NAIST Simultaneous Speech Translation System for IWSLT 2023

Ryo Fukuda[†] **Yuta Nishikawa**[†] **Yasumasa Kano**[†] **Yuka Ko**[†]
Tomoya Yanagita[†] **Kosuke Doi**[†] **Mana Makinae**[†]
Sakriani Sakti[‡†] **Katsuhito Sudoh**[†] **Satoshi Nakamura**[†]
[†]Nara Institute of Science and Technology, Japan
[‡]Japan Advanced Institute of Science and Technology, Japan
fukuda.ryo.fo3@is.naist.jp

Abstract

This paper describes NAIST's submission to the IWSLT 2023 Simultaneous Speech Translation task: English-to-{German, Japanese, Chinese} speech-to-text translation and English-to-Japanese speech-to-speech translation. Our speech-to-text system uses an end-to-end multilingual speech translation model based on large-scale pre-trained speech and text models. We add Inter-connections into the model to incorporate the outputs from intermediate layers of the pre-trained speech model and augment prefix-to-prefix text data using Bilingual Prefix Alignment to enhance the simultaneity of the offline speech translation model. Our speech-to-speech system employs an incremental text-to-speech module that consists of a Japanese pronunciation estimation model, an acoustic model, and a neural vocoder.

1 Introduction

This paper presents NAIST's simultaneous speech translation (SimulST) systems for the IWSLT 2023 English-to-{German, Japanese, Chinese} speech-to-text track and the English-to-Japanese speech-to-speech track (Agarwal et al., 2023).

Many previous studies on end-to-end SimulST have focused on training methodologies and architectures specialized for the simultaneous scenario. However, such a specialized system setup for SimulST is not trivial and increases the difficulty of the system development and the computational complexity. One recent approach to SimulST systems is to use an offline speech translation (ST) model for prefix-to-prefix translation required in SimulST. In last year's IWSLT Evaluation Campaign (Anastasopoulos et al., 2022), Polák et al. (2022) demonstrated superior results using such multilingual offline ST models. In our last year's systems (Fukuda et al., 2022), we used an offline model fine-tuned for SimulST with data augmentation based on Bilingual Prefix Alignment (Kano et al., 2022).

In this year, we use an end-to-end multilingual offline ST model based on large-scale pre-trained speech and text models for the speech-to-text track, following Polák et al. (2022). We used Hidden-Unit BERT (HuBERT) (Hsu et al., 2021) as the speech encoder fine-tuned using English automatic speech recognition (ASR) data and mBART50 (Tang et al., 2020) as the text decoder fine-tuned using a multilingual machine translation data. We prepare the multilingual ST model in the following steps:

1. Initialize the model with the parameters of HuBERT and mBART50 models and add Inter-connections between the intermediate layer of the speech encoder and the text decoder.
2. Train the model using multilingual ST corpora.
3. Fine-tune the model using bilingual prefix pairs in English-to-{German, Japanese, Chinese} extracted using Bilingual Prefix Alignment.

We use a SimulST policy called *local agreement* (Liu et al., 2020) that finds the longest common prefixes among successive decoding steps. For the English-to-Japanese speech-to-speech track, we developed a cascade of the SimulST above and an incremental text-to-speech module using a pronunciation estimation model, an acoustic model, and a neural vocoder.

2 System Architecture

Figure 1 illustrates an overview of our system architecture. The following subsections explain our methodologies: Inter-connection in 2.1 and Bilingual Prefix Alignment in 2.2, the local agreement in 2.3, and the incremental text-to-speech in 2.4.

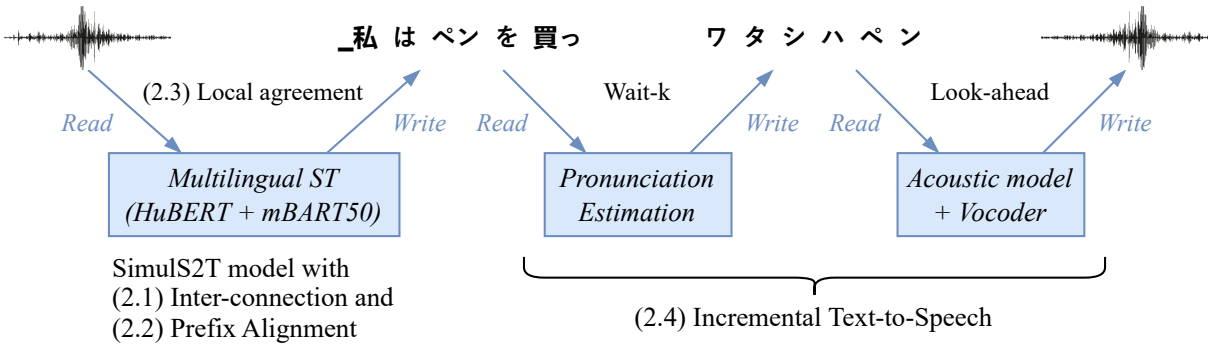


Figure 1: Block diagram of SimulS2S.

2.1 Inter-connection

Intermediate layers of a speech SSL (Self-Supervised Learning) model contain useful information for downstream tasks (Pasad et al., 2021). However, the simple addition of connections from the intermediate layers of the speech encoder to the text decoder does not always work well. We use a weighted integration of the encoder's intermediate layers, called *Inter-connection* (Nishikawa and Nakamura, 2023), where the output tensors from the intermediate layers are aggregated with the weights. The weights are additional learnable parameters optimized through the training. We also apply layer normalization after the weighted aggregation to stabilize the training.

2.2 Prefix Alignment

In simultaneous translation, the model translates a prefix of the entire input to the corresponding output prefix. The prefix translation using a full-sentence model often suffers from so-called *over-translation* (or *hallucination*) due to the lack of training examples in the prefix-to-prefix scenarios. To mitigate this problem, we leverage the training corpus using Bilingual Prefix Alignment (Kano et al., 2022) for data augmentation for prefix-to-prefix pairs to fine-tune the SimulST model.

2.3 Local Agreement

Liu et al. (2020) proposed Local agreement to find a stable prefix translation hypothesis in the prefix-to-prefix translation based on chunk-wise inputs with the fixed length. It verifies the stability of the hypothesis at step t using the hypothesis at step $t+1$ by taking the agreeing prefix (i.e., the longest common prefix) of them. This is based on an idea that the agreeing prefix translation outputs with growing input prefixes should be reliable. Polák et al. (2022) generalized this idea using agreement among the prefixes at n consecutive steps (LA-n) and demonstrated that $n = 2$ works well on SimulST. According to their finding, we use LA-2 as a SimulST policy and adjust the input chunk length (in milliseconds) to control the quality-latency trade-offs.

2.4 Incremental Text-to-Speech

Our English-to-Japanese speech-to-speech simultaneous translation system uses the aforementioned SimulST system with incremental Japanese text-to-speech (TTS). The incremental TTS consists of three modules: a pronunciation estimation, an acoustic model, and a neural vocoder. The pronunciation estimation predicts the pronunciations of SimulST outputs, the acoustic model predicts acoustic features from the pronunciations, and the neural vocoder synthesizes speech from the acoustic features.

We use the wait-k approach (Ma et al., 2019) for the incremental pronunciation estimation, taking a subword sequence as the input and predicting pronunciation symbols in Japanese katakana phonograms and a couple of special characters representing accents as the output. To control the output length, we extend the wait-k policy by allowing the decoder to output an arbitrary length of symbols; The decoder stops its *write* steps when the largest weight of its cross attention goes over the last two tokens in the input prefix. This also works as a lookahead mechanism for pronunciation estimation. We use Tacotron2 (Shen et al., 2018) for the acoustic modeling and Parallel WaveGAN (Yamamoto et al., 2020) as the neural vocoder in the prefix-to-prefix manner (Ma et al., 2020a).

Table 1: Training data measured in hours.

Dataset	En-De	En-Ja	En-Zh
MuST-C v1	408h		
MuST-C v2	436h	526h	545h
Europarl-ST	83h		
CoVoST-2	413h	413h	413h
TED-LIUM	415h		
Total	1,755h	939h	958h

Table 2: Comparison of the removed ratios resulting from data filtering with maximum ratios of 4,800, 4,000, and 3,200.

Filter (max ratio)	Removed Ratio (%)		
	En-De	En-Ja	En-Zh
No filtering	0.0	0.0	0.0
4,800	37.8	59.4	59.7
4,000	53.9	72.5	74.1
3,200	78.0	87.9	89.4

3 System Setup

3.1 Data

We used MuST-C v2.0 (Di Gangi et al., 2019) and CoVoST-2 (Wang et al., 2020) for all language pairs: English-to-German (En-De), English-to-Japanese (En-Ja), and English-to-Chinese (En-Zh). We also used MuST-C v1.0, Europarl-ST (Iranzo-Sánchez et al., 2020), and TED-LIUM (Rousseau et al., 2012) for English-to-German. We included the development and test portions of CoVoST-2 and Europarl-ST in our training data. The overall statistics for these corpora are shown in Table 1. For evaluation, we used the tst-COMMON portion of MuST-C v2.0. All the text data in the corpora were tokenized using a multilingual SentencePiece tokenizer with a vocabulary of 250,000 subwords, distributed with mBART50 pre-trained model.

3.2 Data Filtering

We conducted a data filtering on the prefix translation pairs obtained through the Bilingual Prefix Alignment, following our IWSLT 2022 system (Fukuda et al., 2022). We compared three cut-off ratios of the number of samples in the input speech to the number of tokens in the output: 4,800, 4,000, and 3,200. Table 2 shows the percentage of data that was removed following the application of filters. We also applied the same filtering to the development data.

3.3 Simultaneous Speech-to-Text System

We deveoped an end-to-end speech-to-text model initialized with two pre-trained models for its speech encoder and text decoder. The speech encoder was initialized with HuBERT-Large, which consists of a feature extractor trained on 60 K hours of unlabeled speech data Libri-Light (Kahn et al., 2020) and Transformer encoder layers. The feature extractor has seven convolutional layers with a kernel size of (10, 3, 3, 3, 3, 2, 2), a stride of (5, 2, 2, 2, 2, 2, 2), and 512 channels. The number of the Transformer encoder layers is 24. The text decoder was initialized with the decoder of mBART50 (Tang et al., 2020). The decoder consists of twelve Transformer layers, and an embedding layer and linear projection weights are shared, with a size of 250,000. The size of each Transformer and feed-forward layer is 1,024 and 4,096, respectively, the number of attention heads is 16, the activation function is ReLU, and the layer normalization is applied before the attention operations. The encoder and decoder are also connected via Inter-connection (2.1) and a length adapter (Tsiamas et al., 2022). The length adapter is a 3-layer convolutional network with 1,024 channels, the stride of 2, and the activation function of a Gated Linear Unit (GLU).

Speech input is given as waveforms with a 16-kHz sampling rate, normalized to zero mean and unit variance. During training, each source audio was augmented (Kharitonov et al., 2020) before normalization, with a probability of 0.8. We trained multilingual models on all the data listed in Table 1 with a maximum source length of 400,000 frames and a target length of 1,024 tokens. We applied gradient accumulation and data-parallel computations to achieve a batch size of approximately 32 million tokens. We used Adam with $\beta_1 = 0.99$, $\beta_2 = 0.98$, and a base learning rate of 2.5×10^{-4}. The learning rate was controlled by a tri-stage scheduler with phases of 0.15, 0.15, and 0.70 for warm-up, hold, and decay, respectively, while the initial and final learning rate had a scale of 0.01 compared to base. We used sentence averaging and gradient clipping of 20. We applied a dropout probability of 0.1 and used time masking for 10-length spans with a probability of 0.2, and channel masking for 20-length spans with a probability of 0.1 in the encoder feature extractor's out-

put. The loss was the cross-entropy loss with a label smoothing with 20% probability mass.

The offline SimulST model was fine-tuned, and then checkpoint averaging was performed. In the checkpoint averaging, the model checkpoints were saved every 1,000 training steps, and the averaged parameter values among the five-best models in the loss on the development data were taken for the final model. Subsequently, one epoch of fine-tuning was performed on the training data-only prefix alignment pairs in MuST-C v2. We reduced the learning rate to 2.5×10^{-5} during the fine-tuning using translation pairs obtained using Bilingual Prefix Alignment.

As a SimulST policy, the local agreement with $n = 2$ (LA-2) was used. The chunk size was varied from 200 ms to 1000 ms to adjust the quality-latency trade-off. A beam search of beam size five was used to generate hypotheses for input chunks.

3.4 Simulaneous Speech-to-Speech System

Here, we describe the detailed setup of the incremental TTS. Pronunciation symbols were obtained from the text using Open Jtalk[1]. We used the Balanced Corpus of Contemporary Written Japanese (BCCWJ; Maekawa, 2008) for training the pronunciation estimation model. The training, development, and test data were approximately 1.4 M, 10 K, and 10 K sentences, respectively. We also used the training portion of MuST-C as additional training data. We used an LSTM-based attentional encoder-decoder model for the pronunciation estimation model. Its encoder and decoder were implemented with two-layer uni-directional LSTM, and the cross-attention was based on the dot product. The optimizer was Adam with the learning rate of 1e-3 and hyperparameters of $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batch size was 256 in the number of sentences.

JSUT corpus (Sonobe et al., 2017) was used for training Tacotron2 and Parallel WaveGAN. The numbers of sentences in the training, development, and test data were 7,196, 250, and 250, respectively. Speech is downsampled from 48 kHz to 16 kHz, and 80 dimensional Mel spectrum was used as the acoustic features. The size of the Fourier transform, frameshift length, window length, and window function are 2,048, 10 ms, 50 ms, and Hann window, respectively. We replaced bi-directional LSTM with uni-directional LSTM in Tacotron2 and attention mechanism to the forward attention with the transit agent (Zhang et al., 2018) for incremental processing. Guided Attention Loss (Tachibana et al., 2018) was used as an additional Loss function. The input size of Tactoron2 is 89, and the optimizer was Adam with the learning rate of 1e-3 and the hyperparameters of $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and $\epsilon = 1e-6$. The batch size was 32 in the number of sentences. Experimental conditions for Parallel WaveGan are the same as in the original paper, except for the parameters related to acoustic features and speech.

The pronunciation estimation used the wait-3 policy. The incremental TTS has a couple of look-ahead parameters, indicating the length to control the quality-latency trade-off. We tune these parameters to keep the quality of synthesized speech within the latency threshold requirement (2.5 seconds).

3.5 Evaluation

We evaluated our systems using SimulEval (Ma et al., 2020b) toolkit[2]. For the SimulST systems, translation quality was evaluated by BLEU using sacreBLEU[3]. Translation latency was evaluated using the following metrics:

- Average Lagging (Ma et al., 2019)
- Length Adaptive Average Lagging (Papi et al., 2022)
- Average Token Delay (Kano et al., 2023)
- Average Proportion (Cho and Esipova, 2016)
- Differentiable Average Lagging (Cherry and Foster, 2019)

For the SimulS2S system, translation quality was evaluated by BLEU after transcribing the output speech with Whisper (Radford et al., 2022) (WHISPER_ASR_BLEU). Translation latency was evaluated with ATD and the time offset of the start and end of the translation.

AL is a latency metric commonly used for text-to-text and speech-to-text simultaneous translation. However, AL focuses on when the translation starts but does not consider enough when the translation for each input chunk finishes. Since the speech segments are generated sequentially in

[1] https://open-jtalk.sourceforge.net

[2] https://github.com/facebookresearch/SimulEval

[3] https://github.com/mjpost/sacrebleu

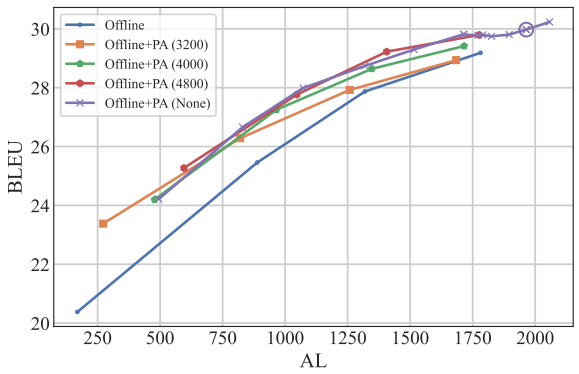


Figure 2: BLEU and AL results of the offline model and the models fine-tuned with prefix alignment on En-De. The parentheses indicate the max ratio of prefix pair filtering. Circled dots indicate our sumitted SimulS2t system.

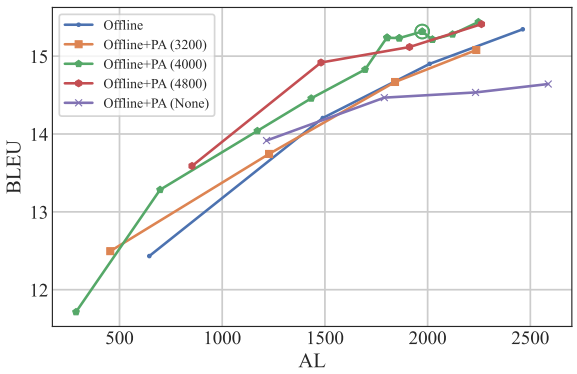


Figure 3: BLEU and AL results of the offline model and the models fine-tuned with prefix alignment on En-Ja.

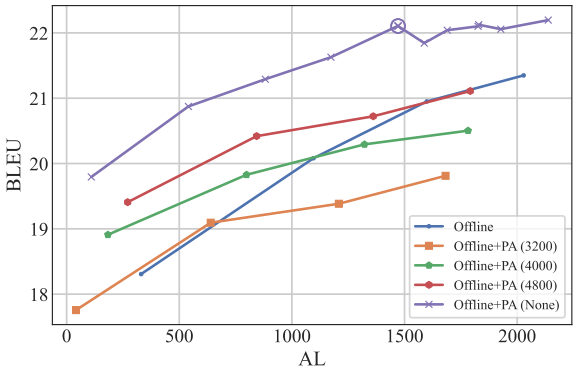


Figure 4: BLEU and AL results of the offline model and the models fine-tuned with prefix alignment on En-Zh.

a speech-to-speech translation scenario, the translation output will be delayed if its preceding translation outputs are delayed and occupy the speech output channel. Thus, AL is not appropriate to evaluate the latency of speech-to-speech simultaneous translation, so we use ATD which includes the delays caused by the outputs in the latency calculation. ATD calculates the delay by having the average time difference between the source token and its corresponding target token. In the setting of SimulEval, assuming one word requires 300 ms to speak, the input and output speech are segmented into the size of 300 ms regarding the segments as the tokens when calculating ATD.

4 Experimental Results

4.1 Submitted Speech-to-Text System

For each language direction, we selected one submission with the settings satisfying the task requirement, $AL \leq 2$ sec. Table 3 shows the scores of the submitted Speech-to-Text systems. The results of all chunk settings for the models used in the submitted systems are shown in Appendix A. The following sections discuss the effectiveness of each of the techniques we used.

4.2 Prefix Alignment

Figures 2 to 4 show quality-latency trade-offs on En-De, En-Ja, and En-Zh tst-COMMON, respectively. For En-De and En-Ja, the quality and latency were roughly proportional in the range of $AL \leq 2000$, while the quality improvement saturated at around $AL = 1,500$ for En-Zh. The fine-tuned model with Bilingual Prefix Alignment outperformed the baseline offline model for all language pairs. In En-Ja, the best results were obtained when prefix pair filtering was applied with the maximum ratio of 4,000, similar to Fukuda et al. (2022). It suggests the importance of the filtering to reduce unbalanced data pairs consisting of long source speech and short target text in language pairs with the large word order differnce. On the other hand, the prefix pair filtering did not work well for the other language directions.

4.3 Inter-connection

We analyzed the effectiveness of Inter-connection through an ablation study on the connection methods and the checkpoint averaging. The results are shown in Table 4.

The results show that checkpoint averaging improved BLEU for the En-Ja and En-Zh and that Inter-connection worked for En-De and En-Ja. This could be attributed to differences in the speech features required for speech translation.

Language pair	chunk size	BLEU	LAAL	AL	AP	DAL	ATD
En-De	950 ms	29.975	2172.927	1964.329	0.846	2856.738	1893.749
En-Ja	840 ms	15.316	2290.716	1973.586	0.892	2889.950	547.752
En-Zh	700 ms	22.105	1906.995	1471.287	0.821	2436.948	667.780

Table 3: Results of the submitted speech-to-text systems on the MuST-C v2 tst-COMMON.

Model	En-De	En-Ja	En-Zh	Ave.
Simple Connection	30.49	15.28	24.50	23.42
Simple Connection + Ckpt Ave.	30.47	15.71	**25.01**	23.73
Inter-connection	30.49	15.53	24.23	23.42
Inter-connection + Ckpt Ave.	**30.89**	**15.89**	24.75	**23.84**

Table 4: BLEU scores for models without and with checkpoint averaging for simple and Inter-connection were evaluated with MuST-C v2 tst-COMMON.

In the multilingual model, the weights required for each language pair are different because the weights of the weighted sum in Inter-connection are shared. In the case of En-Zh, there was larger difference in the weights than in En-De and En-Ja, and sharing weights leads to decrease the performance.

4.4 Computation-aware Latency

We also evaluated models with computation-aware Average Lagging (AL_CA). AL_CA is a variant of AL that adds the actual elapsed time $elapsed_i$ to the delay d_i of i-th target token y_i:

$$d_i = \sum_{k=1}^{j} (T_k + elapsed_i) \quad (1)$$

where T_k is the duration of the k-th input speech segment and j is the position of the input segment already read when generating y_i. The elapsed time $elapsed_i$ is measured as the time from the start of the translation to the output of target token y_i.

The evaluation was conducted using an NVIDIA GeForce RTX 2080 Ti. Figure 5 shows the result. Unlike the non-computation-aware latency metrics, the fixed-size segmentation worked better than the local agreement in the quality-latency trade-off. The local agreement often discards the latter part of the prefix translation due to the disagreement with the next prefix translation, while such a trackback does not happen in the fixed segmentation scenario. Therefore, the local agreement needs to predict more tokens every time and increases the decoding time. This result suggests another trade-off between quality improvement with a sophisticated segmentation strategy and latency reduction with a fixed strategy.

ASR_BLEU	StartOffset	EndOffset	ATD
9.873	2495.01	4134.752	3278.809

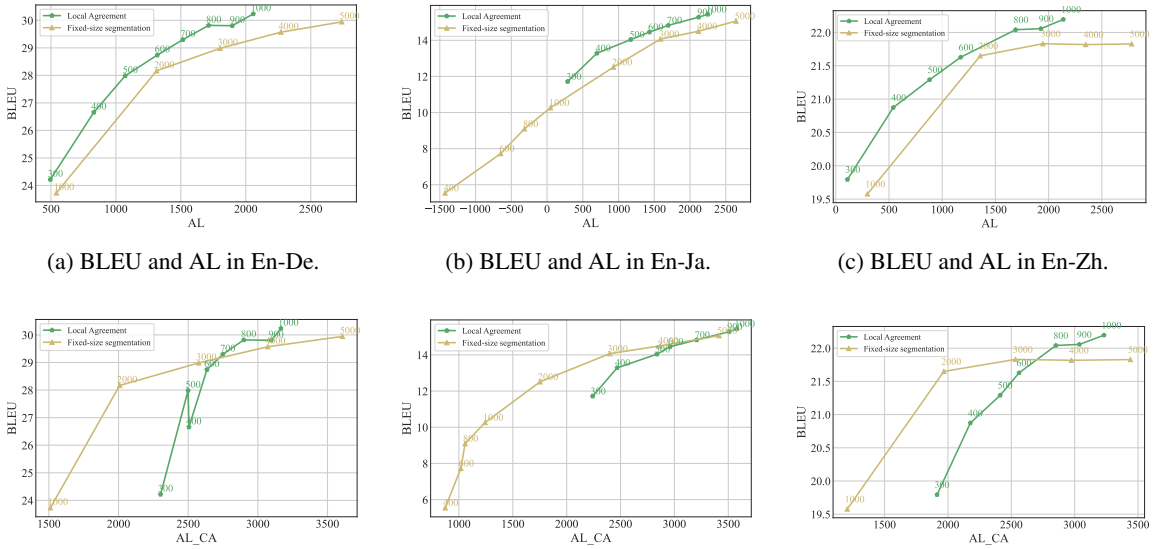
Table 5: Results of the submitted SimulS2S system on the MuST-C v2 tst-COMMON.

4.5 Submitted SimulS2S System

Table 5 shows the scores of the SimulS2S system. Compared to the BLEU results with the SimulS2T systems with similar chunk size settings, the SimulS2S system resulted in much worse ASR_BLEU in nearly five points due to the quality of the synthesized speech and possible ASR errors. Figure 6 shows the quality-latency trade-offs of SimulS2S, with ASR_BLEU stagnating around 10.5 points. In addition, the output of the submitted SimulS2S system had a character error rate of 28.3% relative to the output of the SimulS2T system with the same chunk size. These results indicate that there is a significant room for improvement both in the TTS and ASR.

5 Conclusions

In this paper, we described our SimulST systems for the IWSLT 2023 Simultaneous Speech Translation task. Experimental results demonstrated the effectivenesses of Inter-connection and Bilingual Prefix Alignment. The speech-to-speech system is still challenging but showed promising performance by a simple cascade of speech-to-text SimulST and incremental TTS.



(a) BLEU and AL in En-De. (b) BLEU and AL in En-Ja. (c) BLEU and AL in En-Zh.

(d) BLEU and AL_CA in En-De. (e) BLEU and AL_CA in En-Ja. (f) BLEU and AL_CA in En-Zh.

Figure 5: Comparison of the local agreement with $n = 2$ and fixed-size segmentation policies.

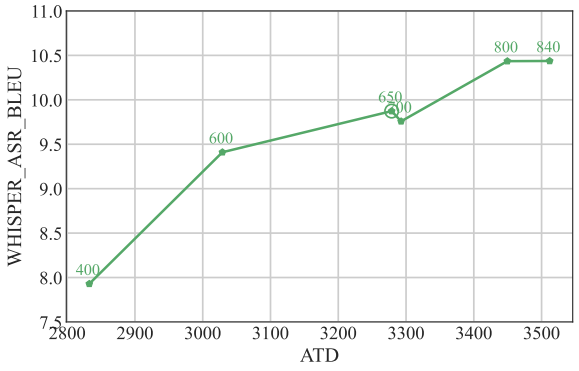


Figure 6: WHISPER_ASR_BLEU and ATD results of the SimulS2S systems on En-Ja. The numbers above the marks indicates chunk size. Circled dots indicate our sumitted system.

Acknowledgements

Part of this work was supported by JSPS KAKENHI Grant Number JP21H05054.

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.

Antonios Anastasopoulos, Loïc Barrault, Luisa Bentivogli, Marcely Zanon Boito, Ondřej Bojar, Roldano Cattoni, Anna Currey, Georgiana Dinu, Kevin Duh, Maha Elbayad, Clara Emmanuel, Yannick Estève, Marcello Federico, Christian Federmann, Souhir Gahbiche, Hongyu Gong, Roman Grundkiewicz, Barry Haddow, Benjamin Hsu, Dávid Javorský, Věra Kloudová, Surafel Lakew, Xutai Ma, Prashant Mathur, Paul McNamee, Kenton Murray, Maria Nădejde, Satoshi Nakamura, Matteo Negri, Jan Niehues, Xing Niu, John Ortega, Juan Pino, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Marco

Turchi, Yogesh Virkar, Alexander 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)*, pages 98–157, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Colin Cherry and George Foster. 2019. Thinking slow about latency evaluation for simultaneous machine translation. *arXiv preprint arXiv:1906.00048*.

Kyunghyun Cho and Masha Esipova. 2016. Can neural machine translation do simultaneous translation? *arXiv preprint arXiv:1606.02012*.

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.

Ryo Fukuda, Yuka Ko, Yasumasa Kano, Kosuke Doi, Hirotaka Tokuyama, Sakriani Sakti, Katsuhito Sudoh, and Satoshi Nakamura. 2022. NAIST simultaneous speech-to-text translation system for IWSLT 2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 286–292, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

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.

Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan. 2020. Europarl-st: A multilingual corpus for speech translation of parliamentary debates. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8229–8233.

J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. 2020. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673. https://github.com/facebookresearch/libri-light.

Yasumasa Kano, Katsuhito Sudoh, and Satoshi Nakamura. 2022. Simultaneous neural machine translation with prefix alignment. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 22–31, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Yasumasa Kano, Katsuhito Sudoh, and Satoshi Nakamura. 2023. Average token delay: A latency metric for simultaneous translation. In *Proc, Interspeech 2023*. To appear.

Eugene Kharitonov, Morgane Rivière, Gabriel Synnaeve, Lior Wolf, Pierre-Emmanuel Mazaré, Matthijs Douze, and Emmanuel Dupoux. 2020. Data augmenting contrastive learning of speech representations in the time domain. *arXiv preprint arXiv:2007.00991*.

Danni Liu, Gerasimos Spanakis, and Jan Niehues. 2020. Low-Latency Sequence-to-Sequence Speech Recognition and Translation by Partial Hypothesis Selection. In *Proc. Interspeech 2020*, pages 3620–3624.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. 2019. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3025–3036, Florence, Italy. Association for Computational Linguistics.

Mingbo Ma, Baigong Zheng, Kaibo Liu, Renjie Zheng, Hairong Liu, Kainan Peng, Kenneth Church, and Liang Huang. 2020a. Incremental text-to-speech synthesis with prefix-to-prefix framework. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3886–3896, Online. Association for Computational Linguistics.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. 2020b. SIMULEVAL: An evaluation toolkit for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 144–150, Online. Association for Computational Linguistics.

Kikuo Maekawa. 2008. Balanced Corpus of Contemporary Written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources*.

Yuta Nishikawa and Satoshi Nakamura. 2023. Interconnection: Effective connection between pretrained encoder and decoder for speech translation. In *Proc, Interspeech 2023*. To appear.

Sara Papi, Marco Gaido, Matteo Negri, and Marco Turchi. 2022. Over-generation cannot be rewarded: Length-adaptive average lagging for simultaneous speech translation. In *Proceedings of the Third Workshop on Automatic Simultaneous Translation*, pages 12–17, Online. Association for Computational Linguistics.

Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. 2021. Layer-wise analysis of a self-supervised speech representation model. *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921.

Peter Polák, Ngoc-Quan Pham, Tuan Nam Nguyen, Danni Liu, Carlos Mullov, Jan Niehues, Ondřej Bojar, and Alexander Waibel. 2022. CUNI-KIT system for simultaneous speech translation task at IWSLT 2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 277–285, Dublin, Ireland (in-person and online). 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*.

Anthony Rousseau, Paul Deléglise, and Y. Estève. 2012. Ted-lium: an automatic speech recognition dedicated corpus. In *International Conference on Language Resources and Evaluation*.

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A. Saurous, Yannis Agiomvrgiannakis, and Yonghui Wu. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4779–4783.

Ryosuke Sonobe, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2017. Jsut corpus: free large-scale japanese speech corpus for end-to-end speech synthesis. *arXiv preprint arXiv:1711.00354*.

Hideyuki Tachibana, Katsuya Uenoyama, and Shunsuke Aihara. 2018. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4784–4788.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning.

Ioannis Tsiamas, Gerard I. Gállego, Carlos Escolano, José Fonollosa, and Marta R. Costa-jussà. 2022. Pretrained speech encoders and efficient fine-tuning methods for speech translation: UPC at IWSLT 2022. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 265–276, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Changhan Wang, Juan Pino, Anne Wu, and Jiatao Gu. 2020. CoVoST: A diverse multilingual speech-to-text translation corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4197–4203, Marseille, France. European Language Resources Association.

Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. 2020. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6199–6203.

Jing-Xuan Zhang, Zhen-Hua Ling, and Li-Rong Dai. 2018. Forward attention in sequence- to-sequence acoustic modeling for speech synthesis. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4789–4793.

A Appendix

Tables 6, 7, and 8 show the results for all chunk size settings for the En-De, En-Ja, and En-Zh models used in the submitted system, respectively.

chunk size	BLEU	LAAL	AL	AP	DAL	ATD
300	24.217	947.509	495.162	0.732	1465.822	814.368
400	26.657	1189.696	829.689	0.753	1738.568	1180.684
500	27.986	1416.459	1071.682	0.774	1992.596	1375.404
600	28.739	1618.746	1318.715	0.791	2232.175	1367.612
700	29.298	1797.061	1515.356	0.811	2432.087	1608.334
800	29.809	1956.321	1714.173	0.826	2617.073	1720.705
820	29.78	2011.518	1772.404	0.827	2672.554	1765.76
840	29.792	2022.322	1790.452	0.832	2680.218	1741.386
860	29.746	2054.923	1825.194	0.834	2726.204	1740.656
900	29.805	2115.625	1895.961	0.841	2783.033	1711.2
950	29.975	2172.927	1964.329	0.846	2856.738	1893.749
1000	30.234	2255.583	2057.579	0.852	2938.408	1884.775

Table 6: Results of the **Offline+PA (None)** model on the MuST-C v2 tst-COMMON En-De.

chunk size	BLEU	LAAL	AL	AP	DAL	ATD
300	11.714	1096.676	288.185	0.807	1643.59	181.268
400	13.284	1377.647	697.522	0.827	1949.44	260.12
500	14.04	1642.289	1171.154	0.845	2246.513	343.565
600	14.458	1858.317	1433.278	0.866	2463.025	386.054
700	14.828	2064.974	1695.339	0.877	2672.509	471.012
800	15.235	2224.392	1803.111	0.895	2831.076	519.566
820	15.232	2256.386	1862.014	0.892	2865.29	537.516
840	15.316	2290.716	1973.586	0.892	2889.95	547.752
860	15.214	2341.734	2023.29	0.896	2946.322	557.76
900	15.281	2389.836	2121.337	0.898	3010.863	563.603
1000	15.439	2528.8	2247.036	0.907	3126.384	630.97

Table 7: Results of the **Offline+PA (4000)** model on the MuST-C v2 tst-COMMON En-Ja.

chunk size	BLEU	LAAL	AL	AP	DAL	ATD
300	19.794	1011.202	109.706	0.755	1411.409	206.106
400	20.874	1283.497	540.576	0.774	1718.894	370.356
500	21.291	1522.251	881.957	0.796	1984.268	474.854
600	21.628	1714.688	1173.412	0.811	2216.213	499.254
700	22.105	1906.995	1471.287	0.821	2436.948	667.78
750	21.844	1994.88	1587.405	0.83	2526.013	672.637
800	22.041	2071.358	1689.633	0.831	2621.874	738.394
840	22.101	2126.632	1826.245	0.829	2689.418	761.502
860	22.125	2167.874	1829.369	0.836	2728.565	760.173
900	22.057	2211.844	1927.426	0.838	2779.555	749.444
1000	22.196	2383.854	2137.905	0.851	2946.303	882.875

Table 8: Results of the **Offline+PA (None)** model on the MuST-C v2 tst-COMMON En-Zh.