin, Ireland. 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*, volume 33, pages 12449–12460.

Parnia Bahar, Patrick Wilken, Mattia A. Di Gangi, and Evgeny Matusov. 2021. Without further ado: Direct and simultaneous speech translation by AppTek in 2021. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 52–63, Bangkok, Thailand (online). Association for Computational Linguistics.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Sathish Indurthi, Mohd Abbas Zaidi, Nikhil Kumar Lakumarapu, Beomseok Lee, Hyojung Han, Seokchan Ahn, Sangha Kim, Chanwoo Kim, and Inchul Hwang. 2021. Task aware multi-task learning for speech to text tasks. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7723–7727.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. 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*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Bei Li, Quan Du, Tao Zhou, Shuhan Zhou, Xin Zeng, Tong Xiao, and Jingbo Zhu. 2021a. Ode transformer: An ordinary differential equation-inspired model for neural machine translation. *arXiv preprint arXiv:2104.02308*.

Bei Li, Yinqiao Li, Chen Xu, Ye Lin, Jiqiang Liu, Hui Liu, Ziyang Wang, Yuhao Zhang, Nuo Xu, Zeyang Wang, Kai Feng, Hexuan Chen, Tengbo Liu, Yanyang Li, Qiang Wang, Tong Xiao, and Jingbo Zhu. 2019. The NiuTrans machine translation systems for WMT19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 257–266, Florence, Italy. Association for Computational Linguistics.

Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2021b. Multilingual speech translation from efficient finetuning of pretrained models. 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)*, pages 827–838, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

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 *Proc. Interspeech 2019*, pages 2613–2617.

Juan Pino, Qiantong Xu, Xutai Ma, Mohammad Javad Dousti, and Yun Tang. 2020. Self-Training for End-to-End Speech Translation. In *Proc. Interspeech 2020*, pages 1476–1480.

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, pages 3728–3738, Online. Association for Computational Linguistics.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. 2019. Learning deep transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1810–1822, Florence, Italy. Association for Computational Linguistics.

Tong Xiao, Jingbo Zhu, Hao Zhang, and Qiang Li. 2012. NiuTrans: An open source toolkit for phrase-based and syntax-based machine translation. In *Proceedings of the ACL 2012 System Demonstrations*, pages 19–24, Jeju Island, Korea. Association for Computational Linguistics.

Chen Xu, Bojie Hu, Yanyang Li, Yuhao Zhang, Shen Huang, Qi Ju, Tong Xiao, and Jingbo Zhu. 2021a. 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)*, pages 2619–2630, Online. Association for Computational Linguistics.

Chen Xu, Xiaoqian Liu, Xiaowen Liu, Tiger Wang, Canan Huang, Tong Xiao, and Jingbo Zhu. 2021b. The NiuTrans end-to-end speech translation system for IWSLT 2021 offline task. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 92–99, Bangkok, Thailand (online). Association for Computational Linguistics.

Yuhao Zhang, Ziyang Wang, Runzhe Cao, Binghao Wei, Weiqiao Shan, Shuhan Zhou, Abudurexiti Reheman, Tao Zhou, Xin Zeng, Laohu Wang, Yongyu Mu, Jingnan Zhang, Xiaoqian Liu, Xuanjun Zhou, Yinqiao Li, Bei Li, Tong Xiao, and Jingbo Zhu. 2020. The NiuTrans machine translation systems for WMT20. In *Proceedings of the Fifth Conference on Machine Translation*, pages 338–345, Online. Association for Computational Linguistics.