ISTIC's Triangular Machine Translation System for WMT' 2021

Hangcheng Guo, Wenbin Liu, Yanqing He*, Tian Lan,
Hongjiao Xu, Zhenfeng Wu, You Pan
Institute of Scientific and Technical Information of China, Beijing, China
{guohc2020, liuwb2019, heyq, lantian,
xuhj, wuzf, pany}@istic.ac.cn

Abstract

This paper describes the ISTIC's submission to the Triangular Machine Translation Task of Russian-to-Chinese machine translation for WMT' 2021. In order to fully utilize the provided corpora and promote the translation performance from Russian to Chinese, the pivot method is used in our system which pipelines the Russian-to-English translator and the English-to-Chinese translator to form a Russian-to-Chinese translator. Our system is based on the Transformer architecture and several effective strategies are adopted to improve the quality of translation, including corpus filtering, data pre-processing, system combination, model averaging, model ensemble and reranking.

1 Introduction

The Institute of Scientific and Technical Information of China (ISTIC) participated in the Triangular Machine Translation Task of Russian-to-Chinese in the Sixth Conference on Machine Translation[1] (WMT' 2021). This paper demonstrates the overall framework of the ISTIC's submission and its technical details.

In this evaluation, we adopted the neural machine translation architecture of Google Transformer(Vaswani et al., 2017) as a part of our system. We use the three parallel corpora released by the evaluation organizer and adopted a two-stage method for data pre-processing. Several filtering methods of the corpus are explored to reduce the data noise and improve the data quality. As for model construction, we use the pivot method to get a Russian-to-Chinese translator by bridging the trained Russian-to-English translator and English-to-Chinese translator. Model averaging(Claeskens and Hjort, 2008), model ensemble(Lutellier et al., 2020) and reranking(Ng et al., 2019) strategies are adopted to generate the final output translation. We removed spaces between words and restored the target language translation results to the prescribed file format in data post-processing. In our experiment, the performance of the system under different settings was compared and further analyzed the experimental results.

The structure of this paper is as follows: the second part introduces the technical architecture of our machine translation system; the third part describes the data pre-processing, parameter settings, experimental results, and related analysis; the fourth part gives the conclusion and future work.

2 System Overview

The overall framework of the ISTIC's triangular machine translation system is shown in Figure 1.

2.1 Single Transformer System

Our baseline single system used in participated evaluation tasks is the Transformer based encoder-decoder architecture. Transformer is completely based on a self-attention mechanism. It can achieve algorithm parallelism, speed up model training, further alleviate long-distance dependence and improve translation quality(Zhang and Zong, 2020). The encoder and the decoder are formed by stacking N identical layer blocks, where N is set to 6.

2.2 Context-based Combination System

As shown in Figure 2, based on the Transformer model, our team adopts a context-based(Voita et al., 2018) system combination method, which is an encoder-decoder structure composed of n identical network layers, where n is set to 6. Two different methods of system combination are designed according to the fusion in different positions, which are Encoder Combination method and Decoder Combination method. Both of them adopt multi-encoder(Li et al., 2020) to encode the source sen-

*Corresponding author: Yanqing He, heyq@istic.ac.cn.

[1] http://www.statmt.org/wmt21/triangular-mt-task.html

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 320–324
November 10–11, 2021.

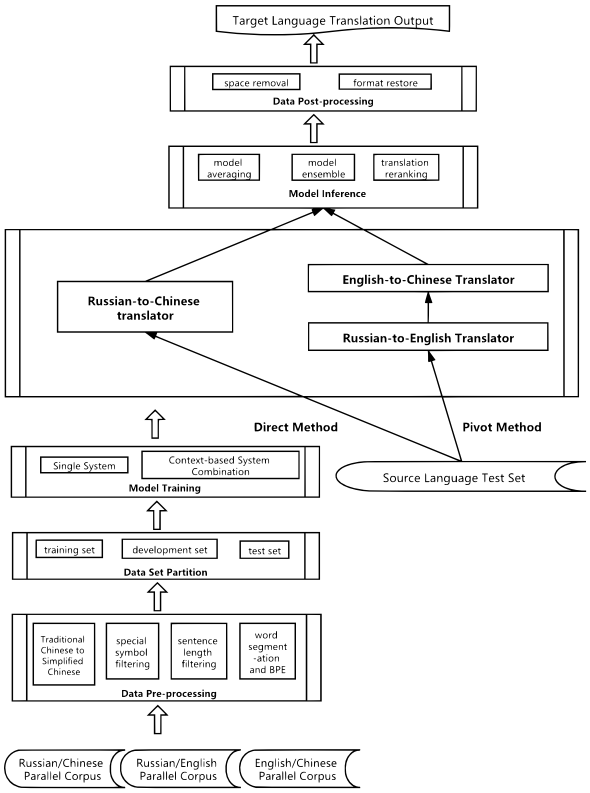

Figure 1: Overall framework

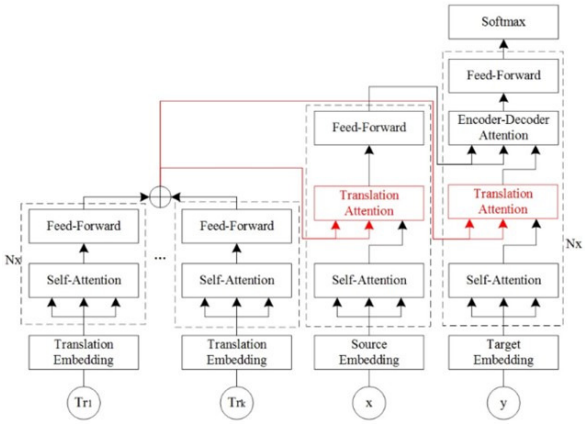

Figure 2: Context-based combination system

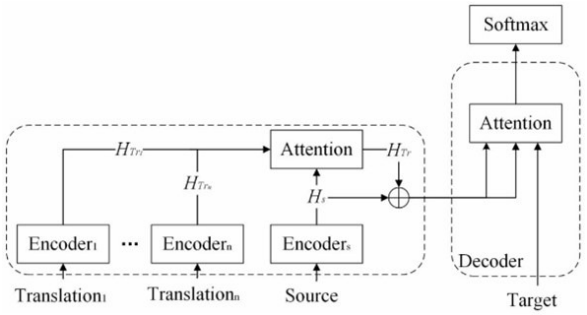

Figure 3: Encoder combination model

tences and the context information from machine translation results of the source sentence. In the Encoder Combination method, the hidden layer information of context (multi-system translation) is transformed into new representation through attention network, and merges the hidden layer information of source sentence through gating mechanism at encoder end; In Decoder Combination method, the hidden layer information of multi-system translation and the hidden layer information of source sentence is calculated at the decoder to obtain the fusion vector. The attention calculation method is the same as the original transformer model, to obtain a higher quality fusion translation.

The Encoder Combination model (see Figure 3) uses multiple system translations, and then converts the system translations into new representations through the attention network, integrating the hidden layer information of homologous language sentences for attention fusion through the gating mechanism in the Encoder. In the Encoder Combination mode and the Self-Attention of the multi-system translation Encoder, Q, K, and V are all from the upper layer output of the multi-system translation Encoder; in the Self-Attention of the source language Encoder, Q, K, and V are all from the upper layer output of the source language Encoder; in the Translation Attention of the source language Encoder, both K and V come from the upper hidden layer state H_{T_r} of the multi-system translation Encoder, and Q comes from the upper layer hidden state H_s of the source language Encoder. H_s represents the hidden state of the source language sentence, H_{T_r} represents the hidden state of the multi-system translation, and H represents the hidden state of the Translation Attention part of the Encoder.

$$H_{T_r} = Concat(H_{T_r1}, ..., H_{T_rn}) \quad (1)$$

$$H = MutiHead(H_{T_r}, H_s) \quad (2)$$

The Decoder Combination model (see Figure 4) combines the hidden layer information of multiple encoders with attention in the decoder. The Decoder can process multiple encoders separately, and then fuse them using the gating mechanism inside the Decoder to obtain the combined vector. In the Decoder Combination mode and the Self-Attention of the target language Decoder, Q, K, and V are all from the output of the previous layer of the target language Decoder; in the Translation Attention of the target language Decoder, Q comes from the

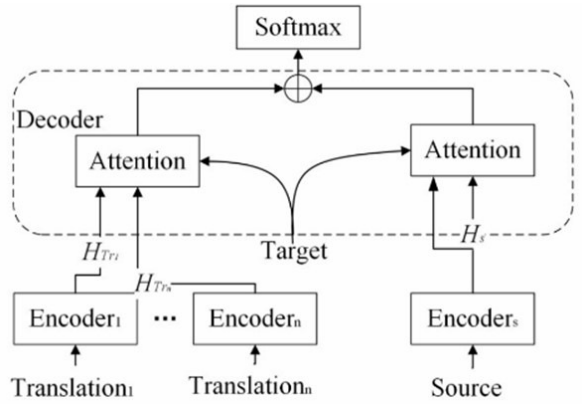

Figure 4: Decoder combination model

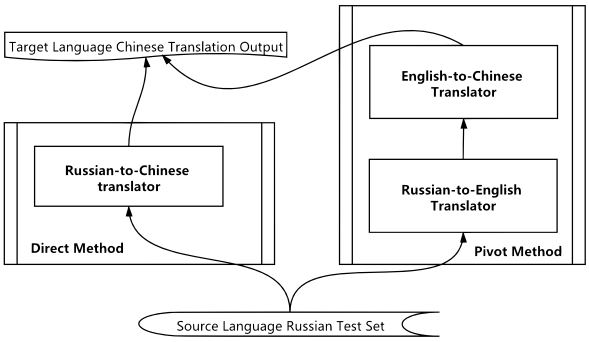

Figure 5: Direct and pivot method

output of the upper layer of the target language Decoder, K comes from the upper hidden layer state H_s of the source language Encoder, and V comes from the upper hidden layer state H_{T_r} of the multi-system translation Encoder; in the Encoder-Decoder Attention of the target language Decoder, Q comes from the upper layer output of the target language Decoder, K, V come from the previous output of the source language Encoder. H_s represents the hidden layer state of the source language sentence, H_{T_r} represents the hidden layer state of the multi-system translation, $H_{Decoder}$ represents the hidden layer state of the upper layer output of the Decoder, and H represents the hidden state of the Translation Attention part of the Decoder.

$$H = MutiHead(H_{T_r}, H_s, H_{Decoder}) \quad (3)$$

2.3 Direct Method

In the direct method (see Figure 5), we use the pre-processed Russian/Chinese parallel corpus to train a direct Russian-to-Chinese translator by means of the single Transformer System or the context-based Combination System, depending on which kind of system performs best.

2.4 Pivot Method

In the pivot method (see Figure 5)(Park and Zhao, 2019), firstly, we use the pre-processed Russian/English parallel corpus to train a Russian-to-English translator; secondly, we use the pre-processed English/Chinese parallel corpus to train an English-to-Chinese translator; finally, we pipeline them to form a pivot Russian-to-Chinese translator. All translators can be trained by means of the single Transformer System or the context-based Combination System. By comparing the experimental results, the system with optimal performance is accepted for Russian-to-Chinese translation.

3 Experiments

3.1 Data Pre-processing

The evaluation organizers provide three parallel corpora: the Chinese/Russian corpus is crawled from the web and aligned at the segment level, and combined with different public resources; the Chinese/English corpus combines several public resources; the Russian/English corpus gathers multiple public resources. A two-stage method(Wei et al., 2020) is used for data pre-processing, consist of a general pre-processing stage and a specific pre-processing stage. The general pre-processing stage includes conversion from traditional Chinese to simplified Chinese by the hanziconv[2] package , conversion between full angle and half-angle, special character filtering, same content filtering, sentence length filtering, and sentence length ratio filtering. Among them, sentence length of the Chinese language is calculated in the unit of "character" and sentence length of non-Chinese language is calculated in the unit of "token". Sentence length filtering removes sentence pairs which source sentence length or target sentence length exceeds the range of [5, 200]. Sentence length ratio filtering excludes the sentence pairs whose ratio of source sentence length and target sentence length exceeds the range of [0.2, 20]. In the specific pre-processing stage, the word segmentation of English and Chinese sentences is implemented using the lexical tool Urheen[3] and the word segmentation of Russian sentences is implemented using the lexical

[2] https://github.com/berniey/hanziconv

[3] https://www.nlpr.ia.ac.cn/cip/software.html

Direction	Before Pre-processing	After Pre-processing
Russian-English	69217438	42939395
English-Chinese	28579587	22233706
Russian-Chinese	33422682	21892537

Table 1: Data pre-processing results

Direction	Train Set	Dev Set	Test Set
Russian-English	42935974	2000	1421
English-Chinese	22231506	1100	1100
Russian-Chinese	21891537	965	1000

Table 2: Data partition results

tool Natasha[4]. The scales of sentence pairs of all corpora before and after data pre-processing are shown in Table 1.

After data preprocessing, we split the corpora into training set, development set and test set. The scales of the data partition are shown in Table 2.

3.2 System Settings

The open-source project fairseq[5](Ott et al., 2019) is chosen for this evaluation system. The main parameters are set as follows. Each model uses 1-3 GPUs for training, and the batch size is 2048. The embedding size and hidden size are set to 1024, the dimension of the feed-forward layer is 4096. We use six self-attention layers for both encoder and decoder, and the multi-head self-attention mechanism has 16 heads. The dropout mechanism(Provilkov et al., 2020) was adopted, and dropout probabilities are set to 0.3. BPE(Sennrich et al., 2016) is used in all experiments, where the merge operations is set to 32000. The maximum number of tokens is set to 4096. The loss function is set to "$label\_smoothed\_cross\_entropy$". The parameter "$adam\_betas$" is set to (0.9, 0.997). For the baseline system, the initial learning rate is 0.0007, the warm-up steps are set to 4000, and the maximum epoch number is set to 15. For the Encoder Combination system and Decoder Combination system[6], the initial learning rate is 0.0001, the warm-up steps are set to 4000, and the maximum epoch number is set to 10.

3.3 Experimental results

In the training of Russian-to-English translator, English-to-Chinese translator and Russian-to-Chinese translator, the single Transformer systems

[4] https://github.com/natasha/natasha

[5] https://github.com/pytorch/fairseq/tree/v0.6.2

[6] https://github.com/libeineu/Context-Aware

System	Russian-English	English-Chinese	Russian-Chinese
Transformer	20.89	17.83	16.07
Transformer+ Encoder Combination	21.53	18.87	16.66
Transformer+ Decoder Combination	21.76	18.91	16.79

Table 3: BLEU results on self-built test set

Method	BLEU
Primary: Pivot Method	19.2
Contrast: Direct Method	18.1

Table 4: BLEU results on released test set

are trained for 15 epochs. The context-based combination systems with Encoder Combination model or Decoder Combination model are trained for 10 epochs. The best epoch model and the last epoch model are ensembled to generate better results. The BLEU(Papineni et al., 2002) scoring results on the self-built test set are shown in Table 3.

The context-based combination systems with Decoder Combination model are used as our final submission since they outperform other systems.

Our primary submission uses the pivot method, which use English translation as the bridge. The Russian sentences are translated into English intermediate results by the well-trained Russian-to-English translator and then the English intermediate results are translated into Chinese output by the well-trained English-to-Chinese translator. Our contrast submission uses the direct method, which uses the well-trained Russian-to-Chinese translator to generate the target output.

As a result, our primary submission achieves a BLEU score of 19.2 and ranked the fourth among all participating teams. Our contrast submission achieves a BLEU score of 18.1 (shown in Table 4).

4 Conclusions

This paper introduces the main technologies and methods of ISTIC's submission in WMT 2021. To sum up, our model is constructed on the Transformer architecture of self-attention mechanism and context-based system combination method. In the aspect of data pre-processing, we explore several corpus filtering methods. In the process of translation output, the strategies of model ensemble and reranking are adopted. Experimental results show that these methods can effectively improve the quality of translation. It is worth mentioning that the pivot language translation bridge method

outperforms the direct translation method.

Acknowledgements

This research has been partially supported by ISTIC Fund ZD2021-17 and QN2021-12.

5 References

References

Gerda Claeskens and Nils Lid Hjort. 2008. *Model Selection and Model Averaging*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Bei Li, Hui Liu, Ziyang Wang, Yufan Jiang, Tong Xiao, Jingbo Zhu, Tongran Liu, and Changliang Li. 2020. Does multi-encoder help? a case study on context-aware neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3512–3518, Online. Association for Computational Linguistics.

Thibaud Lutellier, Hung Viet Pham, Lawrence Pang, Yitong Li, Moshi Wei, and Lin Tan. 2020. Coconut: Combining context-aware neural translation models using ensemble for program repair. In *Proceedings of the 29th ACM SIGSOFT International Symposium on Software Testing and Analysis*, ISSTA 2020, pages 101—-114, New York, NY, USA. Association for Computing Machinery.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, Florence, Italy. Association for Computational Linguistics.

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. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. 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 *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jeonghyeok Park and Hai Zhao. 2019. Korean-to-chinese machine translation using chinese character as pivot clue. In *Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation (PACLIC 33), Pacific Asia Conference on Language, Information and Computation*, pages 558–566.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000—-6010, Red Hook, NY, USA. Curran Associates Inc.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.

Jiaze Wei, Wenbin Liu, Zhenfeng Wu, You Pan, and Yanqing He. 2020. ISTIC's neural machine translation system for IWSLT'2020. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 158–165, Online. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2020. Neural machine translation: Challenges, progress and future. *CoRR*, abs/2004.05809.