Kyoto Speech-to-Speech Translation System for IWSLT 2023

Zhengdong Yang[1] **Shuichiro Shimizu**[1] **Zhou Wangjin**[1] **Sheng Li**[2] **Chenhui Chu**[1]
Kyoto University[1] National Institute of Information and Communications Technology[2]
{zd-yang, sshimizu, chu}@nlp.ist.i.kyoto-u.ac.jp
zhou@sap.ist.i.kyoto-u.ac.jp
sheng.li@nict.go.jp

Abstract

This paper describes the Kyoto speech-to-speech translation system for IWSLT 2023. Our system is a combination of speech-to-text translation and text-to-speech synthesis. For the speech-to-text translation model, we used the dual-decoder Transformer model. For the text-to-speech synthesis model, we took a cascade approach of an acoustic model and a vocoder.

1 Introduction

This paper describes the Kyoto speech-to-speech translation system for IWSLT 2023 (Agarwal et al., 2023). Our system is a combination of speech-to-text translation and text-to-speech synthesis. For speech-to-text translation model, we used dual-decoder Transformer model following Le et al. (2020). For text-to-speech synthesis model, we took cascade approach of an acoustic model and a vocoder. We used FastSpeech 2 (Ren et al., 2021) as the acoustic model and HiFi-GAN (Kong et al., 2020) as the vocoder.

2 System Description

The speech-to-speech translation system is a combination of speech-to-text translation and text-to-speech synthesis.

2.1 Speech-to-Text Translation

We adopt the end-to-end speech-to-text translation architecture. The speech-to-text translation model is based on dual-decoder Transfomer (Le et al., 2020).

As shown in Figure 1, the model is a Transformer-based model, comprising two decoders - one for speech-to-text translation (ST) and the other for automatic speech recognition (ASR). The task of ASR and ST can be defined as follows:

- For ASR, the input sequence $\boldsymbol{s} = [s_1, ..., s_{T_s}]$ is a sequence of speech features. The output sequence $\boldsymbol{x} = [x_1, ..., x_{T_x}]$ is the corresponding transcription, where T_x indicates the length of the transcription.
- For ST, the input sequence $\boldsymbol{s} = [s_1, ..., s_{T_s}]$ is the same with ASR and the output sequence $\boldsymbol{y} = [y_1, ..., y_{T_y}]$ is the corresponding translation in target language, where T_y indicates the length of the translation.

The model performs the multi-task learning of ASR and ST and the output distributions can be written as

$$D_{asr\text{-}st} = p(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{s}) = \prod_{t=0}^{max(T_x,T_y)} p(x_t, y_t|\boldsymbol{x}_{<t}, \boldsymbol{y}_{<t}, \boldsymbol{s}) \quad (1)$$

The training objective is a weighted sum of cross-entropy losses for both tasks:

$$L_{asr\text{-}st} = \alpha L_{asr} + (1-\alpha)L_{st} \quad (2)$$

Different decoders can exchange information with each other with the interactive attention mechanism, which refers to replacing attention sub-layers in the standard Transformer decoder with interactive attention sub-layers (Liu et al., 2020). In our models, the replaced sub-layers are the encoder-decoder attention sub-layers.

As illustrated in the lower part of Figure 1, an interactive attention sub-layer consists of one main attention sub-layer and a cross-attention sub-layers. The main attention sub-layer is the same as the replaced attention sub-layer. The cross-attention sub-layers receive query $\boldsymbol{Q}$ from the same decoder A and receive key $\boldsymbol{K}$ and value $\boldsymbol{V}$ from another decoder B. We adopt the parallel variation of dual-decoder Transformers where $\boldsymbol{K}$ and $\boldsymbol{V}$ are hidden states from the same layer in decoder B.

The final output is obtained by merging the output of the primary attention sub-layer $\boldsymbol{H}_{main}$ with

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 357–362
July 13-14, 2023 ©2023 Association for Computational Linguistics

the output of the cross attention sub-layer $\boldsymbol{H}_{cross}$. We adopt linear interpolation as the merging function. Therefore the output representations of the interactive attention sub-layers are

$$\boldsymbol{H}_{dual} = \boldsymbol{H}_{main} + \lambda \boldsymbol{H}_{cross} \tag{3}$$

where λ is a learnable parameter.

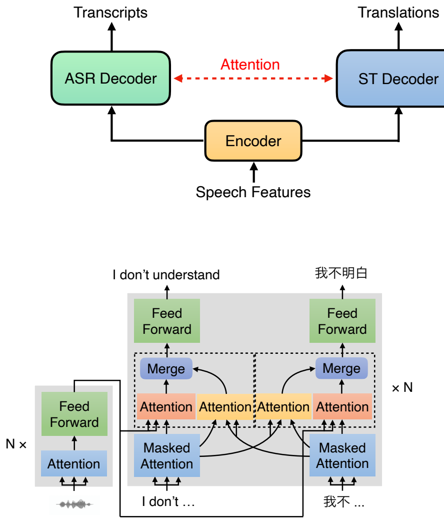

Figure 1: General architecture of dual-decoder Transformer (upper) and interactive attention mechanism (lower). Interactive attention sub-layers are marked with dotted boxes. They merge the outputs of the main attention sub-layers (red boxes) and cross-attention sub-layers (yellow boxes).

2.2 Text-to-Speech Synthesis

We adopted the approach to cascade an acoustic model and a vocoder. We used FastSpeech 2 (Ren et al., 2021) as the acoustic model and HiFi-GAN (Kong et al., 2020) as the vocoder. FastSpeech 2 adopts Transformer-based architecture for the encoder and the Mel-spectrogram decoder, and the variance adapter between them predicts the duration, pitch, and energy of the audio. HiFi-GAN employs generative adversarial networks to generate waveforms from Mel-spectrograms. It is composed of one generator and two discriminators, a multi-period discriminator, and a multi-scale discriminator. We used the PaddleSpeech toolkit (Zhang et al., 2022a) and the pretrained models provided by Zhang et al. (2022a) to generate waveforms.

Dataset	Sentence Embedding Model Used for Filtering	Total Length (Hours)
MuST-C	None	600.2
GigaST	None	9873.2
GigaST	LASER	919.1
GigaST	Sentence Transformers	601.1

Table 1: The size of the datasets and the filtered versions used for training the ST system.

3 Experiments

3.1 Speech-to-Text Translation

3.1.1 Datasets

To train our ST system, we utilized two distinct datasets: MuST-C (Di Gangi et al., 2019) v2 with Chinese translations, and GigaST (Ye et al., 2022) which is the original dataset that was used to construct the GigaS2S dataset provided by the organizers.

Both datasets offer unique advantages. While GigaST is in the same domain as the development and test data, MuST-C is not. In addition, GigaST is considerably larger than MuST-C. However, it is worth noting that the translations in GigaST were generated by a machine translation system and may not be of the same quality as those in MuST-C, which were translated by human. As a result, determining which dataset is more likely to yield better results requires further experimentation.

To shorten the training time and improve performance, we filtered the extremely large GigaST dataset to select utterances with better translation quality. As the translations in GigaST are machine-generated and there are no reference translations available, we evaluated the translation quality using the cosine similarity of sentence embeddings from the source and target sentences. We tested two different models for generating the embeddings: LASER[1] and "paraphrase-xlm-r-multilingual-v1" from Sentence Transformers[2] (simply referred to as "Sentence Transformers" subsequently). The resulting similarity distributions are shown in Figure 2. We selected the top 10% of the data based on similarity scores (data that is on the right-hand side of the red line). Table 1 shows the sizes of MuST-C and GigaST before and after filtering.

[1] https://github.com/facebookresearch/LASER
[2] https://github.com/UKPLab/sentence-transformers/tree/master/examples/training/paraphrases

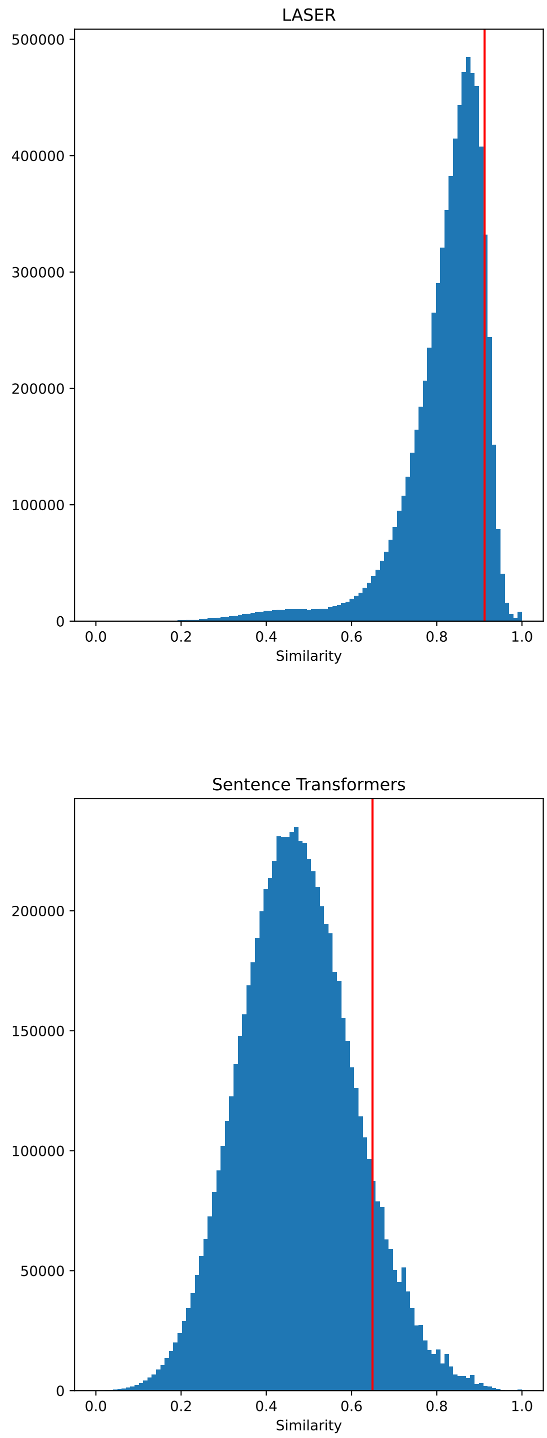

Figure 2: Histograms of cosine similarity between source and target sentence embedding based on LASER and Sentence Transformers. The red line marks the 90th percentile.

3.1.2 Training and Decoding

English sentences were normalized and tokenized using the Moses tokenizer (Koehn et al., 2007), and punctuations were stripped. Chinese sentences were tokenized using jieba.[3] English and Chinese tokens were further split into subwords using the BPE method (Sennrich et al., 2016) with a joint vocabulary of $16,000$ subwords.

We used Kaldi (Ravanelli et al., 2019) to extract 83-dimensional features normalized by the mean and standard deviation computed on the training set. We removed utterances with more than $6,000$ frames or more than 400 characters and used speed perturbation (Inaguma et al., 2020) with factors of 0.9, 1.0, and 1.1 for data augmentation.

Our implementation was based on the ESPnet-ST toolkit (Inaguma et al., 2020). We used the same architecture for all the ST models with a 12-layer encoder and 8-layer decoders. The coefficient α in the loss function (Equation 2) was set to 0.3 in all the experiments. We used the Adam optimizer (Kingma and Ba, 2015) and Noam learning rate schedule (Vaswani et al., 2017) with $25,000$ warm-up steps and a maximum learning rate of $2.5e-3$. We used a batch size of 48 per GPU and trained models on a single machine with 4 Tesla V100 GPUs. The models were trained for 25 epochs. We kept checkpoints after each epoch and averaged the five best models on the development set based on prediction accuracy. For decoding, the beam size was set to 5 for ST and 1 for ASR.

3.1.3 Results

We conducted experiments to investigate the impact of using different datasets for training the system. The results are presented in Table 2. Additionally, we evaluated the performance of the system when using different sentence embedding models for data filtering. Our findings reveal that LASER produces better results compared to Sentence Transformers. Notably, after filtering the data using LASER, the total number of hours of audio is higher compared to that obtained using Sentence Transformers. Given this observation, it might be more appropriate to perform filtering based on the length of the audio rather than the number of utterances.

Our experiments also revealed that training the model with GigaST alone yielded better results compared to using only the MuST-C dataset. Fur-

[3]https://github.com/fxsjy/jieba

Training Data	BLEU
MuST-C	9.71
GigaST (LASER)	**13.96**
GigaST (Sentence Transformers)	11.57
MuST-C → GigaST (LASER)	13.52
GigaST (LASER) → MuST-C	13.30

Table 2: Experimental results on training with different datasets. "→" indicates training with the dataset on the left and use the best checkpoint to initiate the training with the dataset on the right.

thermore, we evaluated an approach in which we trained the model with one dataset and use the best checkpoint to initiate the training with the other dataset. However, we observed that this approach did not yield any improvement compared to training the model with GigaST alone.

Based on these findings, we adopted the translation generated by the ST system trained solely on GigaST filtered based on LASER for our submission.

3.2 Text-to-Speech Synthesis

We used pretrained models provided by Zhang et al. (2022a) trained on the AISHELL-3 dataset (Shi et al., 2021). The PaddleSpeech toolkit provides several models trained with the AISHELL-3 dataset, including FastSpeech 2 and HiFi-GAN. We used the best-performing model combination in terms of MOS reported in (Zhang et al., 2022a). For other configurations, such as grapheme-to-phoneme conversion, we followed Zhang et al. (2022a).

The generated audio files have one channel, a sample width of 16 bit, and a frame rate of $24,000$. Because the predictions of speech-to-text translation sometimes contained English words that were preprocessed to empty strings by the grapheme-to-phoneme conversion, some (less than 1 % of the test set) audio files could not be generated.

4 Conclusion

In this paper, we described our system, which is a combination of speech-to-text translation and text-to-speech synthesis. For speech-to-text translation, we trained the Dual-decoder Transformer model with the GigaST dataset filtered based on the similarity of multilingual sentence embeddings. For the text-to-speech synthesis model, we took a cascade approach of an acoustic model and a vocoder and used a combination of FastSpeech 2 and HiFi-GAN. In the future, we will try to perform multi-level pretraining based on transforming SpeechUT (Zhang et al., 2022b) with phonemes as unit. We will also try to use Encodec-based speech synthesis method similar to VALL-EX (Zhang et al., 2023) to increase the accurate representation of emotions and vocal patterns.

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.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2012–2017, Minneapolis, Minnesota. Association for Computational Linguistics.

Hirofumi Inaguma, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Yalta, Tomoki Hayashi, and Shinji Watanabe. 2020. Espnet-st: All-in-one speech translation toolkit. In *Proceedings of the 58th Annual Meeting of the Association for*

Computational Linguistics: System Demonstrations, ACL 2020, pages 302–311. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*. The Association for Computational Linguistics.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

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

Yuchen Liu, Jiajun Zhang, Hao Xiong, Long Zhou, Zhongjun He, Hua Wu, Haifeng Wang, and Chengqing Zong. 2020. Synchronous speech recognition and speech-to-text translation with interactive decoding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, pages 8417–8424. AAAI Press.

Mirco Ravanelli, Titouan Parcollet, and Yoshua Bengio. 2019. The pytorch-kaldi speech recognition toolkit. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, pages 6465–6469. IEEE.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2021. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech. In *International Conference on Learning Representations*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, Volume 1: Long Papers*. The Association for Computer Linguistics.

Yao Shi, Hui Bu, Xin Xu, Shaoji Zhang, and Ming Li. 2021. AISHELL-3: A Multi-Speaker Mandarin TTS Corpus. In *Proc. Interspeech 2021*, pages 2756–2760.

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*, pages 5998–6008.

Rong Ye, Chengqi Zhao, Tom Ko, Chutong Meng, Tao Wang, Mingxuan Wang, and Jun Cao. 2022. GigaST: A 10,000-hour Pseudo Speech Translation Corpus.

Hui Zhang, Tian Yuan, Junkun Chen, Xintong Li, Renjie Zheng, Yuxin Huang, Xiaojie Chen, Enlei Gong, Zeyu Chen, Xiaoguang Hu, Dianhai Yu, Yanjun Ma, and Liang Huang. 2022a. PaddleSpeech: An easy-to-use all-in-one speech toolkit. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, pages 114–123, Hybrid: Seattle, Washington + Online. Association for Computational Linguistics.

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*.

Ziqiang Zhang, Long Zhou, Chengyi Wang, Sanyuan Chen, Yu Wu, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023. Speak foreign languages with your own voice: Cross-lingual neural codec language modeling. *arXiv preprint arXiv:2303.03926*.