The USTC's Offline Speech Translation Systems for IWSLT 2023

Xinyuan Zhou[2], **Jianwei Cui**[1], **Zhongyi Ye**[2], **Yichi Wang**[1],
Luzhen Xu[1], **Hanyi Zhang**[2], **Weitai Zhang**[1,2], **Lirong Dai**[1]
[1]University of Science and Technology of China, Hefei, China
[2]iFlytek Research, Hefei, China
{jwcui,wangyichi,lzxu,zwt2021}@mail.ustc.edu.cn
lrdai@ustc.edu.cn
{xyzhou15,zyye7,hyzhang56}@iflytek.com

Abstract

This paper describes the submissions of the research group USTC-NELSLIP to the 2023 IWSLT Offline Speech Translation competition, which involves translating spoken English into written Chinese. We utilize both cascaded models and end-to-end models for this task. To improve the performance of the cascaded models, we introduce Whisper to reduce errors in the intermediate source language text, achieving a significant improvement in ASR recognition performance. For end-to-end models, we propose Stacked Acoustic-and-Textual Encoding extension (SATE-ex), which feeds the output of the acoustic decoder into the textual decoder for information fusion and to prevent error propagation. Additionally, we improve the performance of the end-to-end system in translating speech by combining the SATE-ex model with the encoder-decoder model through ensembling.

1 Introduction

This paper describes the submission for the IWSLT 2023 Offline Speech Translation task (Agarwal et al., 2023) by National Engineering Laboratory for Speech and Language Information Processing (NELSLIP) at the University of Science and Technology of China.

Speech translation (ST) solutions include cascaded and end-to-end approaches. The cascaded approach combines Automatic Speech Recognition (ASR) and Machine Translation (MT) systems. The ASR system recognizes the source speech as intermediate text in the source language, and the MT system translates the intermediate text into text in the target language. While the end-to-end approach directly translates the source speech into text in target language, without using source language text as an intermediate representation. Compared with cascaded approaches, the end-to-end paradigm can overcome higher architectural complexity and error propagation (Duong et al., 2016). The Stacked Acoustic-and-Textual Encoding (SATE) (Xu et al., 2021) method combines the acoustic and textual encoders using an adapter module to approach the performance levels of cascaded solutions. Furthermore, ST can be improved using large-scale and cross-modal pretraining methods (Radford et al., 2022; Zhang et al., 2022b) such as Whisper (Radford et al., 2022), which leverages large-scale weak supervision, and SpeechUT (Zhang et al., 2022b), which optimizes the alignment of speech and text modalities by hidden units.

In this study, we employ a cascaded approach wherein the ASR system is built using the pretrained Whisper (Radford et al., 2022) to ensure the recognition performance of speech to source language text. Furthermore, the MT systems in the cascaded setup are created using diverse techniques like back translation (Sennrich et al., 2016a), self-training (Kim and Rush, 2016; Liu et al., 2019), domain adaptation and model ensemble.

In end-to-end condition, we implement two types of architectures, including encoder-decoder (Le et al., 2021) and Stacked Acoustic-and-Textual Encoding extension (SATE-ex). For the encoder-decoder, we use the corresponding components of ASR models to initialize the encoder, and the corresponding components of MT models to initialize the decoder. For SATE-ex, we utilize the textual decoder to receive the output features of the acoustic decoder to assist in generating the target language text, achieving information complementarity of different ASR decoding hidden states, and preventing intermediate error propagation. Additionally, we employ adaptation training, along with the adaptation module and multi-teacher knowledge distillation of Stacked Acoustic-and-Textual Encoding (SATE) (Xu et al., 2021) to bridge the gap between pre-training and fine-tuning. Our approach included the utilization of augmentation strategies commonly used in cascaded systems, like speech synthesis (Casanova et al., 2022) and gen-

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 194–201

Corpus	Duration (h)	Sample Scale
Librispeech	960	1
Europarl	161	1
MuST-C (v1)	399	3
MuST-C (v2)	449	3
TED-LIUM3	452	3
CoVoST2	1985	1
VoxPopuli	1270	1

Table 1: The used speech recognition datasets.

Data	Duration (h)
Raw data	8276
+ concat	16000
+ oversampling	32000
+ TTS	56000

Table 2: Augmented training data for ASR.

erating as much semi-supervised data as possible to enhance the model's performance. Furthermore, we try to achieve further performance optimization with ensemble of cascaded and end-to-end models.

2 Data Preprocessing

2.1 Speech Recognition

The speech recognition datasets utilized in our experiments are listed in Table 1, including Librispeech, MuST-C (v1, v2), TED Lium3, Europarl, VoxPopuli, and CoVoST. We first extracted 40-dimensional log-mel filter bank features computed with a 25ms window size and a 10ms window shift. And then, a baseline ASR model, which is used to filter training samples with WER > 40%, is trained. Moreover, to generate sufficient speech recognition corpora, we applied speed perturbation and oversampling techniques on the TED/MuST-C corpus (Liu et al., 2021). As a result, we generated nearly 8k hours of speech data.

To improve our training data, we applied two more data augmentation techniques. Firstly, we combined adjacent voices to produce longer training utterances. Secondly, we trained a model using Glow-TTS (Casanova et al., 2021) on MuST-C datasets and generated 24,000 hours of audio features by using sentences from EN→DE text translation corpora. The resulting training data for ASR is summarized in Table 2.

	Parallel	Monolingual
EN-ZH	50M	50M

Table 3: Training data for text MT.

2.2 Text Translation

We participate in translating English to Chinese. Both the bilingual data as well as the monolingual data are used for training. To ensure optimal training data quality, we apply several filters including language identification. We remove sentences longer than 250 tokens and those with a source/target length ratio exceeding 3. Additionally, we train a baseline machine translation model to filter out sentences with poor translation quality.

To tokenize the text, we utilize LTP4.0[1] (Wanxiang et al., 2020) for Chinese and Moses for English. The subwords are generated via Byte Pair Encoding (BPE) (Sennrich et al., 2016b) with 30,000 merge operations for each language direction. Table 3 summarizes the detailed statistics on the parallel and monolingual data used for training our systems.

EN→ZH For EN→ZH task, we utilize nearly 50 million sentence pairs collected from CCMT Corpus, News Commentary, ParaCrawl, Wiki Titles, UN Parallel Corpus, WikiMatrix, Wikititles, MuST-C, and CoVoST2, to train our MT models. In addition, we randomly extract 50 million monolingual Chinese sentences from News crawl and Common Crawl for back-translation purposes to augment our training data.

2.3 Speech Translation

Table 4 outlines the speech translation datasets used in our experiments. MuST-C and CoVoST2 are available for speech translation.

To augment our data, we implemented two additional methods. Firstly, we utilized a text translation model to generate the corresponding target language text from the transcriptions of the speech recognition datasets. The generated text was then added to our speech translation dataset along with its corresponding speech, referred to as KD Corpus in Table 4. This process is similar to sentence knowledge distillation. Secondly, we applied the trained Glow-TTS model to produce audio features from randomly selected sentence pairs in EN→ZH text translation corpora. The resulting filter bank features and their corresponding target language

[1]https://github.com/HIT-SCIR/ltp

	Corpus	Duration (h)	Sample Scale
EN-ZH	MuST-C	593	2
	CovoST2	1092	2
	KD	16000	2
	TTS	27000	1

Table 4: Speech Translation Corpora.

text are utilized to enhance our speech translation dataset, referred to as TTS Corpus in Table 4.

3 Cascaded Speech Translation

3.1 Automatic Speech Recognition

We implement ASR model in cascaded condition via Supervised Hybrid Audio Segmentation (SHAS) and Whisper.

Supervised Hybrid Audio Segmentation. Supervised Hybrid Audio Segmentation (SHAS) (Tsiamas et al., 2022) is used to split long audio into short segments with quality comparable to manual segmentation. Hence, we use SHAS as a Voice Activity Detection (VAD) in the ASR system, as well as a speech segmentation tool in the Speech Translation system. This way, the output of the ASR system can be directly fed into the text translation component.

Whisper. We incorporated the pre-trained Whisper (Radford et al., 2022) as the ASR model of the cascaded system to reduce errors in the intermediate source language text.

Whisper scales weakly supervised speech-to-text tasks to 680,000 hours of labeled audio data and expands the pre-training scope from English-only speech recognition to multilingual and multitask. In comparison with the previous unsupervised pre-training approach (Baevski et al., 2020), Whisper not only improves the quality of the audio encoder, but also trains a pre-trained decoder with high equivalency, enhancing usefulness and robustness. Results demonstrate that the pre-trained Whisper model can be well transferred to different or even zero-shot datasets without any dataset-specific fine-tuning.

We used the large version of the pre-trained whisper model, which contains 32 layers and a total of 1550M parameters.

3.2 Neural Machine Translation

We adopted the same strategy as last year's (Zhang et al., 2022a) and built machine translation models based on the Transformer (Vaswani et al., 2017) implemented in the Fairseq (Ott et al., 2019) toolkit. Each single model was executed on 16 NVIDIA V100 GPUs. Our experiments utilized several crucial technologies including Back Translation, Sentence-level Knowledge Distillation, Domain Adaptation, Robust MT Training, and Ensembling.

Back Translation. The utilization of Back-Translation (Sennrich et al., 2016a) is a proficient technique for enhancing translation accuracy. This method generates synthetic sentence pairs by translating target-side monolingual data. It has gained significant popularity in both academic research and commercial applications. We train NMT models with bilingual data, and translate Chinese sentences to English.

Knowledge Distillation. Sentence-level Knowledge Distillation (Kim and Rush, 2016), also known as Self-training, is an effective method for enhancing performance. We expand our training dataset by leveraging a trained NMT model to translate English sentences into Chinese. This approach has proven to be highly beneficial in improving model accuracy.

Domain Adapatation. Due to the critical importance of high-quality, domain-specific translation (Saunders, 2022), we fine-tune the NMT model by using a mix of in-domain data (such as MuST-C, TED-LIUM3, etc.) and out-of-domain data. Additionally, the labelled English sentences from the speech recognition training data is also utilized as augmented in-domain self-training data by translating them.

We adopt a Denoise-based approach (Wang et al., 2018) to assess and select data for domain-specific MT and use it to denoise NMT training. The technique of denoising addresses data quality issues and reduces the adverse effects of noise on MT training, particularly NMT training.

Robust MT Training. To enhance the robustness of the MT model to ASR errors in cascaded ST, the ASR output adaptive training approach (Zhang et al., 2022a) is introduced. The English transcripts of all speech translation datasets are inputted into a trained ASR model to generate text in source side, which is then paired with the transcription text in target side. We improve the robustness of the MT model through three methods: 1) fine-tuning the MT model with synthetic data; 2) incorporating KL loss during fine-tuning to prevent over-fitting; and 3) distilling the model using clean source text and

ASR output.

Ensemble. For each target language, we trained 4 variants based on the large Transformer configuration, and the final model is an ensemble of these 4 models.

- E15D6-v1: 15 layers for the encoder and 6 layers for the docoder. The embedding size is 1024. FFN size is 8192 and attention head is 16. All available corpora including bilingual, BT and FT are used.
- E15D6-v2: 15 layers for the encoder, 10% training data are randomly dropped.
- E18D6: 18 layers for the encoder and 10-30% training data with low machine translation scores are dropped.
- Macaron: A version with macaron architecture (Lu et al., 2019) based on data of E18D6. 36 layers for the encoder and FFN size is 2048.

3.3 End-to-End Speech Translation

In the end-to-end condition, we ensemble the encoder-decoder and the Stacked Acoustic-and-Textual Encoding extension (SATE-ex) models described in Section 3.4.

Encoder-Decoder. The encoder-decoder-based end-to-end ST model processes the speech in the source language by its encoder and generates text in the target language by its decoder. The encoder and decoder are initialized using the corresponding parts of the cascade ASR and MT models. As regards model architecture, we investigate 4 variants in end-to-end ST.

- VGG-C: The encoder of VGG-C is initialized by the ASR VGG-Conformer architecture, which consists of 2 layers of VGG and 12 layers of Conformer. And the ASR VGG-Conformer is trained using the data in Section 2.1. The decoder of VGG-C is 6 layers of Transformer with embedding size of 1024, attention head of 16 and FFN size of 8192.
- VGG-C-init: The encoder is VGG-Conformer, initialized by ASR VGG-Conformer architecture. The decoder is 6 layers of Transformer, initialized by NMT E15D6-v2 variant.
- VGG-T: The encoder of VGG-T is initialized by the ASR VGG-Transformer architecture, which consists of 2 layers of VGG and 16 layers of Transformer. The decoder of VGG-T is 6 layers of Transformer with embedding size of 1024, attention head of 16 and FFN size of 8192.
- VGG-T-init: The VGG-Transformer encoder is initialized by the ASR VGG-Transformer architecture. The decoder is 6 layers of Transformer, initialized by NMT E15D6-v2 variant.

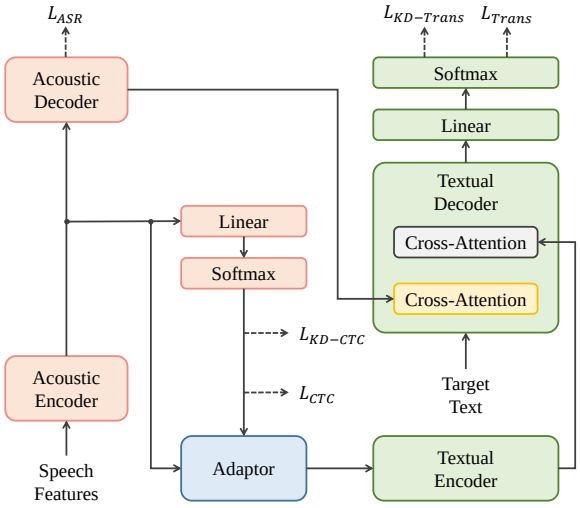

Figure 1: The architecture of Stacked Acoustic-and-Textual Encoding extension (SATE-ex).

3.4 Stacked Acoustic-and-Textual Encoding Extension

To further improve the performance of end-to-end ST, we propose Stacked Acoustic-and-Textual Encoding extension (SATE-ex) based on SATE (Xu et al., 2021).

SATE. The MT encoder captures the long-distance dependency structure, while ASR encoder focuses on local dependencies in the input sequence. Thus, the encoder-decoder model initialized with the ASR encoder and the MT decoder may have inconsistent on intermediate representations.

SATE stacks two encoders, an acoustic encoder and a textual encoder. The acoustic encoder processes the acoustic input, while the textual encoder generates global attention representations for translation. Moreover, an adapter is designed after the acoustic encoder, which maps the acoustic representation to the latent space of the textual encoder while retaining acoustic information. By doing so, SATE can maintain consistency in representation across different pre-trained components. Besides, the multi-teacher knowledge distillation has been

developed to preserve pre-training knowledge during fine-tuning (Hinton et al., 2015).

SATE-ex. Figure 1 shows the SATE-ex architecture, comprising the acoustic encoder, acoustic decoder, textual encoder, and textual decoder components. Theses components are initialized with their corresponding components in cascade ASR and MT models. Notably, the textual decoder in SATE-ex has a Cross-Attention module (highlighted in yellow) that processes the acoustic decoder's output. By doing so, this approach fuses the last layer decoding hidden states of the ASR decoder into the textual decoder, alongside Connectionist Temporal Classification (CTC) decoding hidden states of ASR that are injected through adaptor and textual encoder. Similar to (Zhang et al., 2020), this idea facilitates to fuse and complement different decoding strategies, which can improve inner recognition accuracy, reduce the propagation of intermediate representation errors, and thereby enhance translation performance.

The loss function of SATE-ex, similar to SATE (Xu et al., 2021), computes CTC loss L_{CTC}, ASR loss L_{ASR}, and translation loss L_{Trans}. Additionally, the losses L_{KD-CTC} and $L_{KD-Trans}$ of multi-teacher knowledge distillation are used to preserve pre-trained knowledge during fine-tuning.

Adaptation Training. To further eliminate the intermediate representation mismatch in pre-trained ASR and MT, before end-to-end training, we adopt adaptation training to fine-tune the MT part of SATE-ex (including the textual encoder and textual decoder). Specifically, we first generate greedy CTC decoding without removing duplicates and blanks through the acoustic encoder. Then, we pair these CTC decoding with text in target language to fine-tune the textual encoder and textual decoder. Please note that the textual decoder here does not contain the Cross-Attention module (highlighted in yellow) in Figure 1.

4 Experiments

Our experimental results are presented in Table 5 and Table 6. All experiments are performed using the Fairseq (Ott et al., 2019) toolkit. We report case-sensitive SacreBLEU scores (Post, 2018) for speech translation. The performance of the systems is evaluated on MuST-C-v2 tst-COMMON (tst-COM) and Development set (Dev). Additionally, we set two values for the parameters of SHAS $(min, max, threshold)$, namely $(1, 18, 0.5)$ and $(5, 54, 0.1)$. We also provide the results of MT as reference (System #1-5).

System	tst2018	tst2019	tst2020	tst2022	tst-COM
ASR*	95.59	97.55	95.71	96.67	98.04
Whisper	95.75	98.34	97.17	97.86	97.01

Table 5: The recognition accuracy of the ASR fusion model and pre-trained Whisper. ASR* indicates the ASR fusion model.

4.1 Automatic Speech Recognition

We evaluate the recognition performance of ASR fusion model and pre-trained Whisper. The ASR fusion model comprises three model structures, each trained with and without Text-to-Speech (TTS) data, resulting in a total of six ASR models. These models are fused to obtain the final ASR* model. The three ASR structures are presented below.

- VGG-Conformer: 2 layers of VGG and 12 layers of Conformer in encoder, 6 layers of Transformer in decoder.
- VGG-Transformer: 2 layers of VGG and 16 layers of Transformer in encoder, 6 layers of Transformer in decoder.
- GateCNN-Transformer: 6 layers of GateCNN and 12 layers of Conformer in encoder, 6 layers of Transformer in decoder.

The recognition results of the ASR fusion model and pre-trained Whisper are presented in Table 5. The results indicate that Whisper has a superior recognition performance compared to the ASR fusion model, with an average improvement of 0.51%. However, the ASR fusion model outperforms Whisper slightly on the tst-COM dataset, which could be due to the ASR fusion model upsampling, making its data distribution closer to tst-COM.

4.2 Cascaded Systems

We construct two cascaded systems, one consisting of six-model fusion ASR and six-model fusion MT (System #6), and the other consisting of Whisper and six-model fusion MT (System #7).

For ASR in System #6, we employ the ASR fusion model described in Section 4.1. For MT in System #6, we train the four MT models described in Section 3.2. E18D6 and Macaron are both saved with two different checkpoints, resulting in six MT models that are fused to obtain MT*.

#	System	Official Segment		SHAS (1, 18, 0.5)		SHAS (5, 54, 0.1)	
		Dev	tst-COM	Dev	tst-COM	Dev	tst-COM
	MT						
1	E15D6-v1	27.23	30.19	-	-	-	-
2	E15D6-v2	27.14	29.95	-	-	-	-
3	E18D6	27.53	30.48	-	-	-	-
4	Macaron	27.48	30.71	-	-	-	-
5	ensemble (1-4)	27.81	31.03	-	-	-	-
	Cascaded						
6	ASR*+MT*	26.40	29.83	26.05	29.69	26.45	29.62
7	Whisper+MT*	26.72	29.42	27.00	29.55	26.82	29.03
	End-to-End						
8	SATE-ex-T (w/ TTS)	24.78	28.17	24.43	27.43	23.30	26.49
9	SATE-ex-T (w/o TTS)	25.27	28.00	25.19	27.81	24.37	27.39
10	SATE-ex-M (w/ TTS)	24.52	28.18	23.61	26.62	22.08	24.67
11	SATE-ex-M (w/o TTS)	24.18	27.26	23.96	27.51	20.91	25.66
12	VGG-C-init	24.62	28.74	24.61	28.50	24.12	28.06
13	VGG-T-init	24.59	28.28	24.51	27.84	23.89	27.59
14	VGG-C	24.75	28.68	24.70	28.35	24.29	27.65
15	VGG-T	24.72	28.42	24.60	27.93	24.09	27.77
16	ensemble (8-11)	25.85	29.00	25.50	28.45	24.22	27.54
17	ensemble (12-15)	25.53	28.86	25.54	28.68	25.36	28.68
18	ensemble (8-15)	26.42	29.29	26.22	29.11	25.92	28.92
	Ensemble of cascaded and e2e						
19	ensemble (6, 18)	26.85	29.46	26.65	29.19	26.28	29.41
20	ensemble (7, 18)	27.09	29.53	26.82	29.35	26.62	29.45

Table 6: The BLEU scores of machine translation (MT), cascaded, end-to-end, and ensemble systems. * indicates fusion models. The parameter of SHAS is $(min, max, threshold)$.

System #7 uses the large version of Whisper[3] as ASR, while the MT* is consistent with System #6. As shown, on Dev set, using Whisper to reduce errors in the source language text has improved the performance of ST. However, on tst-COM, the cascade model with ASR* performs better, presumably due to the closer match between the data distribution of ASR* and that of tst-COM.

4.3 End-to-End Systems

In the end-to-end setting, we adopt the encoder-decoder and SATE-ex architectures. Systems #12-15 are built based on the encoder-decoder, with specific parameters referred to Section 3.3. Systems #8-11 adopt the SATE-ex architecture. SATE-ex-T uses the VGG-Conformer ASR model in Section 4.2 to initialize the acoustic encoder and decoder, and the E18D6 MT model in Section 3.2 to initialize the textual encoder and decoder. SATE-ex-M uses the Macaron MT model in Section 3.2 to initialize the textual encoder and decoder.

It can be seen that the results of ensemble SATE-ex (System #16) outperform those of ensemble encoder-decoder (System #17). However, the performance of a single SATE-ex model is slightly worse than that of a single encoder-decoder model, which we attribute to the lack of fine-tuning for the single SATE-ex model. In future work, we will discuss SATE-ex in detail.

4.4 Ensemble Systems

We ensemble the two cascade models (Systems #6 and #7) and the end-to-end model (System #18) separately. The results are shown in Systems #19 and #20 in Table 6. It can be seen that the ensemble

[3]https://github.com/openai/whisper

systems achieves excellent performance.

4.5 System Description

Our system is primarily based on the full dataset allowed by IWSLT 2022, supplemented with Whisper large and SHAS for audio segmentation, which is trained on MUSTC. We have trained six ASR models and six MT models based on the IWSLT 2022 training data for model fusion. Additionally, we have trained four end-to-end ST models and four SATE-ex end-to-end ST models for end-to-end model fusion.

For the end-to-end system, we use a fusion of the above-mentioned eight end-to-end models. For the cascaded systems, we build two cascades: one with ASR based on Whisper and the other with ASR based on six-model fusion. The MT side used six-model fusion for both cascades. The submitted systems are based on these two cascades, each combined with the eight-model fusion end-to-end system.

The system structure and SHAS parameter $(min, max, threshold)$ settings of the five submitted systems are shown below.

- Primary Cascade: System #7 with SHAS parameters set to $(5, 54, 0.1)$.
- Contrastive1: System #20 with SHAS parameters set to $(1, 18, 0.5)$.
- Contrastive2: System #19 with SHAS parameters set to $(1, 18, 0.5)$.
- Contrastive3: System #6 with SHAS parameters set to $(5, 54, 0.1)$.
- Primary e2e: System #18 with SHAS parameters set to $(1, 18, 0.5)$.

5 Conclusion

This paper summarizes the results on the IWSLT 2023 Offline Speech Translation task. We employ various model architectures and data augmentation techniques to build speech translation systems in cascaded and end-to-end settings. The experimental results demonstrate the effectiveness of strategies such as pre-trained Whisper models, adaptation training, and the Stacked Acoustic-and-Textual Encoding extension (SATE-ex). In future work, we will further investigate SATE-ex and explore multimodal representation learning in speech translation.

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.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Edresson Casanova, Christopher Shulby, Eren Gölge, Nicolas Michael Müller, Frederico Santos de Oliveira, Arnaldo Candido Jr., Anderson da Silva Soares, Sandra Maria Aluisio, and Moacir Antonelli Ponti. 2021. SC-GlowTTS: An Efficient Zero-Shot Multi-Speaker Text-To-Speech Model. In *Proc. Interspeech 2021*, pages 3645–3649.

Edresson Casanova, Christopher Shulby, Alexander Korolev, Arnaldo Candido Junior, Anderson da Silva Soares, Sandra Aluísio, and Moacir Antonelli Ponti. 2022. Asr data augmentation using cross-lingual multi-speaker tts and cross-lingual voice conversion. *arXiv preprint arXiv:2204.00618*.

Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn. 2016. An attentional model for speech translation without transcription. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 949–959, San Diego, California. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.

Hang Le, Florentin Barbier, Ha Nguyen, Natalia Tomashenko, Salima Mdhaffar, Souhir Gabiche Gahbiche, Benjamin Lecouteux, Didier Schwab, and Yannick Estève. 2021. ON-TRAC' systems for the IWSLT 2021 low-resource speech translation and multilingual speech translation shared tasks. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 169–174, Bangkok, Thailand (online). Association for Computational Linguistics.

Dan Liu, Mengge Du, Xiaoxi Li, Yuchen Hu, and Lirong Dai. 2021. The USTC-NELSLIP systems for simultaneous speech translation task at IWSLT 2021. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 30–38, Bangkok, Thailand (online). Association for Computational Linguistics.

Yuchen Liu, Hao Xiong, Jiajun Zhang, Zhongjun He, Hua Wu, Haifeng Wang, and Chengqing Zong. 2019. End-to-end speech translation with knowledge distillation. *Proc. Interspeech 2019*, pages 1128–1132.

Yiping Lu, Zhuohan Li, Di He, Zhiqing Sun, Bin Dong, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Understanding and improving transformer from a multi-particle dynamic system point of view. *arXiv preprint arXiv:1906.02762*.

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 (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. 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*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

Danielle Saunders. 2022. Domain adaptation and multi-domain adaptation for neural machine translation: A survey. *Journal of Artificial Intelligence Research*, 75:351–424.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ioannis Tsiamas, Gerard I Gállego, José AR Fonollosa, and Marta R Costa-jussà. 2022. Shas: Approaching optimal segmentation for end-to-end speech translation. *arXiv preprint arXiv:2202.04774*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wei Wang, Taro Watanabe, Macduff Hughes, Tetsuji Nakagawa, and Ciprian Chelba. 2018. Denoising neural machine translation training with trusted data and online data selection. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 133–143, Brussels, Belgium. Association for Computational Linguistics.

Che Wanxiang, Feng Yunlong, Qin Libo, and Liu Ting. 2020. N-ltp: A open-source neural chinese language technology platform with pretrained models. *arXiv preprint*.

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 (Volume 1: Long Papers)*, pages 2619–2630.

Binbin Zhang, Di Wu, Zhuoyuan Yao, Xiong Wang, Fan Yu, Chao Yang, Liyong Guo, Yaguang Hu, Lei Xie, and Xin Lei. 2020. Unified streaming and non-streaming two-pass end-to-end model for speech recognition. *arXiv preprint arXiv:2012.05481*.

Weitai Zhang, Zhongyi Ye, Haitao Tang, Xiaoxi Li, Xinyuan Zhou, Jing Yang, Jianwei Cui, Pan Deng, Mohan Shi, Yifan Song, et al. 2022a. The ustc-nelslip offline speech translation systems for iwslt 2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 198–207.

Ziqiang Zhang, Long Zhou, Junyi Ao, Shujie Liu, Lirong Dai, Jinyu Li, and Furu Wei. 2022b. Speechut: Bridging speech and text with hidden-unit for encoder-decoder based speech-text pre-training. *arXiv preprint arXiv:2210.03730*.