BIT-Xiaomi's Simultaneous Translation System for AutoSimTrans 2022

Mengge Liu[1], Xiang Li[2], Bao Chen[1], Yanzhi Tian[1], Tianwei Lan[1]

Silin Li[1], Yuhang Guo[1], Jian Luan[2], Bin Wang[2]

[1]Beijing Institute of Technology, Beijing, China

[2]Xiaomi AI Lab, Beijing, China

Content

- Introduction
- Data Filtering & Selection
- Data Augmentation
- Streaming Chunking
- Experiment Results

Introduction

Simultaneous translation

- Machine Translation with **low latency**.
- Need to start translating before the end of sentences.

Training methods

- Data filtering, selection, and augmentation.
- Pre-training and fine-tuning.
- Wait-k training.
- R-Drop training.

Inference methods

- **Wait-k strategy**: Fixed-latency policy, keep the output sentence lagging k words behind the input sentence.
- **Chunking strategy**: Adaptive-latency policy, detect segmentation boundaries in source sentences and start translating at boundaries.
- **Streaming Chunking**: Combine wait-k and chunking strategies.

Data Filter & Selection

- Filter noise data of the BSTC(speech domain) and the CWMT(news domain) corpus.
- Select speech domain data of the CWMT corpus.
- **D1** corpus for pre-training.

Pre-training (data)	Data statistic	dev (sacreBLEU)
Orig BSTC+CWMT (D0)	9.1M	16.82
+rules-filter	7.7M	18.09
+align-langid-filter	7.2M	18.04
+PPL-selection (D1)	6.2M	17.99

Table 3: Data filtering and selection in the pre-training stage. BLEU is computed by sacreBLEU in sentence-level. Filtering and selection methods are applied incrementally.

- rules-filter: Hand-crafted rules and deduplication.
- align-langID-filter: Filter noise data with *fast_align* and *langID*.
- PPL-selection: Select by PPL(Perplexity) scores with *KenLM*.

Data Filter & Selection

- **R-Drop** and big dropout enhance the performance.
- **Up-sampling speech domain data** is confirmed to be beneficial.

Pre-training (method)	Data statistic	dev (sacreBLEU)
BSTC+CWMT (D1)	6.2M	17.99
+up-sampling	6.34M	18.40
+dropout 0.25	6.2M	18.59
+R-Drop ($\alpha = 5$)	6.2M	19.72
+up-sampling + dropout 0.25 + R-Drop	6.34M	21.48

Table 4: Data statistic and BLEU on the development of our pre-training methods. BLEU is computed by sacreBLEU in sentence-level.

- up-sampling: Up-sampling BSTC corpus 5 times.
- dropout 0.25: Dropout value set as 0.25 (Default 0.1).
- R-Drop: Utilize R-Drop training method.

Data Augmentation

- **Pre-train** on CWMT and BSTC corpus.
- Forward translation augmentation performs better in pre-training.

Pre-training (Augmentation)	Data statistic (Pre-training)	dev (sacreBLEU)
M1 (only pre-train)	6.34M	21.48
+FT pre-train	10.95M	22.32
+BT pre-train	11.03M	19.90

Table 5: Results of data augmentation in the pre-training stage. We use the M1 model to generate the FT and BT augment data and mixed with the D1 corpus for pre-training.

- FT pre-train: Augment pretrain corpus with Forword-Translation.
- BT pre-train: Augment pretrain corpus with Backword-Translation.

Data Augmentation

- **Fine-tune** on BSTC corpus.
- Forward translation, backward translation, and character-level augmentation are all beneficial in fine-tuning.

Fine-tuning (Augmentation)	Data statistic (Fine-tuning)	dev (sacreBLEU)
M1 (fine-tuned on BSTC)	36K	22.41
+5FT	197K	22.92
+5BT	211K	22.59
+char-aug	185K	22.80
+5BT +5FT +char-aug	525K	23.05

Table 6: Results of data augmentation in the fine-tuning stage. The M1 model is leveraged to generate FT and BT augment data, and beam 5 results are saved. For the char-aug, we use character-level augmentations including insertion, deletion, duplication, and homophone substitution. The models in this table are all based on the same pre-trained model.

- FT pre-train: Augment pretrain corpus with Forword-Translation.
- BT pre-train: Augment pretrain corpus with Backword-Translation.
- char-aug: Character-level augmentation for Chinese sentences.

Streaming Chunking

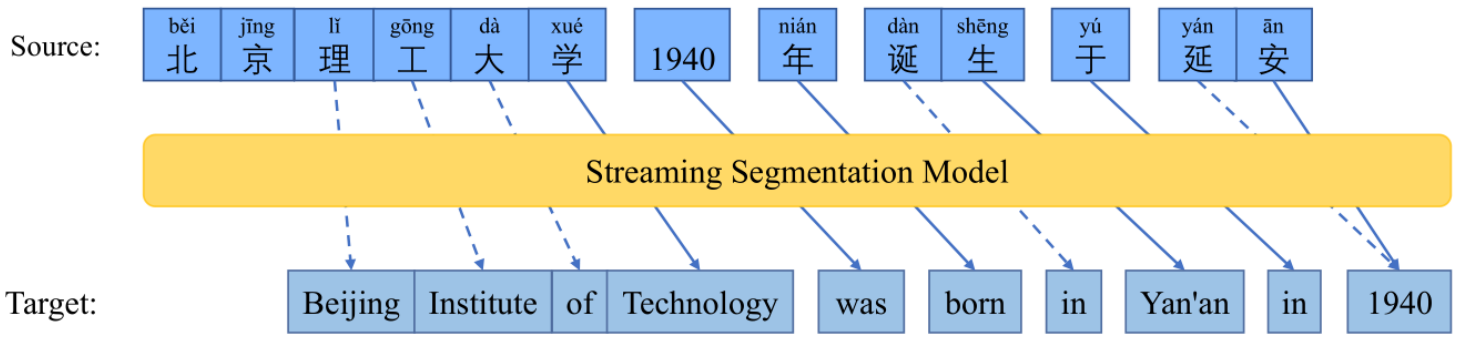
- Example of error word in character-stream.

stream-id	char-stream	word-stream
1	那	那
2	那首	那
3	那首先	那首先
4	那首先呢	那首先呢
5	那首先呢我	那首先呢我
6	那首先呢我先	那首先呢我先
7	那首先呢我先介	那首先呢我先
8	那首先呢我先介绍	那首先呢我先介绍
9	那首先呢我先介绍一	那首先呢我先介绍
10	那首先呢我先介绍一下	那首先呢我先介绍一下
11	那首先呢我先介绍一下我	那首先呢我先介绍一下我
12	那首先呢我先介绍一下我自	那首先呢我先介绍一下我
13	那首先呢我先介绍一下我自己	那首先呢我先介绍一下我自己
full sentence	nà | shǒu xiān | nē | wǒ | xiān | jiè shào | yí xià | wǒ zì jǐ 那 | 首先 | 呢 | 我 | 先 | 介绍 | 一下 | 我自己 then | first | - | I | - | introduce | - | myself	

Table 2: Case analysis of incomplete streaming in a Chinese sentence. Char-stream presents sentences by characters. Word-stream presents sentence by word. The prefixes in red color mean error in char-stream, which contains incomplete word-piece. The partial word piece may cause misunderstanding and incorrect translation.

Streaming Chunking

- **Wait-k** training:
  - Pre-train with multi-path wait-k.
  - Fine-tune with simple wait-k.
- **Segmentation** Model:
  - Based on Chinese Bert.
  - Finetune with word boundaries segmented by *jieba*.
  - Used to detect word boundaries.
- **Streaming Chunking** Inference:
  - Set k like wait-k inference.
  - Perform translation only at word boundaries detected by the segmentation model.



Experiment Results

En-Zh text-to-text track

- The ensemble model shows the best performance.
- The streaming chunking method further improves the performance.

Model	dev (SacreBleu)	dev (Mteval-v13a)
M1	22.43	27.26
M2	23.62	28.96

Table 7: Results of data augmentation on standard transformer model. The M1 model is trained with pre-training and fine-tuning. The M2 model leverage data augmentation in both the pre-training and the fine-tuning stage.

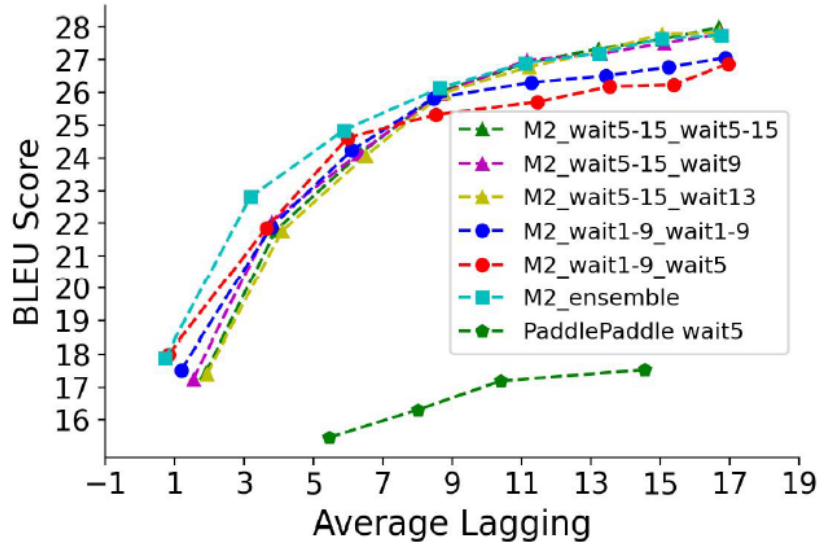


Figure 2: Results of M2 wait-k models. Models are list in Table 8. PaddlePaddle_wait5 is wait-k model provided by organizer.

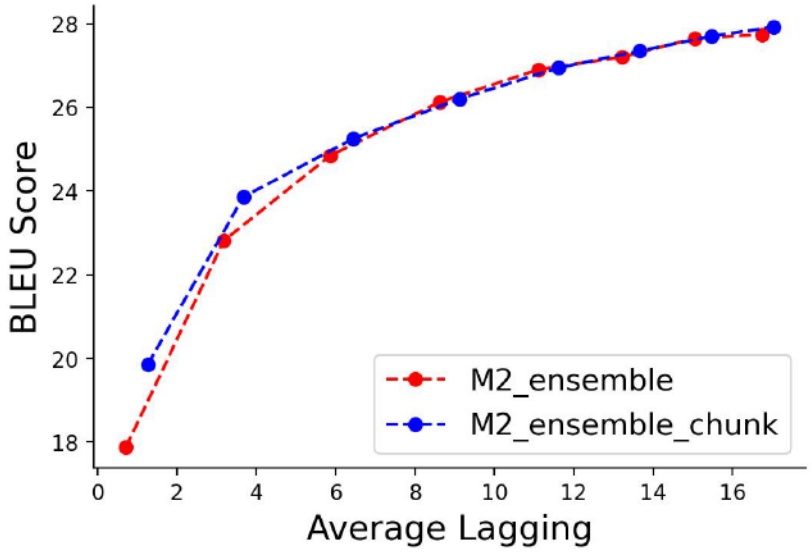


Figure 3: Results of streaming chunking method. M2_ensemble_chunk add streaming segmentation model compare to M2_ensemble.

Experiment Results

En-Zh audio-to-text track

- Performance is influenced by ASR model.

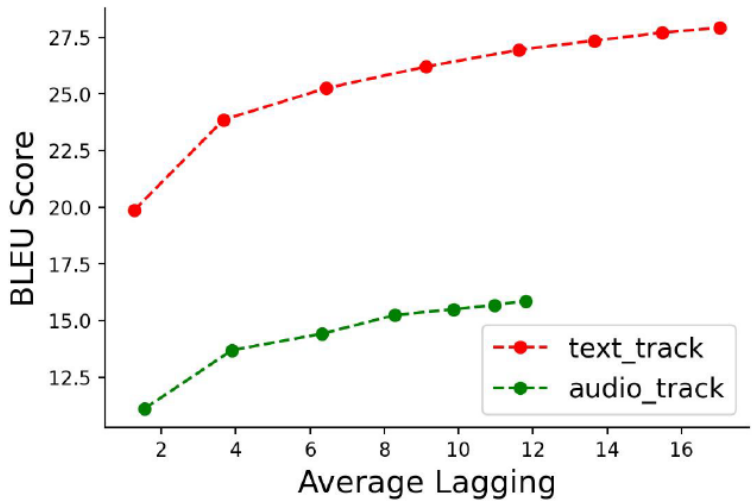


Figure 5: Results of Zh-En audio-to-text track. BLEU is computed in document level with Mteval-v13a.

En-Es text-to-text track

- Wait-k training and inference.

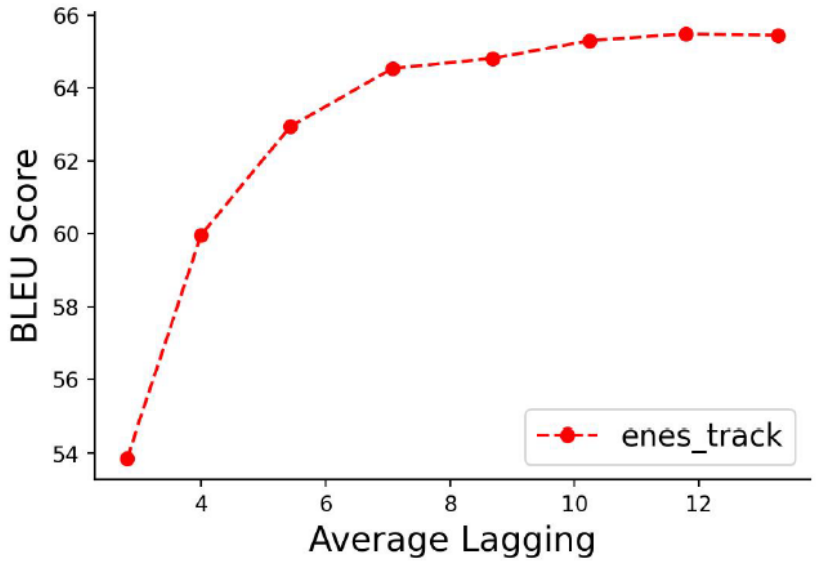


Figure 4: Results of En-Es text-to-text track. BLEU is computed in document level with Mteval-v13a.

Thanks!