BIT's System for Multilingual Track

Zhipeng Wang
Beijing Institute of Technology
wzp13972518l@163.com

Yuhang Guo*
Beijing Institute of Technology
guoyuhang@bit.edu.cn

Shuoying Chen
Beijing Institute of Technology
chensy@bit.edu.cn

Abstract

This paper describes the system we submitted to the IWSLT 2023 multilingual speech translation track, with the input is speech from one language, and the output is text from 10 target languages. Our system consists of CNN and Transformer, convolutional neural networks downsample speech features and extract local information, while transformer extract global features and output the final results. In our system, we use speech recognition tasks to pre-train encoder parameters, and then use speech translation corpus to train the multilingual speech translation model. We have also adopted other methods to optimize the model, such as data augmentation, model ensemble, etc. Our system can obtain satisfactory results on test sets of 10 languages in the MUST-C corpus.

1 Introduction

Speech translation refers to the technology of translating source language speech into target language text (or speech). This task has a very broad application space in real life, such as in international conferences, lectures, and overseas tourism; Adding speech translation to short videos or real-time subtitles in some foreign language videos can provide users with a better experience. Early speech translation is the combination of speech recognition and machine translation. Firstly, the speech recognition model recognizes source language speech as source language transcribed text, and then the machine translation model translates the recognized source language text into the target language text, which is also called cascade method. The advantage of cascade model is that it can use a large amount of data in speech recognition and machine translation to train the model, and it is relatively simple to implement. However, the disadvantages of cascade model are also obvious: errors in speech recognition results will be transferred to the next machine translation task. So researchers have focused on end-to-end speech translation. At present, bilingual end-to-end speech translation has achieved very good results, but using a single model to complete multiple language translations has always been a goal pursued by researchers, that is multilingual speech translation. Compared to bilingual speech translation, the advantages of multilingual speech translation include: (1) completing multilingual translation with fewer parameters; (2) low resource languages can learn knowledge from high resource languages. In this paper, we conducted one-to-many multilingual speech translation, and submitted our system to the IWSLT 2023(Agarwal et al., 2023) multilingual speech translation track. Here is an introduction to our submitted system:

We first use convolutional neural networks to downsample the input features, then input them into the Transformer model for further processing, and finally output the translation results at the output layer. The encoder for speech translation needs to complete both acoustic feature extraction and semantic feature extraction tasks. In order to reduce the encoding pressure of the model, we use speech recognition task to pre-train the parameters of the encoder. Before inputting the data into the model, we applied the SpecAugment(Park et al., 2019) method for data augmentation, which increased data diversity and resulted in better results for the model. After training the multilingual speech translation model, we calculated the average value of the model parameters obtained for the last 10 epochs to generate the model we used during testing, the model with the obtained average parameters can have better results.

The target language includes Arabic, Chinese, Dutch, French, German, Japanese, Farsi, Portuguese, Russian, and Turkish. The training data for these languages can be found in the commonly used corpus for speech translation – MUST-C(Di Gangi et al., 2019). We downloaded the

*Corresponding author

Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023), pages 455–460
July 13-14, 2023 ©2023 Association for Computational Linguistics

Table 1: Training Set Information

Tgt	talks	sentences	time	words src	words tgt
ar	2412	212k	463h	4520k	4000k
de	2043	229k	400h	4196k	3869k
fa	1911	181k	347h	3548k	4559k
fr	2460	275k	484h	5067k	5163k
ja	3258	328k	540h	5712k	69k
nl	2219	248k	434h	4548k	4251k
pt	2001	206k	376h	3887k	3621k
ru	2448	265k	481h	5007k	4192k
tr	2307	236k	445h	4600k	3388k
zh	3583	358k	596h	6251k	97k

data for the relevant languages from MUST-C v1.0, MUST-C v1.2, and MUST-C v2.0, merged them, and preprocessed them to obtain our training dataset. We used the Fairseq(Ott et al., 2019) toolkit to conduct our experiment, and after the training was completed, we scored the translation quality using the sacrebleu metric. Our model achieved our expected results on 10 target languages.

2 Data Preparation

As shown in the Table 1, we collected training data for relevant languages from the MUST-C corpus and provided their information. It can be seen from this that there are significant differences between different languages. There are differences in the number of source language words and target language words among different languages. For example, the number of source language words in the Arabic language corpus is greater than the number of target language words, while the number of source language words in the Farsi language corpus is less than the number of target language words. This indicates that the difficulty of length conversion required by the model when dealing with different languages varies to some extent.

Due to our task of one-to-many multilingual speech translation, the input received by the model is all English speech data, which enables us to perform the same preprocessing operation on all data. The original speech is in wav format, and most of it is long audio. We need to segment and extract features before inputting it into the model. So we segment the speech data based on the start time and duration of each segment given in MUST-C. The preprocessing stage includes extracting MFCC features, training the sentencepeice(Kudo and Richardson, 2018) model, generating a vocabulary, and finally generating a training set. The processed MFCC feature dimension is 80, and SpecAugment is applied for data augmentation. The relevant configurations used in the experiment regarding SpecAugment are shown in Table 2:

Table 2: Parameter settings for SpecAugment

Parameters	Values
freq_mask_F	27
freq_mask_N	2
time_mask_N	2
time_mask_T	100
time_mask_p	1.0
time_wrap_W	0

The SpecAugment method uses three different data augmentation methods: Time warping, Frequency masking, and Time masking. Time warping selects an area from the time dimension for warping. Frequency masking selects an area from the frequency dimension for masking, in our experimental configuration, the length of the masked part is 27, which is the parameter freq_mask_F, and the parameter freq_mask_N refers to the number of masked areas. Time masking selects an area from the time dimension for masking, the parameter time_mask_T we set is 100, and the number of masked areas is 2. SpecAugment increases the diversity of training data, making the trained model more robust.

3 Method

3.1 Speech Recognition

We use speech recognition tasks to pre train encoder parameters. After experimental verification, using speech recognition for pre training parameters is much better than not using pre-training. Due to the need to initialize the parameters of the speech translation model using the encoder of the speech recognition model, we use the same structure to train the speech recognition model. Although extracting MFCC features from the original audio can reduce the sequence length, the processed MFCC features still have a long time dimension and require further downsampling. In speech translation related works, a common practice is to use CNN or Shrink modules(Liu et al., 2020) to compress feature sequences. We use convolutional neural networks to downsample the extracted MFCC feature sequence, the input MFCC features are first extracted through a two-layer convolutional neural network to extract shallow features and downsampling, and then input into the Transformer model to complete the speech recognition task. The model structure is shown in the Figure 1. The reason why Transformer has strong modeling information ability is due to its self attention mechanism, the multi-head attention calculation in transformer is shown in the Figure 2. Perform different linear calculations on the input to obtain Q, K, and V. compute the matrix of outputs as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

Each module in Transformer has its specific role, and the following is an analysis of its main modules:

Multi-head attention module. Self attention refers to calculating the attention of the current token to other tokens in the sequence, and using the calculated attention score as a weight to weight and sum the feature sequence, thus modeling global information. The final output of the multi-head self attention module is obtained by concatenating the results obtained from all the attention heads and then performing a linear mapping.

Feed forward module. In the feed forward module, the extracted global features are linearly combined, which includes two linear mappings: mapping feature sequences to high dimensions and mapping features from high dimensions back to their original dimensions, the calculation in the feed forward module is as follows:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \quad (2)$$

Positional Encoding. The transformer uses position encoding to indicate the relative position between tokens, and the calculation method is as follows:

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}}) \quad (3)$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}}) \quad (4)$$

After extracting shallow features from speech using convolutional neural networks, transformer combines the extracted information. Convolutional neural networks are good at extracting local features, while transformer have a stronger ability to model global features. This structure enables the model to perform well in several speech processing tasks.

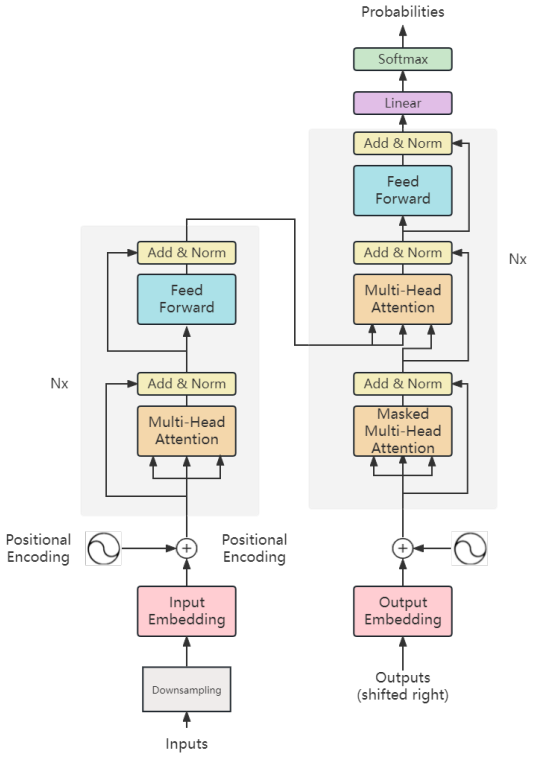

Figure 1: Model structure

3.2 Multilingual Speech Translation

The multilingual speech translation model also adopts the structure shown in the Figure 1, replacing the speech recognition vocabulary with the multilingual speech translation vocabulary, and training

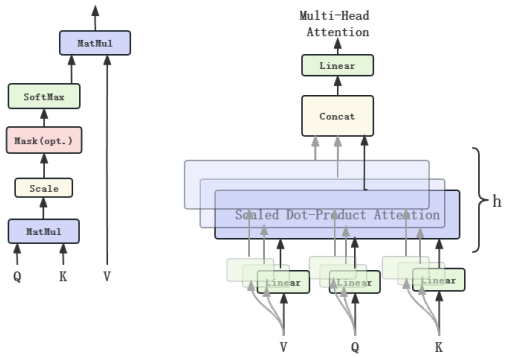

Figure 2: Multi-head Attention

the model using the speech translation training set. Unlike speech recognition task, the vocabulary in multilingual speech translation task contains language labels, and the sub words in the dictionary come from all target language texts. Before training the speech translation model, use the encoder of the trained speech recognition model to initialize the encoder parameters of the speech translation model, and optimize all model parameters during training.

We only use the encoder of the speech recognition model to initialize the multilingual speech translation model because the tasks completed by the two encoders are similar, that is, the shallow layer of the encoder needs to extract acoustic feature information. However, there are task differences between speech recognition and speech translation decoders. The speech translation decoder needs to complete language conversion, and the speech recognition decoder does not involve this task, so the speech recognition decoder is not used to initialize parameters. The speech recognition model will no longer be used in subsequent operations.

The decoder adopts an auto-regressive approach to output the translation sequence, and in this experiment, language labels are used to indicate the current translation direction. For example, <lang: de> indicates that the target language of the current translation task is German.

We conducted parameter fusion on the trained model. After the model was trained to converge, the last 10 checkpoint points were fused and the test set was scored by the fused model. The specific approach is to find variables with the same name from all the models read, calculate the average value, and save it in the new model.

4 Experiments

4.1 Implemention

The downsampling module contains two layers of convolutional neural networks, with convolutional kernel sizes of 5 and step sizes of 2. After the feature sequence passes through the downsampling layer, the sequence length becomes one quarter of the original, The dimension of the output feature is 1024.

The encoder of the model contains 12 transformer blocks, with each layer having an output feature dimension of 512. In order to fully model speech features, 8 attention heads were used to model information in speech from different perspectives. The feedforward neural network module contains two linear maps to reorganize the features. First, the feature dimension is mapped to 2048, and then it is mapped back to 512.

The decoder of the model consists of 6 Transformer blocks, which also use 8 attention heads.

In addition, we use the dropout of the attention matrix to prevent overfitting. The dropout rate of attention is set to 0.1. In speech recognition tasks, we set the vocabulary size to 8000; In the speech translation task, we set the vocabulary size to 10000. Because the speech recognition task only involves English text, while the speech translation task involves translated text from 10 target languages, a larger vocabulary needs to be used. At the time of model output, the probability of speech recognition task computing on 8000 sub words and the probability of speech translation task computing on 10000 sub words.

Adam optimizer and cross entropy loss function are used in model training. We use max tokens to dynamically control the number of samples included in a batch. In our experiment, the max tokens used for both speech recognition and speech translation tasks were 20000. The number of steps for optimizing speech recognition tasks is 100k, and the number of steps for optimizing speech translation tasks is 350k, based on the difficulty of these two tasks. Among them, perform warmup in the first 10k steps. The learning rate is 1e-3, and the label smoothing is 0.1. We trained our model using two NVIDIA TITAN RTX.

4.2 Main Results

We trained a speech recognition model with good performance, and the WER of the model on each language is shown in the Table 3.

Table 3: The WER of the ASR model on data in each language.

	ar	de	fa	fr	ja	nl	pt	ru	tr	zh
WER	16.01	10.64	11.65	10.74	8.79	10.43	10.76	10.71	11.10	8.80

Table 4: BLEU scores on the MUST-C test set.

	ar	de	fa	fr	ja	nl	pt	ru	tr	zh
BLEU	12.35	23.30	12.15	32.59	12.93	27.46	28.57	14.66	11.33	22.07

After training the model on the MUST-C training set, we used its tst-COMMON test set to verify the model's effectiveness. The experimental results are shown in the Table 4.

From the Table 4, it can be seen that our system can complete translations in these 10 target languages, and the BLEU score exceeds 20 in all 5 languages of them. Although using the same model for translation tasks, the difficulty of translation varies among different languages. As shown in the table, the BLEU scores of ar, fa, ja, ru, and tr are lower compared to other languages, but they use a similar amount of data. On the one hand, there are significant differences in grammar rules between these target languages and the source language, making it more difficult for the model to complete language conversion; On the other hand, the differences between target languages make it difficult to share information between them consistently.

In the current work of multilingual speech translation, many methods have modified the model architecture and optimization methods, and our system uses a simple convolutional neural network combined with the Transformer structure to achieve a relatively good effect. Compared to those complex systems that modify models, our system has the following advantages: On the one hand, our system's training method is relatively simple and requires fewer model parameters. On the other hand, this simple structure can also effectively complete multilingual speech translation tasks. Our system can be applied to devices with strict memory requirements, and can achieve relatively satisfactory results with a small number of parameters.

5 Conclusion

This paper introduces our system submitted on the IWSLT 2023 multilingual speech translation track. We used convolutional neural networks combined with Transformer models to complete the task of English speech to 10 target language texts. Our system is characterized by its simplicity and efficiency, effectively modeling local and global features in speech, and completing modal and language transformations within the model. Our system has achieved satisfactory results on the test set of 10 languages in MUST-C corpus.

References

Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Yannick Estève, Kevin Duh, Marcello Federico, Souhir Gahbiche, Barry Haddow, Benjamin Hsu, Phu Mon Htut, Hirofumi Inaguma, Dávid Javorský, John Judge, Yasumasa Kano, Tom Ko, Rishu Kumar, Pengwei Li, Xutail Ma, Prashant Mathur, Evgeny Matusov, Paul McNamee, John P. McCrae, Kenton Murray, Maria Nadejde, Satoshi Nakamura, Matteo Negri, Ha Nguyen, Jan Niehues, Xing Niu, Atul Ojha Kr., John E. Ortega, Proyag Pal, Juan Pino, Lonneke van der Plas, Peter Polák, Elijah Rippeth, Elizabeth Salesky, Jiatong Shi, Matthias Sperber, Sebastian Stüker, Katsuhito Sudoh, Yun Tang, Brian Thompson, Kevin Tran, Marco Turchi, Alex Waibel, Mingxuan Wang, Shinji Watanabe, and Rodolfo Zevallos. 2023. Findings of the IWSLT 2023 Evaluation Campaign. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*. Association for Computational Linguistics.

Mattia A Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2019. Must-c: a multilingual speech translation corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2012–2017. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tok-

enizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Yuchen Liu, Junnan Zhu, Jiajun Zhang, and Chengqing Zong. 2020. Bridging the modality gap for speech-to-text translation. *arXiv preprint arXiv:2010.14920*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.