The Xiaomi AI Lab's Speech Translation Systems for IWSLT 2023 Offline Task, Simultaneous Task and Speech-to-Speech Task

Wuwei Huang[1*†] **Mengge Liu**[2*‡] **Xiang Li**[1] **Yanzhi Tian**[2‡] **Fengyu Yang**[1]
Wen Zhang[1] **Yuhang Guo**[2] **Jinsong Su**[3] **Jian Luan**[1] **Bin Wang**[1]
[1]Xiaomi AI Lab, Beijing, China
[2]Beijing Institute of Technology, Beijing, China
[3]Xiamen University, Xiamen, Fujian, China.
{huangwuwei,lixiang21,yangfengyu1,zhangwen17,luanjian,wangbin11}@xiaomi.com
{liumengge,tianyanzhi,guoyuhang}@bit.edu.cn jssu@xmu.edu.cn

Abstract

This system description paper introduces the systems submitted by Xiaomi AI Lab to the three tracks of the IWSLT 2023 Evaluation Campaign, namely the offline speech translation (Offline-ST) track, the offline speech-to-speech translation (Offline-S2ST) track, and the simultaneous speech translation (Simul-ST) track. All our submissions for these three tracks only involve the English-Chinese language direction. Our English-Chinese speech translation systems are constructed using large-scale pre-trained models as the foundation. Specifically, we fine-tune these models' corresponding components for various downstream speech translation tasks. Moreover, we implement several popular techniques, such as data filtering, data augmentation, speech segmentation, and model ensemble, to improve the system's overall performance. Extensive experiments show that our systems achieve a significant improvement over the strong baseline systems in terms of the automatic evaluation metric.

1 Introduction

We submit an end-to-end offline speech translation system, a cascaded offline speech-to-speech translation system, and an end-to-end simultaneous interpretation system to the Offline-ST track, Offline-S2ST track, and Simul-ST track, respectively. This paper provides a detailed description of the three systems we submit.

There are two commonly used solutions for speech translation models: the end-to-end approach and the cascaded approach. The cascaded system uses a pipeline where an automatic speech recognition (ASR) system is followed by a machine translation (MT) system. The ASR system first transcribes the speech utterances in the source language into text in the same language, and then the MT model translates the ASR output into text in the target language. In contrast, the end-to-end ST system directly translates speech utterances in the source language into text in the target language.

The scarcity of training data makes end-to-end systems still slightly inferior in translation quality to cascaded systems, which suffer from error propagation and information loss (Sperber and Paulik, 2020). Cascaded systems continue to dominate the systems submitted at IWSLT in previous years (Anastasopoulos et al., 2022, 2021; Ansari et al., 2020). However, with the rapid development of pre-training technology, a large number of large-scale pre-training models suitable for various modalities, such as speech (Baevski et al., 2020; Hsu et al., 2021; Tang et al., 2022) and text (Liu et al., 2020), have emerged. Therefore, end-to-end ST systems have gradually attracted attention from both the academic and industrial communities in recent years. In our submission, we have opted for an end-to-end approach to establish the ST system.

We briefly introduce the submitted systems:

Offline Speech Translation System. Our submitted end-to-end offline speech-to-text translation system is based on two pre-trained models: HuBERT (Hsu et al., 2021) and mBART (Liu et al., 2020). It has been proven that these two models have strong capabilities on ST and MT tasks, respectively. Our offline ST model consists of a *speech encoder*, a *text encoder*, and a *text decoder*, with all parameters initialized using the pre-trained HuBERT and mBART models.

Offline Speech-to-Speech Translation System. Speech-to-speech translation has great application value in various scenarios, such as international online lectures and multinational meetings. Lee et al. (2022) trained a sequence-to-sequence speech-to-unit translation (S2UT) model to directly predict the discrete representations of the target speech. Drawing on the method of Lee et al. (2022), we

[*] Equal contribution.
[†] Crossponding Author.
[‡] The work was done during the author's internship at Xiaomi.

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 411–419
July 13-14, 2023 ©2023 Association for Computational Linguistics

implement a cascaded speech-to-speech translation system. Specifically, an end-to-end speech-to-text translation model is trained, followed by a text-to-speech (TTS) synthesis model.

To implement a cascaded speech-to-speech translation system, we first train an end-to-end speech-to-text translation model, followed by a text-to-speech (TTS) synthesis model that we train.

Simultaneous Speech Translation System. Apart from the above two offline systems, we also submit an end-to-end system for the English-Chinese language direction in the Simul-ST track. Simultaneous speech translation involves the challenge of striking a balance between translation quality and latency, as the system starts to translate the input audio even before the entire speech input is received. The Information-Transport-based Simultaneous Translation (ITST) (Zhang and Feng, 2022) architecture is adopted to build our end-to-end Simul-ST system, and we initialize its corresponding components using the HuBERT and mBART pre-trained models. When the AL value is less than 2000, our submitted end-to-end simultaneous ST system achieves a significant improvement of +3.2 BLEU scores over last year's best end-to-end simultaneous ST system. We also explore a streaming simultaneous interpretation approach by training an offline model and applying a wait-k decoding strategy, which even yields better performance.

The rest of this paper is organized as follows: Section 2 describes the data preparation, including data filtering, data augmentation, speech segmentation, etc. Section 3 elaborates on the models and strategies used in our systems. We present our experiment settings, results, and analyses in Section 4. Finally, Section 5 provides the conclusion.

2 Data Preparation

2.1 Statistics

Our English-Chinese (abbreviated as En⇒Zh) ST systems are developed under constrained conditions using two allowed ST corpora: MuST-C v2.0[1] and CoVoST[2]. The only text translation dataset available is OpenSubtitles2018[3]. To construct the English ASR corpus, we gather data from various sources, such as LibriSpeech[4], CommonVoice[5], TED LIUM[6], and Vox Populi[7]. In addition to this, we also utilize the audio-transcription pairs from English-German (En⇒De) ST data, including ST-TED, Europarl-ST, and MuST-C (indicated with a star in Table 1). Furthermore, AISHELL-3[8] and GigaS2S[9] datasets are used to train the TTS model. We filter out those samples in the MuST-C En⇒De training set whose source sentences are included in the MuST-C En⇒Zh training set. Table 1 presents the statistical results of the training samples for different tasks.

Corpora		**Duration**	**#Spl.**
ST	**MuST-C v2.0**	596h	359K
	CoVoST	1119h	870K
	GigaST	10000h	7.6M
MT	**OpenSubtitles**	-	11.2M
ASR	**LibriSpeech**	960h	273K
	Common Voice	2320h	1.62M
	TED LIUM (v3)	452h	268K
	Vox Populi	543h	181K
	ST-TED*	273h	171K
	Europal-ST*	~80h	30K
	MuST-C*	~100h	78K
TTS	**AISHELL-3**	85h	88K
	GigaS2S	10000h	7.6M
Unlabeled Audio	**Vox Populi**	24100h	-

Table 1: The statistical results of all available training corpora in the En⇒Zh translation direction for the offline speech translation track, the offline speech-to-speech translation track, and the simultaneous speech translation track. The tilde symbol (~) indicates a rough estimation. #Spl. indicates the number of samples.

2.2 Offline-ST and Simul-ST Corpus

For both the En⇒Zh offline speech translation and En⇒Zh simultaneous speech translation tracks, we use the same training corpus, the same data filtering and data augmentation methods.

2.2.1 Data Filtering

All text data involved in MT, ST, and TTS tasks are tokenized using `SentencePiece`[10]. For the MT data, we adopt heuristic rules to filter out noisy data

[1] https://ict.fbk.eu/must-c/

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

[3] https://opus.nlpl.eu/OpenSubtitles2018.php

[4] http://www.openslr.org/12/

[5] https://commonvoice.mozilla.org/en/datasets

[6] https://lium.univ-lemans.fr/en/ted-lium3/

[7] https://github.com/facebookresearch/voxpopuli

[8] https://www.aishelltech.com/aishell_3

[9] https://github.com/SpeechTranslation/GigaS2S

[10] https://github.com/google/sentencepiece

in the training set similar to the rules used in (Guo et al., 2022), following these steps:

- A series of hand-crafted rules are adopted to filter out noisy sentences from the training set. In particular, we discard sentences that contain less than 50% linguistic words. For Chinese sentences, Chinese characters are considered linguistic words; for English sentences, words containing only alphabet characters are considered linguistic words;
- We utilize `fast_align`[11] open source tool to exclude sentence pairs with a score lower than -8. We also apply the language identification (LangID) tool[12] to filter out sentence pairs that are neither in Chinese nor English;
- Duplicate sentence pairs are discarded, and any pairs with a length ratio greater than 3.0 or sentences with a length exceeding 200 are also filtered out.

To filter out noise data in the ST training set, we apply the following steps:

- Pairs that have an audio duration exceeding 60 seconds or a text length exceeding 200 tokens are excluded;
- We calculate the ratio of the number of speech frames to tokens in each sample, and remove samples whose ratio exceeds three times the average ratio.

2.2.2 Data Augmentation

To effectively train an end-to-end speech translation model, it is impractical to rely solely on hand-annotated training data, due to the scarcity of hand-annotated data. To mitigate this issue, we utilize a well-trained MT model to translate the transcriptions from ASR data and synthesize a large amount of pseudo-data, which has been widely used in the previous years' competitions (Ding and Tao, 2021; Zhang and Ao, 2022; Zhang et al., 2022b; Li et al., 2022; Zhu et al., 2022).

We initially gather all available English-Chinese bilingual parallel sentence pairs from ST and MT tasks, as listed in Table 1. We then filter the data using the method mentioned in Section 2.2.1, generating 9M sentence pairs. These 9M sentence pairs are used to fine-tune the pre-trained one-to-many mBART50 model for 30 epochs. We further fine-tune mBART50 for another 30 epochs using MuST-C datasets to improve the domain adaptability of the model. The results are shown in Table 2.

Models	BLEU
mBART50 (one-to-many)	25.81
+ domain fine-tuning on 9M corpus	28.41
+ domain fine-tuning on MuST-C	29.50

Table 2: The BLEU scores of MT models obtained by fine-tuning one-to-many mBART50 model using various bilingual datasets on the tst-COMMON test set.

In the Librispeech and TED-LIUM datasets, English sentences do not have punctuation or case information. We fine-tune the mBART50 model to add punctuation and restore case information to English sentences. Furthermore, samples already included in the CoVoST corpus are removed from the CommonVoice dataset. The transcriptions of the ASR data are then translated using the best fine-tuned mBART50 model and filtered using the same rules as the ST data in Section 2.2.1, resulting in a total of 1.6 million synthesized speech-to-text translation pairs.

Finally, for constrained data, we combine the hand-annotated ST corpus with the synthesized ST corpus to produce the final training corpus for the Offline-ST and Simul-ST models, yielding a total of 2.9 million speech-to-text translation pairs. In the case of unconstrained training on the offline track, we augment our training corpus with the GigaST corpus, resulting in 9 million speech-to-text translation pairs.

2.3 Cascaded S2ST Corpus

In the En⇒Zh speech-to-speech translation track, we leverage all available constrained data from the offline speech translation track as well as the GigaST corpus[13] to train our offline speech translation model. This model is then followed by a TTS model that is trained on the AISHELL-3 and GigaS2S datasets.

2.4 Speech Segmentation

Since the speech in the evaluation set is not pre-segmented, we apply SHAS (Tsiamas et al., 2022) to segment the full speech into shorter segments. However, we observe two issues. Firstly, some segments have incomplete final words, which could negatively impact the performance of the ST model. To alleviate this problem, we add a few extra frames

[11] `https://github.com/clab/fast_align`

[12] `https://github.com/saffsd/langid.py`

[13] `https://st-benchmark.github.io/resources/GigaST.html`

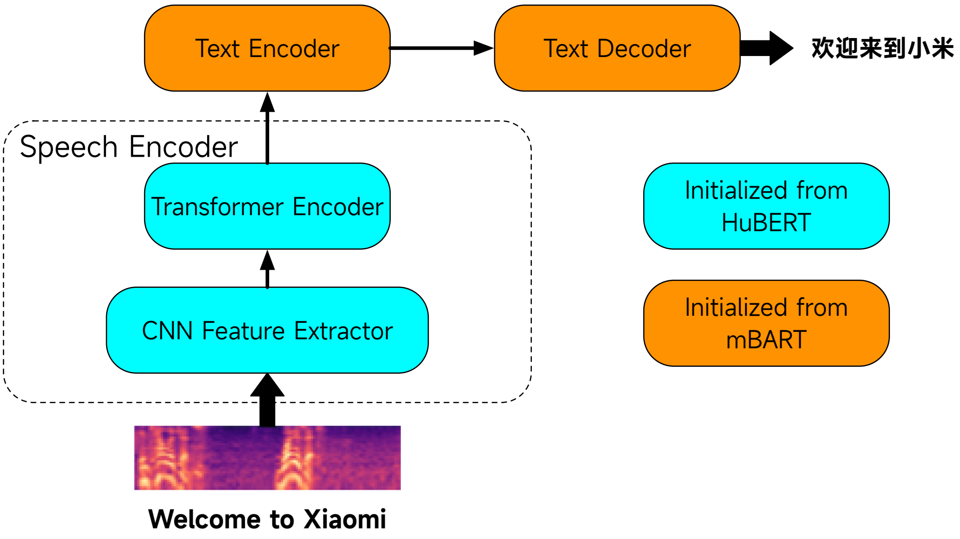

Figure 1: The architecture of our end-to-end offline speech translation model consists of three components: *speech encoder*, *text encoder*, and *text decoder*. The speech encoder is composed of a *CNN feature extractor* and a 24-layer *Transformer encoder* with a CNN positional encoder. Both the text encoder and the text decoder are 12-layer standard Transformer structures. Note that the speech encoder is initialized with the pre-trained HuBERT model, and both the text encoder and text decoder are initialized with the pre-trained mBART model.

at the end of each segment to ensure that the final word is fully pronounced. Secondly, the speaking rate varies among different speakers or types of speeches, resulting in different amounts of words being spoken within a given time period. Excessive words in a speech segment may result in missing translations. We choose different hyperparameters for different speakers or different types of speeches.

3 Methods

We build our Offline-ST system in an end-to-end manner (End-to-End Offline-ST) based on the HuBERT and mBART pre-trained models. Our simultaneous speech translation system (End-to-End Simul-ST) utilizes the same model architecture as the Offline-ST system and adopts wait-k and ITST strategies. The cascaded S2ST system involves an end-to-end speech-to-text translation model followed by a TTS model.

3.1 End-to-End Offline-ST System

The speech translation corpus typically consists of triples (x, z, y) that contain speech, transcription, and translation data, where x = $(x_1, \cdots, x_{|x|})$ represents a sequence of acoustic features, while z = $(z_1, \cdots, z_{|z|})$ and y = $(y_1, \cdots, y_{|y|})$ denote the corresponding transcription in the source language and translation in the target language, respectively.

Our end-to-end Offline-ST system is based on an encoder-decoder architecture from the pre-trained HuBERT and mBART models. Figure 1 illustrates the architecture of our model, which consists of a *speech encoder*, a *text encoder*, and a *text decoder*. More specifically, the speech encoder is composed of a feature extractor based on convolutional neural networks (CNN), named *CNN feature extractor* and a 24-layer *Transformer encoder*. The CNN feature extractor is used to extract speech features from waveform, with 7 layers each containing 512 channels and kernel widths of $[10, 3, 3, 3, 3, 2, 2]$ and strides of $[5, 2, 2, 2, 2, 2, 2]$. The Transformer encoder is derived from the standard Transformer (Vaswani et al., 2017) encoder, except for using CNN as the position encoder. The text encoder is a 12-layer standard Transformer encoder, and the text decoder is a 12-layer standard Transformer decoder. The training objective of our speech translation model can be formulated as:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta}_e, \boldsymbol{\theta}_d) = \sum_{t=1}^{|\mathbf{y}|} -\log p\left(\mathbf{y}_t | \mathbf{y}_{<t}, \mathbf{x}; \boldsymbol{\theta}_e, \boldsymbol{\theta}_d\right) \quad (1)$$

where $\boldsymbol{\theta}_e$ and $\boldsymbol{\theta}_d$ represent the parameters of the encoder and the decoder, respectively.

3.2 Cascaded S2ST System

In the cascaded S2ST system, we reuse the offline speech translation model discussed in Section 3.1 as the ST model. For the TTS model, we first train a base TTS model and vocoder using the AISHELL-3 dataset with the Tacotron2 (Shen et al., 2018) open source framework. The final TTS model is obtained by fine-tuning the base model on the GigaS2S dataset.

3.3 End-to-End Simul-ST System

In order to take full advantage of the powerful capabilities of large pre-trained models, we develop an end-to-end Simul-ST system based on the HuBERT and mBART models. Furthermore, we employ two strategies, namely wait-k and ITST.

3.3.1 Wait-k

Ma et al. (2020b) adapts methods originally proposed for simultaneous machine translation to develop an end-to-end Simul-ST system. To achieve this, they employ the wait-k (Ma et al., 2019) strategy and a fixed pre-decision module. Under this approach, the system first reads k speech segments, each of which contains a fixed number (q, a hyper-parameter in the pre-decision module) of speech frames. When k speech segments have been read, the decoder generates one token in the target language. Similarly, we also apply the wait-k strategy in the decoding process of our end-to-end offline-ST system, as it strikes a good balance between translation quality and latency without requiring any streaming strategy during training (Papi et al., 2022; Polák et al., 2022). During inference, once a speech segment is accepted, the decoder takes the following action:

$$\textbf{Action} = \begin{cases} \text{continue to read} & |\mathbf{x}| - |\mathbf{y}| < k \\ \text{output } \mathbf{y}_t & |\mathbf{x}| - |\mathbf{y}| \geq k \end{cases} \qquad (2)$$

where $\mathbf{y}_t$ denotes the t-th token of the target language, while $|\mathbf{x}|$ and $|\mathbf{y}|$ refer to the number of source speech segments and target tokens, respectively.

3.3.2 ITST

The Information-Transport-based Simultaneous Translation (ITST) architecture has achieved state-of-the-art performance in end-to-end simultaneous speech translation. To implement this strategy, we initialize the corresponding parameters by using the pre-trained HuBERT and mBART models, and randomly initialize additional parameters for computing the information transport matrix. We then optimize the quality and latency objectives using the ITST criterion, varying the δ value to control the latency in streaming inference.

Our end-to-end speech translation system is built based on the ITST architecture, equipped with a wait-k streaming decoding strategy, and finally evaluated using the SimulEval (Ma et al., 2020a) toolkit. To ensure accurate translations, we enforce a constraint that the model should not produce the final translation until it has fully processed the speech in the source language.

3.4 Self-Training

Self-training is a simple semi-supervised learning method that involves using unlabeled data to augment labeled data (Pino et al., 2020; Sun et al., 2021; Wang et al., 2021; Popuri et al., 2022). To leverage the large-scale unlabeled audio introduced in Section 2.1, we employ self-training in our approach. In particular, we first train the end-to-end speech translation model on both manually annotated data and augmentation data, as described in Section 2. Next, we use the model to generate Chinese translation text, which we merge with the original training data and unlabeled audio. We then continue training the end-to-end speech translation model on this merged dataset.

3.5 Contrastive Learning

The objective of contrastive learning (Chen et al., 2020; Gao et al., 2021; Ye et al., 2022; Zhang et al., 2023) is to learn an encoder that produces similar representations for similar instances, while producing dissimilar representations for dissimilar instances, as measured by their cosine similarity. In our approach, we assume that the same utterance, regardless of whether it is in speech or text modality, will have similar hidden representations. Therefore, we aim to minimize the cosine distance between the hidden representations of the two modalities for the same utterance, while increasing the cosine distance between the hidden representations of different utterances. Specifically, we minimize the cosine distance between the speech encoder output and the corresponding word embedding for the same utterance, while maximizing the distance between the representations of different utterances. The training objective is as follows:

$$\mathcal{L}_{CTR} = \sum_{t=1}^{\boldsymbol{N}} \text{-} \log p \frac{\exp(sim(u, v)/T)}{\sum^{X} \exp(sim(u, v(x_j))/T)} \qquad (3)$$

where u is the average state of the speech encoder output along the sequence length, v is the average word embedding, and T is the temperature hyper-parameter. More specifically, $\mathcal{L}_{CTR}$ quantifies the negative logarithm of the probability that the similarity between u and v is greater than the similarity between u and other candidate word embeddings $v(x_j)$. The probabilities are normalized using a softmax function over all candidate embeddings. In addition to contrastive learning, we also conduct multitask learning using labeled ASR and MT training data, which results in the final optimization objective:

$$\mathcal{L} = \mathcal{L}_{ST} + \mathcal{L}_{ASR} + \mathcal{L}_{MT} + \mathcal{L}_{CTR} \tag{4}$$

where $\mathcal{L}_{ST}$, $\mathcal{L}_{ASR}$, $\mathcal{L}_{MT}$, and $\mathcal{L}_{CTR}$ denote the losses for speech-to-text translation, ASR, MT, and contrastive learning, respectively.

4 Experiments

4.1 Experiment Settings

The `fairseq` toolkit[14] is used to train our speech-to-text models. During training, the models take the original waveform sampled at 16kHz as the input. The Adam optimizer (Kingma and Ba, 2015) with a fixed learning rate of 5e-5 is used to train the models. Each model is trained for 200k steps, and we save the model every 2.5k steps using an early stopping mechanism. In detail, if the BLEU score on the development set does not improve for 10 consecutive checkpoints, the training will be terminated. During the fine-tuning stage, we set the maximum number of updates to 50k and the learning rate to 2e-5. Our TTS model is implemented using the Tacotron2 toolkit[15].

4.2 Evaluation

As the official automatic evaluation criterion, the BLEU score (Papineni et al., 2002) is used to evaluate the translation quality of all our systems. For the Simul-ST system, we employ the average lag (AL) (Ma et al., 2019, 2020b) metric to measure the translation latency, which is a standard metric for simultaneous speech translation. The SimulEval open-source toolkit[16] is utilized to calculate both the BLEU and AL metrics for the Simul-ST system. All BLEU scores are calculated with the SacreBLEU[17] (Post, 2018) toolkit at the character level.

	Models	BLEU
0	wav2vec2.0 (small)	23.84
1	HuBERT + mBART50 (one-to-many)	27.74
2	+ fine-tuning on MuST-C	27.90
3	+ Self-Training	27.69
4	+ Contrastive Learning	28.11
5	+ fine-tuning on MuST-C	27.94
6	data2vec + mBART50 (one-to-many)	27.66
7	+ fine-tuning on MuST-C	27.59
8	Ensemble $(2, 5)$	27.79
9	Ensemble $(2, 7)$	27.61
10	Ensemble $(2, 5, 7)$	27.94

Table 3: The BLEU scores of ST models on the tst-COMMON test set.

4.3 Main Results

Offline En⇒Zh Speech Translation

We evaluate our offline-ST models on the tst-COMMON test set by reporting the BLEU score in accordance with the official evaluation criteria. To establish a baseline for comparison, we use the widely-used standard wav2vec2.0 model for speech translation tasks. Table 3 shows the comparison results among all models. Our end-to-end models exhibit a significant improvement of approximately 4 BLEU points over the wav2vec2.0 baseline, which demonstrates the effectiveness of our methods. Additionally, we also conduct experiments using data2vec (Baevski et al., 2022) pre-trained model and obtain comparable results on the tst-COMMON test set.

By analyzing our experimental results, we observe that domain fine-tuning does not significantly improve the performance of the model. Nevertheless, we believe domain fine-tuning will be beneficial for final human evaluation on the TED[18] test set. Our final submission is an ensemble of the models listed in rows 2, 5, and 7 of Table 3.

It is worth mentioning that we encounter some challenges when training our model. When the HuBERT model is used to initialize our model, instabilities are observed during training, with sudden gradient explosions leading to training collapse. After careful analysis, we determine that the problem is that the gradients of the CNN layers are relatively large during the entire training process. We address this issue by scaling down the gradients of the CNN layers.

[14] `https://github.com/pytorch/fairseq`

[15] `https://github.com/NVIDIA/tacotron2`

[16] `https://github.com/facebookresearch/SimulEval`

[17] `https://github.com/mjpost/sacrebleu`

[18] `https://www.ted.com/`

	Models	BLEU
1	Offline-ST	30.10
2	Offline-ST + GigaST	31.56
3	Ensemble $(1, 2)$	31.81

Table 4: BLEU scores of our ST models on the development set of the S2ST track in IWSLT 2023. Offline-ST is trained on all manually annotated data and the augmented data described in Section 2.2.2. In addition to the data used by the offline-ST model, the Offline-ST + GigaST model incorporates additional GigaST data.

	Models	ASR-BLEU
1	Offline-ST	28.88
2	Offline-ST + GigaST	30.10
3	Ensemble $(1, 2)$	30.18

Table 5: ASR-BLEU scores of our ST models on the development set of the S2ST track in IWSLT 2023. The models are identical to those presented in Table 4.

Offline En⇒Zh Speech-to-Speech Translation

We evaluate the performance of our end-to-end speech-to-text translation system and cascaded speech-to-speech system on the development set of the S2ST track in IWSLT 2023, comprising $5,000$ utterances. The results of the speech-to-text translation models and speech-to-speech translation models are demonstrated in Table 4 and 5, respectively. For the speech-to-text translation model, we adopt the ensemble of models corresponding to rows 1 and 2 in Table 4. To build the speech-to-speech translation system, we then leverage our trained Chinese TTS model to synthesize Chinese speech and generate the corresponding Chinese transcript with the Conformer model [19] trained on the Wenet-Speech dataset (Zhang et al., 2022a). Finally, the generated Chinese transcript and reference are used to calculate the ASR-BLEU score.

Simultaneous En⇒Zh Speech Translation

We use the SimulEval toolkit to evaluate the quality and latency of our simultaneous speech translation model on the tst-COMMON set. In order to achieve a better balance between quality and latency, when the prediction probability is lower than 20%, the READ action is performed; when the delay exceeds 6000ms, the model performs a WRITE action to predict the next target token.

	Strategies	Models	BLEU	AL
1	Wait-k	HuBERT+mBART	25.99	1980
2	Wait-k	+ ST & CL	26.59	1966
3	ITST	HuBERT+mBART	26.25	1906

Table 6: The evaluation results of Simul-ST models on tst-COMMON. ST and CL denote self-training and contrastive learning for the Offline-ST model.

We evaluate the wait-k strategy using models 1 and 4 in Table 3, and train the ITST model with the same configuration as model 1 in Table 3. The results of the Simul-ST models are presented in Table 6. Although ITST shows better performance than wait-k in the same setting, the wait-k strategy combined with self-training and contrastive learning can achieve better results. Therefore, we finally submit the system corresponding to the second row in Table 6.

5 Conclusion

In this paper, we present our submissions for the IWSLT 2023 shared tasks. We participate in three tracks, namely the offline speech translation track, the offline speech-to-speech translation track, and the simultaneous speech translation track. All of our submissions use large-scale pre-trained models, and we further improve these models using various effective techniques, such as data augmentation, contrastive learning, and model ensembles. Extensive experiments validate the effectiveness of our proposed method and demonstrate that our submitted systems are comparable to state-of-the-art baseline systems in terms of performance.

Acknowledgements

Mengge Liu, Yanzhi Tian and Yuhang Guo have been supported in part by the National Key R&D Program of China (No. 2020AAA0106600). Jinsong Su has been supported by Youth Innovation Fund of Xiamen (No. 3502Z20206059).

[19] https://wenet-1256283475.cos.ap-shanghai.myqcloud.com/models/wenetspeech/wenetspeech_u2pp_conformer_exp.tar.gz

References

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 *Proc. of IWSLT*.

Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha Elbayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changhan Wang, and Matthew Wiesner. 2021. FINDINGS OF THE IWSLT 2021 EVALUATION CAMPAIGN. In *Proc. of IWSLT*.

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondřej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, Alexander Waibel, and Changhan Wang. 2020. FINDINGS OF THE IWSLT 2020 EVALUATION CAMPAIGN. In *Proc. of IWSLT*.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. data2vec: A general framework for self-supervised learning in speech, vision and language. In *Proc. of ICML*.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. of NIPS*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proc. of ICML*.

Liang Ding and Dacheng Tao. 2021. The USYD-JD speech translation system for IWSLT2021. In *Proc. of IWSLT*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proc. of EMNLP*.

Bao Guo, Mengge Liu, Wen Zhang, Hexuan Chen, Chang Mu, Xiang Li, Jianwei Cui, Bin Wang, and Yuhang Guo. 2022. The Xiaomi text-to-text simultaneous speech translation system for IWSLT 2022. In *Proc. of IWSLT*.

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. In *Proc. of TALSP*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR*.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, Juan Pino, and Wei-Ning Hsu. 2022. Direct speech-to-speech translation with discrete units. In *Proc. of ACL*.

Yinglu Li, Minghan Wang, Jiaxin Guo, Xiaosong Qiao, Yuxia Wang, Daimeng Wei, Chang Su, Yimeng Chen, Min Zhang, Shimin Tao, Hao Yang, and Ying Qin. 2022. The HW-TSC's offline speech translation system for IWSLT 2022 evaluation. In *Proc. of IWSLT*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. In *Proc. of TACL*.

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 *Proc. of ACL*.

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

Xutai Ma, Juan Pino, and Philipp Koehn. 2020b. SimulMT to SimulST: Adapting simultaneous text translation to end-to-end simultaneous speech translation. In *Proc. of AACL/IJCN*.

Sara Papi, Marco Gaido, Matteo Negri, and Marco Turchi. 2022. Does simultaneous speech translation need simultaneous models? In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 141–153. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*.

Juan Pino, Qiantong Xu, Xutai Ma, Mohammad Javad Dousti, and Yun Tang. 2020. Self-training for end-to-end speech translation. In *Proc. of Interspeech*.

Peter Polák, Ngoc-Quan Pham, Tuan-Nam Nguyen, Danni Liu, Carlos Mullov, Jan Niehues, Ondrej 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@ACL 2022, Dublin, Ireland (in-person and online), May 26-27, 2022*, pages 277–285. Association for Computational Linguistics.

Sravya Popuri, Peng-Jen Chen, Changhan Wang, Juan Pino, Yossi Adi, Jiatao Gu, Wei-Ning Hsu, and Ann Lee. 2022. Enhanced direct speech-to-speech translation using self-supervised pre-training and data augmentation. In *Proc. of Interspeech*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proc. of WMT*.

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, R. J. Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. 2018. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. In *Proc. of ICASSP*.

Matthias Sperber and Matthias Paulik. 2020. Speech translation and the end-to-end promise: Taking stock of where we are. In *Proc. of ACL*.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2021. Self-training for unsupervised neural machine translation in unbalanced training data scenarios. In *Proc. of NAACL*.

Yun Tang, Hongyu Gong, Ning Dong, Changhan Wang, Wei-Ning Hsu, Jiatao Gu, Alexei Baevski, Xian Li, Abdelrahman Mohamed, Michael Auli, and Juan Pino. 2022. Unified speech-text pre-training for speech translation and recognition. In *Proc. of ACL*.

Ioannis Tsiamas, Gerard I. Gállego, José A. R. Fonollosa, and Marta R. Costa-jussà. 2022. SHAS: approaching optimal segmentation for end-to-end speech translation. In *Proc. of Interspeech*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of NIPS*.

Changhan Wang, Anne Wu, Juan Pino, Alexei Baevski, Michael Auli, and Alexis Conneau. 2021. Large-scale self- and semi-supervised learning for speech translation. In *Proc. of Interspeech*.

Rong Ye, Mingxuan Wang, and Lei Li. 2022. Cross-modal contrastive learning for speech translation. In *Proc. of NAACL*.

Binbin Zhang, Hang Lv, Pengcheng Guo, Qijie Shao, Chao Yang, Lei Xie, Xin Xu, Hui Bu, Xiaoyu Chen, Chenchen Zeng, Di Wu, and Zhendong Peng. 2022a. Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition. In *Proc. of ICASSP*.

Hao Zhang, Nianwen Si, Yaqi Chen, Wenlin Zhang, Xukui Yang, Dan Qu, and Wei-Qiang Zhang. 2023. Improving speech translation by cross-modal multi-grained contrastive learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Shaolei Zhang and Yang Feng. 2022. Information-transport-based policy for simultaneous translation. In *Proc. of EMNLP*.

Weitai Zhang, Zhongyi Ye, Haitao Tang, Xiaoxi Li, Xinyuan Zhou, Jing Yang, Jianwei Cui, Pan Deng, Mohan Shi, Yifan Song, Dan Liu, Junhua Liu, and Lirong Dai. 2022b. The USTC-NELSLIP offline speech translation systems for IWSLT 2022. In *Proc. of IWSLT*.

Ziqiang Zhang and Junyi Ao. 2022. The YiTrans speech translation system for IWSLT 2022 offline shared task. In *Proc. of IWSLT*.

Qinpei Zhu, Renshou Wu, Guangfeng Liu, Xinyu Zhu, Xingyu Chen, Yang Zhou, Qingliang Miao, Rui Wang, and Kai Yu. 2022. The AISP-SJTU simultaneous translation system for IWSLT 2022. In *Proc. of IWSLT*.