ISTIC's Neural Machine Translation System for CCMT' 2021[⋆]

Hangcheng Guo, Wenbin Liu, Yanqing He*, Zhenfeng Wu, You Pan, Tian Lan, and Hongjiao Xu

Research Center of Information Theory and Methodology, Institute of Scientific and Technical Information of China, Beijing 100038, China
{guohc2020, liuwb2019, heyq, wuzf, pany, lantian, xuhj}@istic.ac.cn

Abstract. This paper demonstrates an overview and the technical details of the neural machine translation system developed by the Institute of Scientific and Technical Information of China (ISTIC) for the 17th China Conference on Machine Translation (CCMT' 2021). ISTIC participated in the following four machine translation (MT) evaluation tasks: MT task of Mongolian-to-Chinese daily expressions, MT task of Tibetan-to-Chinese government documents, MT task of Uyghur-to-Chinese news, and MT task of Russian-to-Chinese in low resource languages. Our system is based on Transformer architecture and several effective strategies are adopted to improve the quality of translation, such as corpus filtering, back translation, data augmentation, context-based system combination, model averaging, model ensemble, and reranking. The paper presents the system performance under different parameter settings.

Keywords: Neural Machine Translation · Self-Attention Mechanism · Context-based System Combination

1 Introduction

The machine translation team of the Institute of Scientific and Technical Information of China (ISTIC) participated in four machine translation evaluation tasks in the 17th China Conference on Machine Translation (CCMT'2021), including three bilingual evaluation tasks (Mongolian-to-Chinese daily expressions track, namely M2C; Tibetan-to-Chinese government documents track, namely T2C; Uyghur-to-Chinese news track, namely U2C) and one low resource evaluation task (Russian-to-Chinese tourism oral track, namely R2C). This paper describes the general overview and technical details of ISTIC's neural machine translation system for CCMT' 2021.

In this evaluation, we adopted the neural machine translation architecture of Google Transformer [1] as the basis of our system. As regards data source, the monolingual data released by the evaluation organizer is filtered to construct

[⋆] Corresponding author: Yanqing He, heyq@istic.ac.cn. This work was supported by ISTIC Fund (ZD2021-18).

pseudo parallel corpus through the back-translation method in M2C, T2C, and U2C evaluation tasks; the pseudo parallel corpus and the original given bilingual parallel corpus are used together as the training set of our neural machine translation system. External data of self-built Russian-Chinese dictionary and bilingual parallel corpus are introduced in the R2C evaluation task since the scale of given data is too small. In terms of data pre-processing, we proposed a general pre-processing method and a specific pre-processing method for the given data. Several filtering methods of the corpus are explored to reduce the data noise and improve the data quality. As for model construction, the context-based system combination method inputs the source sentence and its translation results from multiple machine translation systems as additional signals into multi-encoders respectively, which are weighted by attention mechanism to get combination result by encoder combination and decoder combination through gate mechanism. Model averaging and model ensemble strategies are adopted to generate the final output translation. We removed spaces between words and restored the target language translation results to the prescribed XML format in data post-processing. For each task, we compared the system performance under different parameter settings and further analyzed the experimental results.

The structure of this paper is as follows: the second part introduces the technical architecture of ISTIC's neural machine translation system; the third part introduces methods used in different tasks; the fourth part introduces the parameter settings, data pre-processing, experimental results, and related analysis; the fifth part gives the conclusion and future work.

2 System Architecture

Figure 1 shows the overall flow chart of our neural machine translation system in this evaluation which includes data pre-processing, model training, model decoding, and data post-processing (see Fig. 1).

2.1 Baseline System

Our baseline system used in participated evaluation tasks is Transformer, which includes an encoder and a decoder (see Fig. 2). The transformer is completely based on an attention mechanism. It can achieve algorithm parallelism, speed up model training, further alleviate long-distance dependence and improve translation quality [2].

The encoder and decoder are formed by stacking n identical layer blocks, where n is set to 6. Each layer of encoder contains two sub-modules, namely a multi-head self-attention module and a feed-forward neural network module. The multi-head self-attention module divides the dimension of the hidden state into multiple parts, and each part is separately calculated by using the self-attention function. Furthermore, these output vectors are concatenated together. The multi-head mechanism enables the model to pay more attention to the feature information of different positions and different sub-spaces. The multi-head attention method includes two steps: 1) dot product attention calculation; 2) multi-head attention calculation. The calculation method of dot product attention can be expressed as:

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

where Q is the query vector, K is the key vector, V is the value vector, and d_k is the dimension of the hidden layer state. Based on dot product attention, the calculation method of the multi-head attention mechanism can be expressed as:

$$MutiHead(Q, K, V) = Concat(head_i, ..., head_n) \cdot W^O \tag{2}$$

where W^O is the matrix parameter. The attention value of each head is:

$$head_i = Attention(Q \cdot W_i^Q, K \cdot W_i^K, V \cdot W_i^V) \tag{3}$$

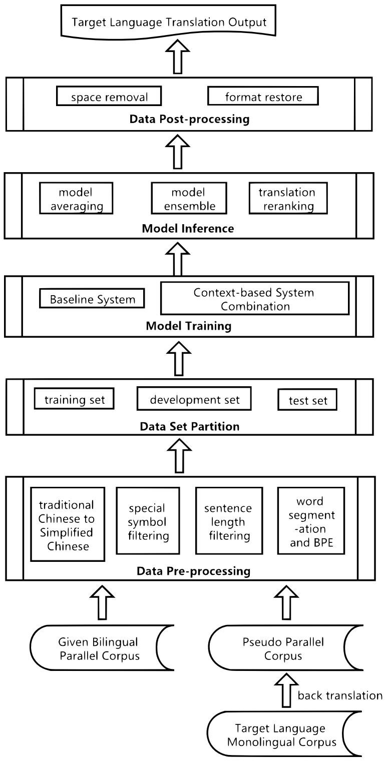

Fig. 1. Overall flow chart for machine translation tasks.

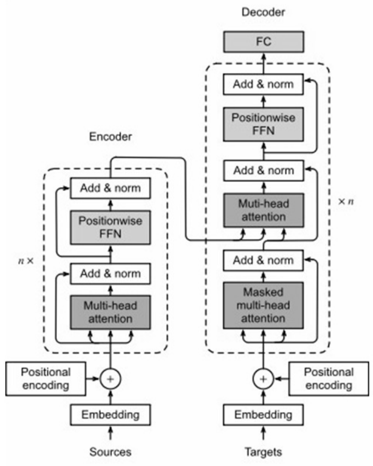

Fig. 2. Transformer model structure.

Each layer of the decoder is composed of three sub-modules. In addition to the two modules similar to the encoder, a decoder-encoder attention module is added between them and can focus attention on source language information in the decoding process. To avoid the problem that too many layers cause the model to be difficult to converge, both the encoder and the decoder use residual connection and hierarchical regularization techniques. To make the model better obtain the position information of the input sentence, additional position encoding vectors are added to the input layer of the encoder and decoder. After the encoder obtains a hidden state, the Transformer model inputs the hidden state into the softmax layer and scores with candidate vocabulary to obtain the final translation result.

2.2 Our System

Based on the transformer model, we propose a context-based [3] system combination method, which also adopts 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 [4] to encode the source sentences 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 Fig. 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.

$$H_{T_r} = Concat(H_{T_r 1}, ..., H_{T_r n}) \tag{4}$$

$$H = MutiHead(H_{T_r}, H_s) \tag{5}$$

The Decoder Combination model (see Fig. 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 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.

$$H = MutiHead(H_{T_r}, H_s, H_{Decoder}) \tag{6}$$

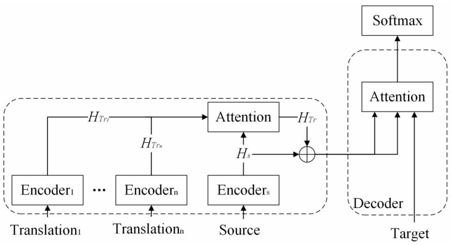

Fig. 3. Encoder Combination Model.

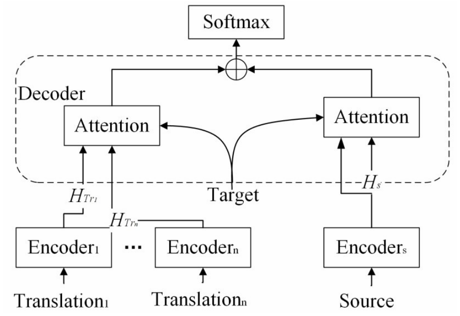

Fig. 4. Decoder Combination Model.

3 Methods In Different Tasks

In this evaluation, ISTIC participated in the four tasks of the Mongolia-Chinese, Tibetan-Chinese, Uyghur-Chinese, and Russian-Chinese. The methods used in each task are introduced below. For convenience, M2C represents Mongolian-to-Chinese daily expressions MT task, U2C represents Uyghur-to-Chinese news field MT task, T2C represents Tibetan-to-Chinese government documents MT task, and R2C represents Russian-to-Chinese low resource languages MT task.

3.1 M2C Task, U2C Task, and T2C Task

In the M2C task, U2C task, and T2C task, the parallel corpus is small in scale, so the back translation method is used to construct a pseudo parallel corpus. We used the given parallel corpus data of Mongolian-Chinese, Uyghur-Chinese, and Tibetan-Chinese released by the evaluation organizer to train the Chinese-to-Mongolian, Chinese-to-Uyghur, and Chinese-to-Tibetan translation models. The Chinese monolingual data filtered by Elasticsearch [5] is translated into the pseudo parallel corpus of Mongolian-Chinese, Uyghur-Chinese, and Tibetan-Chinese as a supplement to the training set of neural machine translation model.

In model training, the Transformer model based on the self-attention mechanism is adopted as the baseline model, and the Encoder Combination and Decoder Combination are introduced respectively. The source sentence and context information are integrated into the combination system. Here, the source language sentence, the target language sentence, and the machine translation of the source language sentence through the baseline model are used as the context information respectively.

The model averaging strategy [6] is used in the evaluation. Model averaging means that the parameters of the same model at different training moments are averaged and the more robust model parameters are obtained, which is helpful to reduce the instability of model parameters and enhance the robustness of the model. After the max epoch parameter is specified in the trainer and the training process is completed, our team gets the best epoch checkpoint and the last epoch checkpoint and averages the two checkpoints. The more stable and robust single model obtained by the model averaging strategy will also be used in model averaging, to jointly predict the probability distribution.

The model ensemble strategy [7] is also used in the evaluation. Model ensemble means that multiple models simultaneously predict the probability distribution of target words at the current time in decoding, and finally, a weighted average of the probability distribution predicted by multiple models is calculated, to jointly determine the final output after the model ensemble.

3.2 R2C Low Resource Task

The success of neural machine translation is closely related to computing resources, algorithm models, and data resources, especially the scale of bilingual training data. In the R2C task, the number of sentence pairs of parallel corpus available for training is as low as 50,000. Therefore, the introduction of external resources can effectively improve the performance of the machine translation system. Here, 123,605 phrase pairs and 55,504 sentence pairs from a self-built Russian-Chinese dictionary are used.

In our constrained system, the Encoder Combination and Decoder Combination are also adopted based on the Transformer model. Here, the target sentences in the training corpus are directly used as the context of the source sentences for system combination training. The strategies of the model averaging and model ensemble are also used. In our unconstrained system, a self-built 123605 Russian Chinese dictionary and 55504 bilingual parallel corpus data together with the original corpus are used as the training set. In our unconstrained system, 123,605 phrase pairs and 55,504 sentence pairs from a self-built Russian-Chinese dictionary are used as the training set together with the training corpus released by the evaluator.

4 Experiments

4.1 System Settings

The open-source project fairseq [8] [9] 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 [10] was adopted, and dropout probabilities are set to 0.3. BPE [11] 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 30. For the Encoder Combination system and Decoder Combination system, the initial learning rate is 0.0001, the warm-up steps are set to 4000, and the maximum epoch number is set to 10.

4.2 Data Pre-processing

In the M2C task, U2C task, and T2C task, the bilingual parallel corpus, and monolingual corpus are released by the evaluation organizer. In the R2C low resource task, the only bilingual parallel corpus is released.

1. Bilingual Parallel Corpus Pre-processing.

The Mongolian-Chinese parallel corpus is in the field of daily expressions, the Tibetan-Chinese parallel corpus is in the field of government literature, the Uygur-Chinese parallel corpus is in the field of news, and the Russian-Chinese parallel corpus belongs to low resource languages. The characteristics of language pairs of the evaluation tasks are similar as well as different. Therefore, a two-stage pre-processing method is designed as a general pre-processing stage and a specific pre-processing stage [12].

The the general pre-processing stage includes conversion from traditional Chinese to simplified Chinese, 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 [1, 200]. Sentence length ratio filtering excludes the sentence pairs whose ratio of source sentence length and target sentence length exceeds the range of [0.1, 10]. In the specific pre-processing stage, Chinese word segmentation is implemented using the lexical tool Urheen [13] and Chinese word segmentation is implemented using the lexical tool Polyglot [14].

In the M2C task, U2C task, and T2C task, parallel corpus data other than ”imu-test-mnzh-cwmt2018”, ”imu-test-uyzh-cwmt2018” and ”imu-test-tizh-cwmt2018” are taken as training sets. All the training set is processed in two stages. In the pre-processing of the development set and test set, the same content filtering, sentence length filtering, and sentence length ratio filtering are excluded. In the R2C task, all parallel corpus data is taken as the training set. Same content filtering, sentence length filtering, and sentence length ratio filtering are excluded in training data, development set, and test set. The number of sentence pairs of training set before and after data pre-processing is shown in Table 1.

Table 1. Training set data reprocessing results.

Direction	Before Pre-processing	After Pre-processing
mn-ch	269462	249069
uy-ch	170061	165143
ti-ch	162096	153324
ru-ch	50000	50000

2. Monolingual Corpus Pre-processing.

In the M2C task, U2C task, and T2C task, the scale of the Chinese monolingual corpus released by the CCMT’2021 evaluation organizer is 662904 news articles, about 11 million words. The monolingual data is filtered and screened by the Elasticsearch retrieval tool [12], and then both general pre-processing and specific pre-processing are used to obtain the final monolingual data for back-translation. The quantity is shown in Table 2.

In the M2C task, U2C task, and T2C task, the scale of the Chinese monolingual corpus released by the CCMT’2021 evaluation organizer is 662904 news articles, about 11 million words. The monolingual data is filtered and screened by Elasticsearch [12] retrieval tool, and then both general preprocessing and specific preprocessing are used to obtain the final monolingual data for back translation. The quantity is shown in Table.

For the selected monolingual data, this evaluation adopts a back-translation strategy to construct pseudo parallel corpus to enhance the machine translation results. According to the parallel corpus of Mongolian-Chinese, Uyghur-Chinese, and Tibetan-Chinese provided by CCMT’ 2021, the neural machine translation model of Chinese-to-Mongolian, Chinese-to-Uyghur and Chinese-to-Tibetan are constructed, and then the selected Chinese monolingual corpus is translated into the corresponding minority languages through these models. Finally, the pseudo

Table 2. The scale of pseudo parallel corpus.

Direction	sentence scale
mn-ch	240000
uy-ch	159894
ti-ch	140000

parallel corpus obtained from back translation and the preprocessed high-quality bilingual parallel corpus provided by CCMT' 2021 are mixed for training, to improve the machine translation quality of Mongolian Chinese, Tibetan Chinese, and Uighur Chinese.

4.3 Experimental results

In the Mongolian-to-Chinese translation evaluation task, the primary system (mc-2021-istic-primary-a) trains 10 epochs with the Encoder Combination model system and uses the last epoch checkpoint to decode. The contrast system 1 (mc-2021-istic-contrast-b) trains 10 epochs with the Encoder Combination model system and uses the model ensembling strategy to decode. The contrast system 2 (mc-2021-istic-contrast-c) trains 10 epochs with a Decoder Combination model system and uses the last epoch checkpoint to decode. The above three systems take the translated sentences of the source language sentences decoded by the intermediate translation model as the context and use the pseudo-parallel corpus constructed from monolingual data as the supplement of the training set. The BLEU5-SBP [15] scoring results on the released test set are shown in Table 3. Among all constrained systems, the primary system (mc-2021-istic-primary-a) ranked third. Among all of the participated systems, mc-2021-istic-contrast-b ranked fifth, mc-2021-istic-primary-a ranked sixth, and mc-2021-istic-contrast-c ranked seventh.

Table 3. BLEU5-SBP scoring of Mongolian-to-Chinese track on released test set.

System	BLEU5-SBP
mc-2021-istic-primary-a (Encoder Combination model)	0.3566
mc-2021-istic-contrast-b (Encoder Combination + model ensembling)	0.3607
mc-2021-istic-contrast-c (Decoder Combination model)	0.354

In the Tibetan-to-Chinese translation evaluation task, the primary system (tc-2021-istic-primary-a) takes the source language sentences as the context, trains 30 epochs with the Transformer baseline system, and uses the last epoch checkpoint to decode. The contact system 1 (tc-2021-istic-contact-b) uses the source language sentences as context, trains 30 epochs with the Transformer baseline system, and uses the model ensembling strategy to decode. The contrast system 2 (tc-2021-istic-contact-c) takes the target language sentence as the context, trains 10 epochs with the Encoder Combination model system and uses the last epoch checkpoint to decode. The above three systems do not use monolingual data, and the BLEU5-SBP scoring results on the released test set are shown in Table 4. Among all constrained systems, the primary system (tc-2021-istic-primary-a) ranked third. Among all of the participated systems, tc-2021-istic-contrast-c ranked fourth, tc-2021-istic-contrast-b ranked fifth, and tc-2021-istic-primary-a ranked the sixth.

Table 4. BLEU5-SBP scoring of Tibetan-to-Chinese track on released test set.

System	BLEU5-SBP
tc-2021-istic-primary-a (baseline)	0.1567
tc-2021-istic-contrast-b (baseline + model ensembling)	0.1678
tc-2021-istic-contrast-c (Encoder Combination model)	0.1737

In the Uyghur-to-Chinese translation evaluation task, the primary system (uc-2021-istic-primary-a) trains 10 epochs with a Decoder Combination model system and uses the best epoch checkpoint to decode. The contrast system 1 (uc-2021-istic-contrast-b) trains 30 epochs with a Transformer baseline system and uses the model ensembling strategy to decode. The contrast system 2 (uc-2021-istic-contrast-c) trains 10 epochs with the Encoder Combination model system and uses the model ensembling strategy to decode. The above three systems take the translated sentences of the source language sentences decoded by the intermediate translation model as the context and use the pseudo-parallel corpus constructed from monolingual data as the supplement of the training set. The BLEU5-SBP scoring results on the released test set are shown in Table 5. Among all constrained systems, the primary system (uc-2021-istic-primary-a) ranked fourth. Among all of the participated systems, uc-2021-istic-contrast-b ranked sixth, uc-2021-istic-contrast-c ranked seventh, and uc-2021-istic-primary-a ranked eighth.

Table 5. BLEU5-SBP scoring of Uyghur-to-Chinese track on released test set.

System	BLEU5-SBP
uc-2021-istic-primary-a (Decoder Combination model)	0.3495
uc-2021-istic-contrast-b (baseline + model ensembling)	0.352
uc-2021-istic-contrast-c (Encoder Combination + model ensembling)	0.35

In the Russian-to-Chinese translation evaluation task, the primary system (rc-2021-istic-primary-a) takes the target language sentences as the context, trains 30 epochs with the Transformer baseline system, and uses the last epoch checkpoint to decode. The contrast system 1 (rc-2021-istic-contrast-b) uses external 123,605 Russian-Chinese dictionary data and 55,504 bilingual parallel corpus data and the training process is the same as the primary system. The BLEU5-SBP scoring results on the released test set are shown in Table 6. Among all of the participated systems, rc-2021-istic-contrast-b ranked second, and rc-2021-istic-primary-a ranked third.

Table 6. BLEU5-SBP scoring of Russian-to-Chinese track on released test set.

System	BLEU5-SBP
rc-2021-istic-primary-a (baseline)	0.069
rc-2021-istic-contrast-b (baseline + dictionary)	0.1077

The results show that: (1) Model averaging, model ensembling, and multi-encoder system combination are helpful to improve translation quality; (2) The construction of pseudo parallel corpus by monolingual data back translation is conducive to the improvement of translation quality; (3) The accuracy of data pre-processing has a great influence on the quality of translation; (4) The method of multi-dimensional and multi similarity fusion is helpful to filter the corpus and select higher quality parallel sentence pairs.

5 Conclusions

This paper introduces the main technologies and methods of ISTIC in CCMT '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 the model averaging, model ensemble are adopted. In the process of corpus filtering, Elasticsearch is used for similar corpus filtering. Experimental results show that these methods can effectively improve the quality of translation. For machine translation tasks of low resource language, adding external dictionaries and parallel corpus can effectively improve the translation performance.

Due to the limited time, many methods have not been tried in this evaluation. Some problems have been found in the evaluation process, and the translation model still has a lot of room for improvement. In the future, we hope to learn more advanced technology and contribute to the research of machine translation.

References

1. Ashish Vaswani et al.: Attention is all you need. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 30 (NIPS 2017), pp. 5998–6008. Neural Information Processing Systems, Online. (2017)
2. Jiajun Zhang, Chengqing Zong.: Neural Machine Translation: Challenges, Progress and Future. In: Science China Technological Sciences, vol 63, pp. 2028—2050. Springer, Heidelberg (2020). https://doi.org/10.1007/s11431-020-1632-x
3. Elena Voita et al.: Context-aware neural machine translation learns anaphora resolution. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp. 1264—1274. Association for Computational Linguistics, Online. (2018)
4. Bei Li et al.: 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, pp. 3512—3518. Association for Computational Linguistics, Online. (2020)
5. Elasticsearch Homepage, https://github.com/elastic/elasticsearch. Last accessed 25 May 2021
6. Gerda Claeskens, Nils Lid Hjort.: Model Selection and Model Averaging. Cambridge University Press, Cambridge, United Kingdom. (2008). https://doi.org/10.1017/CBO9780511790485
7. Thibaud Lutellier et al.: 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, pp. 101–114. Association for Computing Machinery, New York, United States. (2020)
8. Myle Ott et al.: 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), pp. 48–53. Association for Computational Linguistics, Online. (2019)
9. Fairseq Homepage, https://github.com/pytorch/fairseq. Last accessed 15 May 2021
10. Ivan Provilkov et al. BPE-Dropout: Simple and Effective Subword Regularization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 1882-–1892. Association for Computational Linguistics, Online. (2020)
11. Rico Sennrich et al.: 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), pp. 1715—1725. Association for Computational Linguistics, Online. (2016)
12. Jiaze Wei, Wenbin Liu, Zhenfeng Wu, You Pan, Yanqing He.: ISTIC's Neural Machine Translation System for IWSLT' 2020. In: Proceedings of the 17th International Conference on Spoken Language Translation, pp. 158—165. Association for Computational Linguistics, Online. (2020)
13. Urheen, https://www.nlpr.ia.ac.cn/cip/software.html. Last accessed 15 May 2021
14. Polyglot, https://github.com/aboSamoor/polyglot. Last accessed 15 May 2021
15. Kishore Papineni et al. BLEU: a Method for Automatic Evaluation of Machine Translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311—318. Association for Computational Linguistics, Online. (2002)