The NiuTrans End-to-End Speech Translation System for IWSLT23 English-to-Chinese Offline Task

Yuchen Han[1]*, Xiaoqian Liu[1]*, Hao Chen[1], Yuhao Zhang[1], Chen Xu[1], Tong Xiao[1,2], Jingbo Zhu[1,2]

[1]School of Computer Science and Engineering, Northeastern University, Shenyang, China

[2]NiuTrans Research, Shenyang, China

{hanyuchen114,yoohao.zhang}@gmail.com,methanechen@126.com

{liuxiaoqian0319,xuchennlp}@outlook.com

{xiaotong,zhujingbo}@mail.neu.edu.cn

Abstract

This paper describes the NiuTrans end-to-end speech translation system submitted for the IWSLT 2023 English-to-Chinese offline task. Our speech translation models are composed of pre-trained ASR and MT models under the stacked acoustic and textual encoding framework. Several pre-trained models with diverse architectures and input representations (e.g., log Mel-filterbank and waveform) were utilized. We proposed an iterative data augmentation method to iteratively improve the performance of the MT models and generate the pseudo ST data through MT systems. We then trained ST models with different structures and data settings to enhance ensemble performance. Experimental results demonstrate that our NiuTrans system achieved a BLEU score of 29.22 on the MuST-C En-Zh tst-COMMON set, outperforming the previous year's submission by 0.12 BLEU despite using less MT training data.

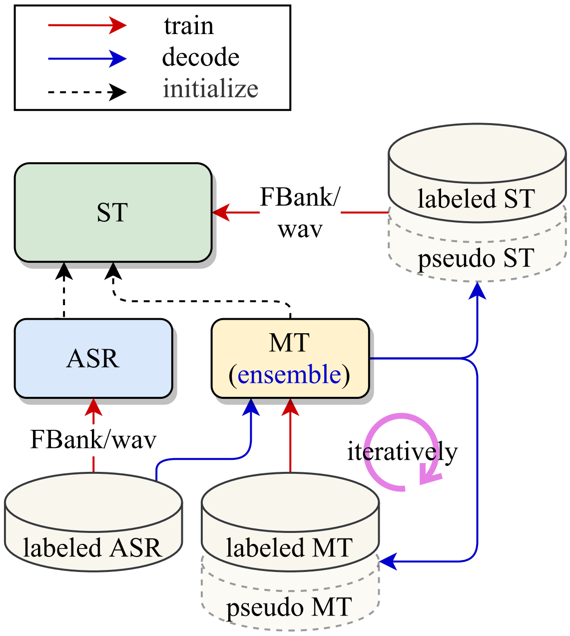

Figure 1: Overview of our system.

1 Introduction

End-to-end speech translation (E2E ST) directly translate speech in the source language into text in the target language without generating an intermediate representation, which has gained significant attention in recent years due to several advantages over cascade methods, including low latency and the ability to avoid error propagation (Berard et al., 2016; Weiss et al., 2017). In this paper, we describe our NiuTrans E2E ST system that participated in the IWSLT23 English-to-Chinese offline track, the overview of our system is shown in Fig 1.

To improve the performance of our system, we aim to maximize the diversity of our ensemble of E2E ST models. Our E2E ST models are based on the stacked acoustic and textual encoding (SATE) method (Xu et al., 2021a), which is a framework to make the best of pre-trained automatic speech recognition (ASR) and machine translation (MT) components. Using this framework, we explore multiple architectures of pre-trained ASR and MT models with varying numbers of parameters and input representations such as FBank features or waveform data.

Pseudo data is a crucial component of E2E ST, often generated by ensemble MT systems (Gaido et al., 2020). This year, we focused more on the performance of MT models and developed an Iterative Data Augmentation method to leverage text data from all corpora, improving the MT models and enabling the generation of multiple pseudo data. We then used these multiple pseudo data to train diverse E2E ST models for optimal performance. Our best ST ensemble system includes models with different input representations, architectures, and training corpora, achieving a BLEU score of 29.22 on the MuST-C En-Zh tst-COMMON set.

The remainder of the paper is organized as follows: Section 2 describes the data processing, data

*Authors contributed equally.

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 211–218
July 13-14, 2023

augmentation and speech segmentation. Section 3 outlines the construction of the vocabulary and structures of our ASR, MT and ST models. The experimental settings and final results are presented in Section 4. Finally, Section 5 concludes the submission.

2 Data

2.1 Data Processing

Our system was built under the "constrained" training condition. The training data can be divided into three categories: ASR, MT, and ST corpora. We used the NiuTrans toolkit (Xiao et al., 2012) to segment English and Chinese text in all corpora.

ASR corpora. We followed the previous work (Xu et al., 2021b) and standardized all audio samples to a single channel and a sample rate of 16,000 Hz. For the Common Voice corpus, we selected only the cleaner parts according to the CoVoST v2 En-Zh corpus. In the MuST-C v1 En-De corpus, we removed repetitive items by comparing the MuST-C v2 En-Zh transcriptions. We used the Librispeech corpus to train the ASR model and scored the Common Voice, TED LIUM, and ST TED corpus. Data with a WER greater than 0.75 were removed, and frames with lengths less than 5 or greater than 3000 were filtered. In addition, utterances with more than 400 characters were removed.

MT corpora. Following the methodology of (Zhang et al., 2020), we cleaned the parallel texts of the OpenSubtitle corpus and used fast-align to score all sentences. We averaged the scores by the sentence length and filtered out sentences with scores below -6.0. In the News Commentary v16 corpus, we used langid (Lui and Baldwin, 2012) to filter out sentences with incorrect language identification results. In the Tatoeba corpus, we converted 90% of the sentences from traditional Chinese to simplified Chinese using OpenCC[1].

ST corpora. For the MuST-C v2 En-Zh and CoVoST v2 En-zh corpus, we only filtered frames by length, similar to the ASR corpora. For the pseudo ST data, we removed sentences containing repeated n-gram words (n is 2 to 4) more than four times. Additionally, sentences with length ratios outside the range of 0.25 to 4 and those with incorrect language identification results were filtered out.

[1] https://github.com/BYVoid/OpenCC

Task	Corpus	Sentence	Hour
ASR	LibriSpeech	0.28	960
	Europarl-ST	0.03	77
	TED LIUM	0.26	448
	ST TED	0.16	235
	VoxPopuil	0.17	478
	MuST-C V1 En-De	0.07	138
	MuST-C V2 En-Zh	0.36	572
	CoVoST v2 En-Zh	0.28	416
	Total	1.61	3324
MT	News Commentary	0.31	-
	OpenSubtitle	8.62	-
	MuST-C V2 En-Zh	0.36	-
	CoVoST V2 En-Zh	0.28	-
	Tatoeba	0.05	-
	Total	9.62	-
ST	MuST-C En-Zh	0.36	572
	CoVoST V2 En-Zh	0.28	416
	Total	0.64	988

Table 1: Details about the size of all labeled corpora. The unit of sentence is million (M).

Task	Corpus	Sentence	Hour
MT	ASR corpora+MT	1.38	-
ST	ASR corpora+MT	1.61	3323
	Audio+ASR+MT	1.4e-2	3

Table 2: Details about the size of all pseudo corpora.

2.2 Data Augmentation

We only used SpecAugment (Bahar et al., 2019) and not used speed perturb for ASR data augmentation, because speed perturb requires more training resources but has the limited improvement. It is also worth noting that we did not use back translation technology in either MT or E2E ST, as there was no target-side monolingual data available.

The MT model or ensemble MT systems represent the upper limit for E2E ST. Translating the transcript in the ASR corpus into the target language using MT models is a simpler and more effective way to augment the ST corpus than generating source speech features from the source texts in the MT corpus using TTS models. Based on this, we propose an **I**terative **D**ata **A**ugmentation (IDA) method, which aims to use text data from all corpora to improve the performance of MT models and generate high-quality ST corpus iteratively, as illustrated in Algorithm 1.

We also discovered incomplete transcriptions in a few sentences from the TED LIUM, ST-TED, and voxpupil corpus. Therefore, we generated pseudo transcriptions using the ASR model and then translated them using the best MT ensemble systems.

Algorithm 1: IDA

Input: $D_{ASR} = \{(s_{asr}, x_{asr})\}, D_{MT} = \{(x_{mt}, y_{mt})\}$
Output: $D^*_{ST_{aug}} = \{(s_{asr}, x_{asr}, y'_{asr})\}$

$D^*_{MT} \leftarrow D_{MT}$;
$s^* \leftarrow 0$;
for $i \leftarrow 1$ **to** MAXITER **do**
  $M_1, M_2, \cdots, M_n \leftarrow \text{train}(D^*_{MT})$;
  $E^i \leftarrow \text{ensemble}(M_1, M_2, \cdots, M_n)$;
  $s^i \leftarrow \text{score}(E^i)$;
  if $i \neq 1$ *and* $s^i <= s^*$ **then**
    break;
  else
    $y'_{asr} \leftarrow \text{decode}(E^i, x_{asr})$;
    $D^i_{MT_{aug}} \leftarrow \{(x_{asr}, y'_{asr})\}$;
    $D^i_{ST_{aug}} \leftarrow \{(s_{asr}, x_{asr}, y'_{asr})\}$;
    $D^*_{MT} \leftarrow D_{MT} \cup D^i_{MT_{aug}}$;
    $s^* \leftarrow s^i$;
return $D^*_{ST_{aug}}$;

2.3 Speech Segmentation

To avoid the significant performance drop due to the mismatch between the training and inference data, we adopted Supervised Hybrid Audio Segmentation (SHAS) (Tsiamas et al., 2022) to split long audios in the test sets. However, we did not fine-tune our models on the resegmented data, according the findings in Gaido et al. (2022).

3 Model Architecture

We explored the performances of different ASR, MT, and ST architectures and found that using larger models is more conducive to performance improvement in all three tasks.

3.1 Vocabulary

We adopted a unified vocabulary for all tasks, trained by the SentencePiece (Kudo and Richardson, 2018) model (SPM) from the MT corpora. To incorporate more subwords from the TED domain, we up-sampled the MuST-C corpus by 10x [2] in the training corpora for the SPM. The vocabulary size for English and Chinese is 10k and 44k, respectively.

[2]Specifically, we created 10 copies of the MuST-C corpus and combined them with additional MT data.

3.2 ASR Models

Inspired by Zhang et al. (2022a), we used three ASR encoders with different architectures and input representations to achieve better ensemble performance.

- Transformer-HuBERT (TH): This encoder consists of 7 layers of 512-channel-CNN with strides [5,2,2,2,2,2,2] and 12 layers of Transformer (Vaswani et al., 2017). The hidden size, ffn size, and number of heads are 768, 3072, and 8, respectively. This architecture takes waveform data as input.

- Conformer-PDS-Medium (CPM): This encoder consists of 18 layers of Conformer (Gulati et al., 2020) with progressive down-sampling (PDS) methods (Xu et al., 2023). The hidden size, ffn size, and number of heads are 512, 2048, and 8, respectively. This architecture takes log Mel-filterbank features as input.

- Conformer-PDS-Deep (CPD): This encoder is the same as the Conformer-PDS-Medium, but with the number of layers adjusted from 18 to 24.

Due to limited computational resources, we pre-trained the Transformer-HuBERT only on the Librispeech corpus using the method outlined in Hsu et al. (2021). The Conformer-PDS-Medium/Deep architectures were trained on all ASR corpora, and we employed an additional decoder with 6 layers to utilize the Cross Entropy loss. We also adopted CTC loss (Graves et al., 2006) and inter-CTC loss (Lee and Watanabe, 2021) to accelerate the convergence.

3.3 MT Models

While deep models have shown success in translation tasks, we observed that wider architectures with more parameters generally yield superior performance (Shan et al., 2022). As such, we selected the DLCL Transformer (Wang et al., 2019) and the ODE Transformer (Li et al., 2022) for the deep and wide models, respectively.

- DLCL: This model consists of 30 layers of Transformer encoder and 6 layers of Transformer decoder with dynamic linear combination of layers and relative position encoding (Shaw et al., 2018) methods. The hidden size, ffn size, and number of heads are 512, 2048, and 8, respectively.

- ODE: This model consists of 12 layers of Transformer encoder and 6 layers of Transformer decoder with an ordinary differential equation-inspired method, which has been proven to be efficient in parameters. The hidden size, ffn size, and number of heads are 1024, 4096, and 16, respectively.

- ODE-Deep: This model is the same as ODE but with the number of encoder layers adjusted from 12 to 18.

Since the transcript in the ASR corpora lacks punctuation and is in lower-case, we lowered-cased and removed punctuation from the source text of the MT corpora for consistency before training the MT models. While this operation may have a negative impact on MT performance, we have demonstrated its usefulness for data augmentation and the final ST performance in Section 4.3.

3.4 ST Models

We utilized the SATE method to enhance the usage of pre-trained ASR and MT models for the ST task. Specifically, we decoupled the ST encoder into an acoustic encoder and a textual encoder, with an adapter in between. The pre-trained ASR encoder was used to initialize the acoustic encoder, while the pre-trained MT model was used to initialize the textual encoder and decoder. To optimize performance with limited memory, we successively attempted multiple structures, ranging from small to large, as presented in Table 3. The models with TH-DLCL structure were trained using the techniques outlined in Zhang et al. (2022b).

Structure	ASR	MT	Params.
TH-DLCL	TH	DLCL	251M
CPM-DLCL	CPM	DLCL	289M
CPM-ODE	CPM	ODE	444M
CPD-ODE	CPD	ODE	472M

Table 3: The ST structures initialized with different ASR and MT models under the SATE framework.

Model	dev	tst-M	test-clean	test-other
CPM	5.01	4.17	2.81	6.51
CPD	4.76	4.25	2.86	6.10

Table 4: WER scores on the dev, tst-COMMON (tst-M), and test sets of Librispeech.

4 Experiments

4.1 Experimental settings

All experiments were implemented using the Fairseq toolkit (Ott et al., 2019). We trained all models using pre-norm and utilized dropout with a ratio ranging from 0.1 to 0.3 and label smoothing with 0.1 to prevent overfitting. Training was stopped early when the indicators on the dev set did not improve for 5 consecutive times. During decoding, we averaged the best 5 or 10 models in the dev set in all tasks. For single models, we set the beam size and length penalty to 5 and 1.0, respectively, while for ensemble systems we used different values adapted from our test sets. The MT and ST models were evaluated using SacreBLEU (Post, 2018), while the ASR models were evaluated using WER. All the models were trained on 8 NVIDIA 3090 or 8 TITAN RTX GPUs.

4.2 ASR

Table 4 presents the ASR results. We observed that the deeper model performed better in confronting noise test sets (dev set of MuST-C and test-other), but it also overfitted in some test sets (tst-COMMON and test-clean). We did not calculate the WER of Transformer-HuBERT because it was only pre-trained as a feature extractor and was not fine-tuned for speech recognition tasks.

4.3 MT and IDA

Table 5 shows the MT and IDA results on the test sets of MuST-C and CoVoST. We found that pre-training on all the MT corpora and fine-tuning on the in-domain corpora can improve performance. Fine-tuning on both MuST-C and CoVoST together is better than only on MuST-C corpus (ODE1 vs. ODE2). It is worth noting that fine-tuning not only improves the performance of in-domain test sets, but also enhances the performance on out-domain test sets, such as the test set of WMT21-news (not included in this paper for simplicity).

We found that both DLCL and ODE models outperformed our baseline, which was a Transformer-Base model with fewer parameters. Additionally,

Model	Pre-train		Fine-tune	
	tst-M	tst-C	tst-M	tst-C
Baseline[◇†]	28.20	50.98	28.96	50.18
-	-	-	26.25	46.27
Baseline[†]	26.99	49.12	28.04	49.49
DLCL1	27.68	50.66	28.62	54.12
ODE1[†]	28.28	51.67	28.56	51.09
ODE2	-	-	29.03	55.28
ODE3	28.17	50.98	29.06	54.41
E^1: ensemble (above four)			**29.61**	**56.20**
DLCL2	29.12	53.95	29.46	55.24
ODE4	29.27	54.31	29.56	55.47
ODE-Deep1	29.39	54.21	29.36	55.47
ODE-Deep2	29.44	54.28	29.47	55.71
E^2: ensemble (above four)			**30.02**	**57.18**

Table 5: BLEU scores on the tst-COMMON (tst-M) and the test set of CoVoST (tst-C). All data are in lower case. Models marked with ◇ indicate that the punctuation of the source text in corpora for pre-training, fine-tuning and testing was kept. The † means that only the MuST-C corpus was used in fine-tuning.

ID	Model	Data	tst-M	tst-C
1	Baseline	M	23.09	-
2	TH-DLCL2	P^2	27.50	41.94
3	CPM-DLCL1	P^1	28.37	44.20
4	CPM-DLCL2	P^1	28.44	45.58
5	CPM-DLCL2	P^2	28.57	45.98
6	CPM-ODE4	P^1	28.72	46.76
7	CPM-ODE4	P^2	29.00	47.15
8	CPD-ODE4	P^1	28.79	47.18
9	CPD-ODE4	P^2	29.01	47.65
10	ensemble (7,9)		29.07	48.67
11	ensemble (2,7,9)		29.11	48.88
12	ensemble (2,7,8,9)		29.16	48.98
13	+adjusted beam/alpha		**29.22**	**49.27**

Table 6: BLEU scores on the tst-COMMON (tst-M) and the test set of CoVoST (tst-C). M refers to the MuST-C corpus, C refers to the CoVoST corpus, and P^i refers to $M\&C\&D^i_{ST_{aug}}$. The models with different parameters are separated by the dotted line.

we demonstrated that although models trained on the corpora with punctuation perform better on test sets including punctuation (28.96 vs. 28.04), they do not perform as well on test sets without punctuation (26.25 vs. 28.04), which is more consistent with the situation of the ASR transcript.

Since each round of iteration in IDA requires retraining multiple MT models, we set the MAX-ITER parameter in IDA to 2 to balance computing resources and model performance. We observed that models trained during the second iteration outperformed those trained during the first iteration. During the second iteration, we found that further increasing the number of parameters resulted in limited improvement (ODE4 vs. ODE-Deep1/2). Additionally, iterative training resulted in a considerable improvement in ensemble systems (from 29.61 to 30.02). Finally, we employed the ensemble systems E^1 and E^2 to generate the pseudo data $D^1_{ST_{aug}}$ and $D^2_{ST_{aug}}$ for ST, respectively.

4.4 ST and Ensemble

Table 6 displays the ST results on the test sets of MuST-C and CoVoST. In contrast to MT, we did not use in-domain fine-tuning, as we found in the pre-experiments that it did not improve performance and may even have caused some damage.

Experiments 1-9 demonstrated that increasing the number of parameters, initializing with better pre-trained models, and training with higher-quality pseudo ST corpora were all effective ways for enhancing the performance of the ST model. These modifications resulted in a significant improvement over the baseline model, which has 32M parameters and was trained solely on the MuST-C dataset.

In the ensemble stage, we aimed to maximize the diversity between models. To achieve this, we selected models with different input representations, architectures, and training corpora. Finally, by expanding the beam size and adjusting the length penalty (alpha), we achieved a BLEU score of 29.22 on tst-COMMON sets, which represents a 0.12 BLEU improvement over our optimal result from the previous year, despite using less MT training data than last year (Agarwal et al., 2023).

5 Conclusion

This paper presented our submission to the IWSLT23 English-to-Chinese offline speech translation task. Our system aimed to find the optimal ensemble system under the "constrained" training condition. To achieve this goal, we explored different input representations, model architectures, and proposed an IDA method to utilize all available texts to improve the MT systems and generate multiple pseudo ST data. Our final system achieved a BLEU score of 29.22 on the MuST-C En-Zh tst-COMMON set, and the results on the IWSLT 23 test sets are shown in Table 7.

System	TED					ACL	
	Comet		BLEU			Comet	BLEU
Ref	2	1	2	1	both		
NiuTrans	0.8376	0.7740	50.0	34.3	57.9	0.7733	47.1

Table 7: Scores on the IWSLT23 test sets.

Acknowledgement

The authors would like to thank anonymous reviewers for their insightful comments. This work was supported in part by the National Science Foundation of China (No. 62276056), the National Key R&D Program of China, the China HTRD Center Project (No. 2020AAA0107904), the Natural Science Foundation of Liaoning Province of China (2022-KF-16-01), the Yunnan Provincial Major Science and Technology Special Plan Projects (No. 202103AA080015), the Fundamental Research Funds for the Central Universities (Nos. N2216016, N2216001, and N2216002), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No. B16009).

References

Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Yannick Estève, Kevin Duh, Marcello Federico, Souhir Gahbiche, Barry Haddow, Benjamin Hsu, Phu Mon Htut, Hirofumi Inaguma, Dávid Javorský, John Judge, Yasumasa Kano, Tom Ko, Rishu Kumar, Pengwei Li, Xutail Ma, Prashant Mathur, Evgeny Matusov, Paul McNamee, John P. McCrae, Kenton Murray, Maria Nadejde, Satoshi Nakamura, Matteo Negri, Ha Nguyen, Jan Niehues, Xing Niu, Atul Ojha Kr., John E. Ortega, Proyag Pal, Juan Pino, Lonneke van der Plas, Peter Polák, Elijah Rippeth, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Yun Tang, Brian Thompson, Kevin Tran, Marco Turchi, Alex Waibel, Mingxuan Wang, Shinji Watanabe, and Rodolfo Zevallos. 2023. Findings of the IWSLT 2023 Evaluation Campaign. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*. Association for Computational Linguistics.

Parnia Bahar, Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2019. On using specaugment for end-to-end speech translation. In *Proceedings of the 16th International Conference on Spoken Language Translation, IWSLT 2019, Hong Kong, November 2-3, 2019*. 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.

Marco Gaido, Mattia Antonino Di Gangi, Matteo Negri, and Marco Turchi. 2020. End-to-end speech-translation with knowledge distillation: Fbk@iwslt2020. In *Proceedings of the 17th International Conference on Spoken Language Translation, IWSLT 2020, Online, July 9 - 10, 2020*, pages 80–88. Association for Computational Linguistics.

Marco Gaido, Sara Papi, Dennis Fucci, Giuseppe Fiameni, Matteo Negri, and Marco Turchi. 2022. Efficient yet competitive speech translation: Fbk@iwslt2022. In *Proceedings of the 19th International Conference on Spoken Language Translation, IWSLT@ACL 2022, Dublin, Ireland (in-person and online), May 26-27, 2022*, pages 177–189. Association for Computational Linguistics.

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*, volume 148 of *ACM International Conference Proceeding Series*, pages 369–376. ACM.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented transformer for speech recognition. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 5036–5040. ISCA.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE ACM Trans. Audio Speech Lang. Process.*, 29:3451–3460.

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, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics.

Jaesong Lee and Shinji Watanabe. 2021. Intermediate loss regularization for ctc-based speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 6224–6228. IEEE.

Bei Li, Quan Du, Tao Zhou, Yi Jing, Shuhan Zhou, Xin Zeng, Tong Xiao, Jingbo Zhu, Xuebo Liu, and Min

Zhang. 2022. ODE transformer: An ordinary differential equation-inspired model for sequence generation. 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 8335–8351. Association for Computational Linguistics.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the System Demonstrations, July 10, 2012, Jeju Island, Korea*, pages 25–30. The Association for Computer Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. 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, Demonstrations*, pages 48–53. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 186–191. Association for Computational Linguistics.

Weiqiao Shan, Zhiquan Cao, Yuchen Han, Siming Wu, Yimin Hu, Jie Wang, Yi Zhang, Hou Baoyu, Hang Cao, Chenghao Gao, Xiaowen Liu, Tong Xiao, Anxiang Ma, and Jingbo Zhu. 2022. The niutrans machine translation systems for WMT22. In *Proceedings of the Seventh Conference on Machine Translation, WMT 2022, Abu Dhabi, United Arab Emirates (Hybrid), December 7-8, 2022*, pages 366–374. Association for Computational Linguistics.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 464–468. Association for Computational Linguistics.

Ioannis Tsiamas, Gerard I. Gállego, José A. R. Fonollosa, and Marta R. Costa-jussà. 2022. SHAS: approaching optimal segmentation for end-to-end speech translation. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 106–110. ISCA.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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 Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1810–1822. Association for Computational Linguistics.

Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly translate foreign speech. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pages 2625–2629. ISCA.

Tong Xiao, Jingbo Zhu, Hao Zhang, and Qiang Li. 2012. Niutrans: An open source toolkit for phrase-based and syntax-based machine translation. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the System Demonstrations, July 10, 2012, Jeju Island, Korea*, pages 19–24. The Association for Computer 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2619–2630. Association for Computational Linguistics.

Chen Xu, Xiaoqian Liu, Xiaowen Liu, Laohu 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, Bangkok, Thailand (online), August 5-6, 2021*, pages 92–99. Association for Computational Linguistics.

Chen Xu, Yuhao Zhang, Chengbo Jiao, Xiaoqian Liu, Chi Hu, Xin Zeng, Tong Xiao, Anxiang Ma, Huizhen Wang, and Jingbo Zhu. 2023. Bridging the granularity gap for acoustic modeling. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics.

Yuhao Zhang, Canan Huang, Chen Xu, Xiaoqian Liu, Bei Li, Anxiang Ma, Tong Xiao, and Jingbo Zhu. 2022a. The niutrans's submission to the IWSLT22 english-to-chinese offline speech translation task. In *Proceedings of the 19th International Conference on Spoken Language Translation, IWSLT@ACL 2022, Dublin, Ireland (in-person and online), May 26-27, 2022*, pages 232–238. 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, WMT@EMNLP 2020, Online, November 19-20, 2020*, pages 338–345. Association for Computational Linguistics.

Yuhao Zhang, Chen Xu, Bojie Hu, Chunliang Zhang, Tong Xiao, and Jingbo Zhu. 2022b. Improving end-to-end speech translation by leveraging auxiliary speech and text data. *CoRR*, abs/2212.01778.