CMU's IWSLT 2023 Simultaneous Speech Translation System

Brian Yan*[1] Jiatong Shi*[1] Soumi Maiti[1] William Chen[1]
Xinjian Li[1] Yifan Peng[2] Siddhant Arora[1] Shinji Watanabe[1,3]
[1]Language Technologies Institute, Carnegie Mellon University, USA
[2]Electrical and Computer Engineering, Carnegie Mellon University, USA
[3]Human Language Technology Center of Excellence, Johns Hopkins University, USA
{byan, jiatongs}@cs.cmu.edu

Abstract

This paper describes CMU's submission to the IWSLT 2023 simultaneous speech translation shared task for translating English speech to both German text and speech in a streaming fashion. We first build offline speech-to-text (ST) models using the joint CTC/attention framework. These models also use WavLM front-end features and mBART decoder initialization. We adapt our offline ST models for simultaneous speech-to-text translation (SST) by 1) incrementally encoding chunks of input speech, re-computing encoder states for each new chunk and 2) incrementally decoding output text, pruning beam search hypotheses to 1-best after processing each chunk. We then build text-to-speech (TTS) models using the VITS framework and achieve simultaneous speech-to-speech translation (SS2ST) by cascading our SST and TTS models.

1 Introduction

In this paper, we present CMU's English to German simultaneous speech translation systems. Our IWSLT 2023 (Agarwal et al., 2023) shared task submission consists of both simultaneous speech-to-text (SST) and simultaneous speech-to-speech (SS2ST) systems. Our general strategy is to first build large-scale offline speech translation (ST) models which leverage unpaired speech data, ASR data, and ST data. We then adapt these offline models for simultaneous inference. Finally, we use a text-to-speech model to achieve SS2ST in a cascaded manner.

In particular, our system consists of:

1. Offline ST using joint CTC/attention with self-supervised speech/text representations (§3.1)
2. Offline-to-online adaptation via chunk-based encoding and incremental beam search (§3.2)
3. Simultaneous S2ST by feeding incremental text outputs to a text-to-speech model (§3.3)

2 Task Description

The IWSLT 2023 simultaneous speech translation track[1] is a shared task for streaming speech-to-text and speech-to-speech translation of TED talks. This track mandates that systems do not perform re-translation, meaning that the streaming outputs cannot be edited after the system receives more input audio. Systems are required to meet a particular latency regime: SST systems must have <2 seconds average lagging (AL) and SS2ST systems must have <2.5 seconds start offset (SO) (Ma et al., 2020).

Of the allowed training data, we selected a subset of in-domain data to train our ASR and ST models: for ASR we use TEDLIUM v1 and v2 (Zhou et al., 2020) and for ST we used MuST-C v2 (Di Gangi et al., 2019). We also use a set of cross-domain data to train our MT and TTS models due to the lack of in-domain data: for MT we use Europarl, NewsCommentary, OpenSubtitles, TED2020, Tatoeba, and ELRC-CORDIS News (Tiedemann et al., 2020). For TTS we use CommonVoice (Ardila et al., 2020). The following section describes how each of the ASR, ST, MT, and TTS components fit together in our ultimate systems.

3 System Description

3.1 Offline Speech Translation (ST)

As shown in Figure 1, our offline ST models are based on the joint CTC/attention framework (Watanabe et al., 2017; Yan et al., 2023a). Compared to a purely attention-based approach, joint CTC/attention has been shown to reduce the soft-alignment burden, provide a positive ensembling effect, and improve the robustness of end-detection during inference (Yan et al., 2023a).

To leverage unpaired speech data, we use first use WavLM representations (Chen et al., 2022) as

[1]https://iwslt.org/2023/simultaneous

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 235–240

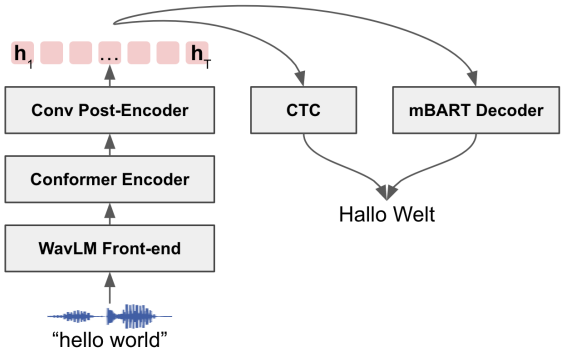

Figure 1: Offline ST model architecture based on the joint CTC/attention framework with a WavLM front-end and mBART decoder.

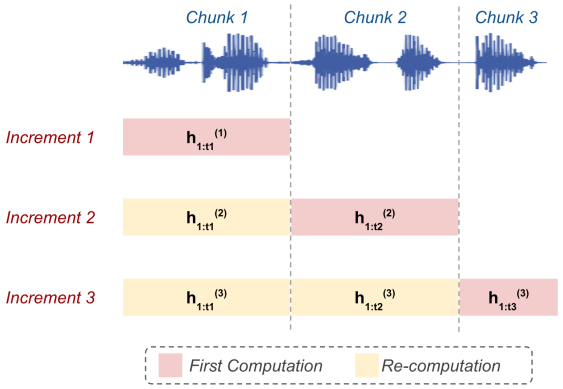

Figure 2: Incremental encoding strategy which processes chunks of input speech by re-computing representations corresponding to earlier chunks.

front-end features to train ASR models. In these models, a pre-encoder module (Chang et al., 2021) applies feature dimension down-sampling and a learned weighted combination of WavLM layers before feeding to a Conformer encoder (Gulati et al., 2020). The pre-encoder and encoder modules from ASR are then used to initialize our ST models.

To leverage unpaired text data, we use the mBART decoder (Tang et al., 2020) as an initialization for our ST models. Following (Li et al., 2020), we freeze all feed-forward layers during fine-tuning and use a post-encoder down-sampling layer to reduce the computational load.

We fine-tune our ST models using the following interpolated loss function: $\mathcal{L} = \lambda_1 \mathcal{L}_{\text{ASR\_CE}} + \lambda_2 \mathcal{L}_{\text{ASR\_CTC}} + \lambda_3 \mathcal{L}_{\text{ST\_CE}} + \lambda_4 \mathcal{L}_{\text{ST\_CTC}}$. Here, the cross-entropy (CE) losses are used to train attentional decoders. Note that in Figure 1, we omit the ASR attentional decoder and CTC components as these function as training regularizations and do not factor into the inference proceedure. We perform fine-tuning on in-domain data consisting primarily of MuST-C (Di Gangi et al., 2019).

To leverage additional in-domain data, we apply MT pseudolabeling to TEDLIUM ASR data (Zhou et al., 2020). We also use the same MT model to apple sequence-level knowledge distillation to the MuST-C data. The MT model is a pre-trained DeltaLM-large (Ma et al., 2021) fine-tuned on the corpora listed in Section 2. The pseudo-labels and distilled sequences were then translated from English to German using a beam size of 10.

3.2 Simultaneous Speech Translation (SST)

We adapt our offline ST model for streaming inference by using a chunk-based processing of input speech. As shown in Figure 2, our scheme uses a fixed duration (e.g. 2 seconds) to compute front-end and encoder representations on chunks of input speech. With each new chunk, we re-compute front-end and encoder representations using the incrementally longer input speech.

To produce incremental translation outputs, we apply several modifications to the offline joint CTC/attention beam search. As shown in Algorithm 1, we run beam search for each chunk of input. Unless we know that the current chunk is the final chunk, we perform end-detection using the

Algorithm 1 Beam search step with rewinding of unreliable hypotheses on non-final chunks and incremental pruning upon end-detection.

procedure BEAMSTEP(hyps, prevHyps, isFinal)
  newHyps = {}; endDetected = False
  for $y_{1:l-1} \in$ prtHs **do**
    attnCnds = top-k($P_{\text{Attn}}(y_l|X, y_{1:l-1}), \text{k} = p$)
    for $c \in$ attnCnds **do**
      $y_{1:l} = y_{1:l-1} \oplus \text{c}$
      $\alpha_{\text{CTC}} = \text{CTCScore}(y_{1:l}, X_{1:T})$
      $\alpha_{\text{Attn}} = \text{AttnScore}(y_{1:l}, X_{1:T})$
      $\beta = \text{LengthPen}(y_{1:l})$
      $P_{\text{Beam}}(y_{1:l}|X) = \alpha_{\text{CTC}} + \alpha_{\text{Attn}} + \beta$
      newHyps[$y_{1:l}$] = $P_{\text{Beam}}(\cdot)$
      if (!isFinal) and (c is <eos> or repeat) **then**
        endDetected = True
        newHyps = prevHyps ▷ rewind
      else if l is max$L$ **then**
        endDetected = True
      end if
    end for
  end for
  if endDetected **then** ▷ incremental pruning
    newHyps = top-k($P_{\text{Beam}}(\cdot), \text{k} = 1$)
  else ▷ standard pruning
    newHyps = top-k($P_{\text{Beam}}(\cdot), \text{k} = b$)
  end if
  return newHyps, endDetected
end procedure

Model	Quality	Latency	
Offline Speech Translation (ST)	BLEU ↑	-	
Multi-Decoder CTC/Attn (Yan et al., 2023b)	30.1	-	-
WavLM-mBART CTC/Attn (Ours)	32.5	-	-
Simul Speech Translation (SST)	BLEU ↑	AL ↓	LAAL ↓
Time-Sync Blockwise CTC/Attn (Yan et al., 2023b)	26.6	1.93	1.98
WavLM-mBART CTC/Attn (Ours)	30.4	1.92	1.99
Simul Speech-to-Speech Translation (SS2T)	ASR-BLEU ↑	SO ↓	EO ↓
WavLM-mBART CTC/Attn + VITS (Ours)	26.7	2.33	5.67

Table 1: Results of our English to German ST/SST/SS2ST models on MuST-C-v2 tst-COMMON.

heuristics introduced by (Tsunoo et al., 2021). If any of the hypotheses in our beam propose a next candidate which is the special end-of-sequence token or a token which already appeared in the hypothesis, then this strategy determines that the outputs have likely covered all of the available input. At this point, the current hypotheses should be considered unreliable and thus the algorithm rewinds hypotheses to the previous step.

After the end has been detected within the current chunk, we prune the beam to the 1-best hypothesis and select this as our incremental output – this pruning step is necessary to avoid re-translation. When the next input chunk is received, beam search continues from this 1-best hypothesis.

3.3 Simultaneous Speech-to-Speech Translation (S2ST)

Simultaneous S2ST model is created by feeding incremental text outputs to a German text-to-speech model. We use end-to-end TTS model VITS (Kim et al., 2021) and train a single speaker German TTS model using CommonVoice dataset(Ardila et al., 2020). VITS consists of text-encoder, flow based stochastic duration predictor from text, variational auto-encoder for learning latent feature from audio and generator-discriminator based decoder for generating speech from latent feature. We use character as input to the TTS model.

We select a suitable speaker from CommonVoice German dataset and train single speaker TTS. As CommonVoice may contain many noisy utterances which can hurt performance of TTS, we use data-selection for high-quality subset. The data selection process involves identifying the speaker who has the highest number of utterances with high speech quality. To determine the speech quality, we use speech enhancement metric DNSMOS (Reddy et al., 2021) which provides an estimation of the speech quality. We evaluate the speech quality for the top five speakers with the largest number of utterances. To establish the high-quality subset, we set a threshold of 4.0 for selecting sentences that meet the desired quality level. Based on this criterion, we choose the second speaker, who has approximately 12 hours of high-quality data.

Finally, we combine our trained German TTS model with SST module during inference. We feed incremental translation text outputs to TTS and synthesize translated speech.

4 Experimental Setup

Our models were developed using the ESPnet-ST-v2 toolkit (Yan et al., 2023b). Our ST/SST model uses WavLM-large as a front-end (Chen et al., 2022). A linear pre-encoder down-samples from 1024 to 80 feature dim. Our encoder is a 12 layer Conformer with 1024 attention dim, 8 attention heads, and 2048 linear dim (Gulati et al., 2020). A convolutional post-encoder then down-samples along the length dimension by a factor of 2. Our decoder follows the mBART architecture and we initialize using the mBART-large-50-many-to-many model (Tang et al., 2020). Our ST CTC branch uses the same 250k vocabulary as the mBART decoder to enable joint decoding. Our TTS model consists of 6 transformer encoder layers for text-encoder, 4 normalizing flow layers for duration predictor, 16 residual dilated convolutional blocks as posterior encoder and multi-period HiFiGan (Kong et al., 2020) style decoder. We train VITS model for 400 epochs with AdamW (Loshchilov and Hutter, 2019) optimizer.

During inference, we use a chunk size of 2 sec-

onds for SST and 2.5 seconds for SS2ST. For both SST and SS2ST we use beam size 5, CTC weight 0.2, and no length penalty/bonus. To account for incremental outputs which end in a prefix of a word rather than a whole word, we delay outputs for scoring by 1 token. There are two exceptions to this token delay: if the last token is a valid German word or a punctuation, then we do not delay.

We evaluate translation quality using BLEU score (Papineni et al., 2002) for ST/SST and ASR-BLEU score for SS2ST. ST/SST references are case-sensitive and punctuated while SS2ST references are case-insensitive and un-punctuated. The ASR model used for ASR-BLEU is Whisper-small (Radford et al., 2022). We evaluate translation latency for SST using average lagging (AL) (Ma et al., 2020) and length-adaptive average lagging (LAAL) (Papi et al., 2022). We evaluate translation latency for SS2ST using start (SO) and end-offset (EO) (Ma et al., 2020).

5 Results

Table 1 shows the quality and latency of our SST and SS2ST models as measured on En-De tst-COMMON. We also show the ST performance of our model for reference. As a baseline, we compare to the IWSLT-scale ST and SST systems developed in Yan et al. (2023b) – our systems show improved quality, primarily due to the use of WavLM and mBART self-supervised representations.

From ST to SST, we observe a 6% quality degradation. Note that the average duration of tst-COMMON utterances is around 5 seconds, meaning the corresponding latency gain is 60%. From SST to SS2ST, we observe a 12% quality degradation. Note that both the TTS model and the Whisper ASR model powering the ASR-BLEU metric contribute to this gap.

6 Conclusion

We describe our English to German simultaneous speech-to-text and speech-to-speech translation systems for the IWSLT 2023 shared task. We start by building large-scale offline speech-to-text systems which leverage self-supervised speech and text representations. We then adapt these offline models for online inference, enabling simultaneous speech-to-text translation. Finally, we feed streaming text outputs to a down-stream TTS model, enabling simultaneous speech-to-speech translation.

Acknowledgements

Brian Yan and Shinji Watanabe are supported by the Human Language Technology Center of Excellence. This work used the Extreme Science and Engineering Discovery Environment (XSEDE) (Towns et al., 2014), which is supported by National Science Foundation grant number ACI-1548562; specifically, the Bridges system (Nystrom et al., 2015), as part of project cis210027p, which is supported by NSF award number ACI-1445606, at the Pittsburgh Supercomputing Center. This work also used GPUs donated by the NVIDIA Corporation.

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.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Xuankai Chang, Takashi Maekaku, Pengcheng Guo, Jing Shi, Yen-Ju Lu, Aswin Shanmugam Subramanian, Tianzi Wang, Shu-wen Yang, Yu Tsao, Hung-yi Lee, et al. 2021. An exploration of self-supervised pretrained representations for end-to-end speech recognition. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 228–235. IEEE.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki

Kanda, Takuya Yoshioka, Xiong Xiao, et al. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

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.

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 *Proceedings of Interspeech*, pages 5036–5040.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*. PMLR.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. volume 33, pages 17022–17033.

Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2020. Multilingual speech translation with efficient finetuning of pretrained models. *arXiv preprint arXiv:2010.12829*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, Alexandre Muzio, Saksham Singhal, Hany Hassan Awadalla, Xia Song, and Furu Wei. 2021. DeltaLM: Encoder-decoder pre-training for language generation and translation by augmenting pretrained multilingual encoders.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. 2020. Simuleval: An evaluation toolkit for simultaneous translation. In *Proceedings of the EMNLP*.

Nicholas A Nystrom, Michael J Levine, Ralph Z Roskies, and J Ray Scott. 2015. Bridges: a uniquely flexible hpc resource for new communities and data analytics. In *Proceedings of the 2015 XSEDE Conference: Scientific Advancements Enabled by Enhanced Cyberinfrastructure*, pages 1–8.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. 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*.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. 2021. Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE.

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.

Jörg Tiedemann, Santhosh Thottingal, et al. 2020. Opus-mt–building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*. European Association for Machine Translation.

J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkins-Diehr. 2014. Xsede: Accelerating scientific discovery. *Computing in Science & Engineering*, 16(5):62–74.

Emiru Tsunoo, Yosuke Kashiwagi, and Shinji Watanabe. 2021. Streaming transformer asr with blockwise synchronous beam search. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 22–29. IEEE.

Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. 2017. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253.

Brian Yan, Siddharth Dalmia, Yosuke Higuchi, Graham Neubig, Florian Metze, Alan W Black, and Shinji Watanabe. 2023a. CTC alignments improve autoregressive translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1615–1631, Dubrovnik, Croatia. Association for Computational Linguistics.

Brian Yan, Jiatong Shi, Yun Tang, Hirofumi Inaguma, Yifan Peng, Siddharth Dalmia, Peter Polák, Patrick Fernandes, Dan Berrebbi, Tomoki Hayashi, et al. 2023b. Espnet-st-v2: Multipurpose spoken language translation toolkit. *arXiv preprint arXiv:2304.04596*.

Wei Zhou, Wilfried Michel, Kazuki Irie, Markus Kitza, Ralf Schlüter, and Hermann Ney. 2020. The rwth asr system for ted-lium release 2: Improving hybrid hmm with specaugment. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7839–7843. IEEE.