Multi-Strategy Enhanced Neural Machine Translation for Chinese Minority Languages

Zhanglin Wu, Daimeng Wei, Xiaoyu Chen, Ming Zhu, Zongyao Li, Hengchao Shang, Jinlong Yang, Zhengzhe Yu, Zhiqiang Rao, Shaojun Li, Lizhi Lei, Song Peng, Hao Yang and Ying Qin

Text Machine Translation Lab, Huawei Beijing Research Center, Beijing, 100038, China.

Abstract. This paper presents HW-TSC's submissions to CCMT 2022 Chinese Minority Language Translation task. We participate in three language directions: Mongolian→Chinese Daily Conversation Translation, Tibetan→Chinese Government Document Translation, and Uighur→Chinese News Translation. We train our models using the Deep Transformer architecture, and adopt enhancement strategies such as Regularized Dropout, Tagged Back-Translation, Alternated Training, and Ensemble. Our enhancement experiments have proved the effectiveness of above-mentioned strategies. We submit enhanced systems as primary systems for the three tracks. In addition, we train contrast models using additional bilingual data and submit results generated by these contrast models.

Keywords: CCMT 2022; neural machine translation; regularized dropout; tagged back-translation; alternated training; ensemble

1 Introduction

CCMT 2022 Chinese Minority Language Translation Task is a challenging low-resource task. How to maximize low-resource translation performances using multiple enhancement strategies is the subject of this task, which is also our long-term research focus. We participate in the Mongolian→Chinese Daily Conversation Translation, Tibetan→Chinese Government Document Translation, and Uighur→Chinese News Translation tracks. For each track, we submit a primary system result and a copy of contrast translation. In the following chapters we will introduce our data processing method, model training strategies, experiment results, and findings.

2 Dataset

2.1 Data Volume

We strictly comply with the task requirements and use only officially-provided bilingual and monolingual data to train our primary systems. For our contrast models, additional bilingual data is used. Table 1 presents the data size for each language pair after pre-processing.

Table 1 Data size for each language pair after pre-processing

	Mongolian→Chinese	Tibetan→Chinese	Uygur→Chinese
bilingual	1.24M	0.97M	0.16M
monolingual	3.94M	3.94M	3.94M
additional bilingual	4.97M	1.54M	6.89M

2.2 Data Pre-processing

The data pre-processing process is as follows:

- Remove duplicate sentences.
- Remove invisible characters.
- Reverse xml escape character.
- Convert full-width symbols to half-width symbols.
- Use jieba word segmentation tool for Chinese sentences.
- Use joint BPE[1], and the vocabulary size is set to 32k.
- Filter out sentences with more than 150 tokens.
- Filter out sentence pairs with token ratio greater than 4 or less than 0.25.

3 System Overview

3.1 Model

The Transformer[2] model adopts the full self-attention mechanism, which can realize algorithm parallelism, speed up model training, and improve translation quality. Deep Transformer[3] can further improve the transformer performance by applying layer normalization to the input of every sub-layer and increasing the number of encoder layers. Therefore, in all three tracks, we use the following model architecture:

- Deep Transformer: Based on the Transformer-big model architecture, our Deep Transformer model features pre-layer-normalization, 25-layer encoder, 6-layer decoder, 16-head self-attention, 1024 dimensions of word embedding and 4096-hidden-state.

3.2 Regularized Dropout

Dropout[4] is a powerful and widely used technique for regularizing deep neural networks. Though it can help improve training effectiveness, the randomness introduced

by dropouts may lead to inconsistencies between training and inference. Regularized Dropout[5] forces the output distributions of different sub models generated by dropout be consistent with each other. Therefore, we use Regularized Dropout to enhance the baseline for each track and reduce inconsistencies between training and inference.

3.3 Back-Translation

In order to utilize target-side monolingual data to improve model performance, we use Back-Translation[6] to expand the training corpus. There are many specific implementation methods[7–10] for Back-Translation. During the experiment, we verify the effectiveness of two methods, namely, Top-K Sampling Back-Translation[8] and Tagged Back-Translation[9], and finally choose to use Tagged Back-Translation according to the experimental results.

3.4 Alternated Training

Due to the scarcity of authentic bilingual data, pseudo-bilingual data plays an important role in improving translation quality, but it inevitably introduces noise and translation errors. In order to alleviate the noise and translation errors caused by pseudo-bilingual data and improve the translation quality, we use the Alternated Training strategy[11]. The basic idea is to alternately use pseudo-bilingual data and authentic bilingual data in the training process until there is no noticeable improvement in translation quality.

3.5 Ensemble

Ensemble[12] is a widely-used technique to integrate different models for better performance. It should be noted that when using the Ensemble strategy, increasing the number of models does not always lead to better performance and may even hurt the final accuracy. Therefore, for each track, we train four models using the same data, and then select the models used for ensemble according to the strategy we used in the WMT21 Biomedical Translation Task[13]. The core idea is traverse all combinations of models and find the best one in the dev set.

4 Experiments

In the training phase, we use the Pytorch-based Fairseq[14] open-source framework and use the Deep Transformer model as the benchmark system. Each model uses 8 GPUs for training, and the batch size is 1024. The update frequency is set to 4, and the learning rate is 5e-4. The label smoothing rate[15] is set to 0.1, the number of warmup steps is 4000, and the dropout is 0.3. Adam optimizer[16] with $\beta 1$=0.9 and $\beta 2$=0.98 is used. In addition, when applying Regularized Dropout, we follow the setting of Liang et al[5], using reg_label_smoothed_cross_entropy as the loss function, and set reg-alpha to 5. In the inference phase, we use the Marian[17] tool to perform decoding. The beam size is set to 10, and the length penalties for Mongolian→Chinese, Tibetan→Chinese and Uyghur→Chinese machine translation are 1.0, 0.6, and 1.4

respectively. During the experiment, we find that there are super-long sentences in the development sets and test sets. Therefore, we segment sentences with more than 150 tokens based on punctuations indicating the end of a sentence before translation.

4.1 Mongolian→Chinese

With regard to the Mongolian→Chinese translation track, we found that a large portion of target-side text in the CCMT 2019 and CCMT 2020 development sets is also found in this year's training set, resulting in model overfitting. In order to fairly and accurately assess the model performance, we use a subset of CCMT 2020 development set. The subset contains only bitexts whose reference are not in the training set. During the training, we adopt enhancement strategies such as Regularized Dropout, Tagged Back-Translation, Alternated Training, and Ensemble. In addition, we train two contrast systems: contrast system b is fine-tuned on CCMT 2019 and CCMT 2020 development sets; while contrast system c is trained with additional bilingual data in the last step of alternated training, and ensembled by multiple models.

Table 2 presents the sacreBLEU[18] results for Mongolian→Chinese translations under different strategies. Using the CCMT 2020 subset for assessment, we found that Regularized Dropout, Tagged Back-Translation, Alternated Training, and Ensemble can all improve model performance. On the contrary, additional bilingual data used in contrast system c does not lead to further improvements. As a result, we fine-tune models on development sets in hope of further improving model performance on CCMT 2022 test set.

Table 2 BLEU scores of Mongolian→Chinese translation

	CCMT 2019	CCMT 2020	CCMT 2020 subset
baseline	**69.15**	67.96	33.50
+ regularized dropout	68.21	69.88	37.44
+ tagged back-translation	54.56	67.04	45.77
+ alternated training	57.70	69.74	47.01
+ ensemble(primary a)	57.87	70.33	47.63
+ fine-tuning(contract b)	61.99	**72.96**	**52.63**
+ additional bilingual(contract c)	60.76	70.40	47.26

4.2 Tibetan→Chinese

With regard to the Tibetan→Chinese track, we train a baseline model with only bilingual data and use multiple enhancement strategies. We ensemble multiple models to generate the primary system. We also train a contrast system by ensemble models that use additional bilingual data in the last step of alternated training. Table 3 presents the experiment results, demonstrating that Regularized Dropout, Tagged Back-Translation, Alternated Training, and Ensemble all help improve model performance. In addition, adding additional bilingual data during training can lead to further improvement.

Table 3 BLEU scores of Tibetan→Chinese translation

	CCMT 2019	CCMT 2020
baseline	47.85	61.45
+ regularized dropout	49.35	62.55
+ tagged back-translation	50.38	65.94
+ alternated training	53.56	66.97
+ ensemble(primary a)	54.46	67.96
+ additional bilingual(contract b)	**66.44**	**74.11**

4.3 Uyghur→Chinese

With regard to the Uyghur→Chinese track, we adopt the same training strategy as that in the Tibetan→Chinese track. Table 4 presents the experiment results. The results also demonstrate that all enhancement strategies mentioned, as well as additional bilingual data, can lead to model improvements.

Table 4 BLEU scores of Uyghur→Chinese translation

	CCMT 2019	CCMT 2020
baseline	44.59	47.36
+ regularized dropout	48.26	51.71
+ tagged back-translation	54.89	59.59
+ alternated training	55.56	60.20
+ ensemble(primary a)	55.66	60.44
+ additional bilingual(contract b)	**59.06**	**64.28**

5 Analysis

5.1 The Effect of Different Back-Translation Methods

Past experience demonstrates that Tagged Back-Translation and Top-K Sampling Back-Translation are effective Back-Translation variants. We conduct comparative experiments on the two methods on the three minority language translation tracks. Experiment results shown in Table 5 indicate that Tagged Back-Translation can achieve better results in low-resource translation scenarios.

Table 5 BLEU scores of two different back-translation methods

	Mongolian→Chinese			Tibetan→Chinese		Uyghur→Chinese	
CCMT devset	2019	2020	2020 subset	2019	2020	2019	2020
tagged back-translation	54.56	**67.04**	**45.77**	50.38	**65.94**	**54.89**	**59.59**
top-k sampling back-translation	**54.63**	66.19	45.52	**51.64**	64.18	53.24	57.34

5.2 The Impact of Sentence Segmentation on the Translation Quality of Machine Translation

During experiments, we found development sets and test sets in all three language pairs contain some super-long sentences with more than 150 tokens. During training, we have filter out sentences more than 150 tokens. We assume that models cannot directly translate those super-long sentences well and do segmentation on those sentences based on punctuations that indicate the end of a sentence. Table 6 presents BLEU results before and after segmentation and demonstrate that segmentation is effective in improving Tibetan→Chinese and Uyghur→Chinese translation tasks. But we see no improvement on Mongolian→Chinese translation.

Table 6 Bleu scores of whether the baseline uses sentence segmentation.

	Mongolian→Chinese			Tibetan→Chinese		Uyghur→Chinese	
CCMT devset	2019	2020	2020 subset	2019	2020	2019	2020
baseline	69.15	67.96	**33.50**	**47.85**	**61.45**	**44.59**	**47.36**
- sentence segmentation	**69.58**	**68.08**	33.47	45.88	59.91	42.84	45.82

5.3 Analysis of BLEU scores of Mongolian→Chinese machine translation on the development set

We find an abnormal phenomenon during Mongolian→Chinese experiment: we see no consistent improvements on CCMT 2019 and CCMT 2020 development sets when using Regularized Dropout and Tagged Back-Translation. So we conduct an analysis on the overlapping between development sets and training set. We found that the majority of Chinese text in CCMT 2019 development set and half of Chinese text in CCMT 2020 development set are also in this year's training data. So we construct a sub development set containing only sentences not in the training data, in hope of evaluating the model performance in a more fair way.

Table 7 The number of source sentences, target sentences and sentence pairs in the development set that appear in the training set.

	source in Training Set	target in Training Set	sentence pair in Training Set
CCMT 2019	21	958	20
CCMT 2020	6	584	6

6 Conclusion

This paper presents our submissions to the CCMT 2022 Mongolian→Chinese, Tibetan→Chinese, and Uyghur→Chinese translation tasks. We train our models using the Deep Transformer architecture and employ enhancement strategies such as Regularized Dropout, Tagged Back-Translation, Alternated Training, and Ensemble.

We also train contrast models with additional bilingual data. In addition, we conduct experiments on two Back-Translation variants (Tagged Back-Translation and Top-K Sampling Back-Translation), analyze how segmentation influences the translation quality of neural machine translation model, and find a better solution to the abnormal phenomenon on Mongolian→Chinese development sets.

References

[1] Sennrich R, Haddow B, Birch A. Neural Machine Translation of Rare Words with Subword Units[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1715-1725.

[2] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[3] Wang Q, Li B, Xiao T, et al. Learning Deep Transformer Models for Machine Translation[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 1810-1822.

[4] Hinton G E, NitishSrivastava A K, Salakhutdinov I S R R. Improving neural networks by preventing co-adaptation of feature detectors[J].

[5] Wu L, Li J, Wang Y, et al. R-Drop: Regularized Dropout for Neural Networks[C]//Advances in Neural Information Processing Systems. 2021.

[6] Burlot F, Yvon F. Using Monolingual Data in Neural Machine Translation: a Systematic Study[C]//Proceedings of the Third Conference on Machine Translation: Research Papers. 2018: 144-155.

[7] Edunov S, Ott M, Auli M, et al. Understanding Back-Translation at Scale[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 489-500.

[8] Graça M, Kim Y, Schamper J, et al. Generalizing Back-Translation in Neural Machine Translation[C]//Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers). 2019: 45-52.

[9] Caswell I, Chelba C, Grangier D. Tagged Back-Translation[C]//Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers). 2019: 53-63.

[10] Abdulmumin I, Galadanci B S, Isa A. Enhanced back-translation for low resource neural machine translation using self-training[C]//International Conference on Information and Communication Technology and Applications. Springer, Cham, 2020: 355-371.

[11] Jiao R, Yang Z, Sun M, et al. Alternated Training with Synthetic and Authentic Data for Neural Machine Translation[C]//Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. 2021: 1828-1834.

[12] Garmash E, Monz C. Ensemble learning for multi-source neural machine translation[C]//Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016: 1409-1418.

[13] Yang H, Wu Z, Yu Z, et al. HW-TSC's Submissions to the WMT21 Biomedical Translation Task[C]//Proceedings of the Sixth Conference on Machine Translation. 2021: 879-884.

[14] Ott M, Edunov S, Baevski A, et al. fairseq: A Fast, Extensible Toolkit for Sequence Modeling[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations). 2019: 48-53.

[15] Szegedy C, Vanhoucke V, Ioffe S, et al. Rethinking the Inception Architecture for Computer Vision[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2016, 2016: 2818-2826.

[16] Kingma D P, Ba J L. Adam: A Method for Stochastic Optimization[J]. 2015.

[17] Junczys-Dowmunt M, Grundkiewicz R, Dwojak T, et al. Marian: Fast neural machine translation in c++[C]//ACL 2018-56th Annual Meeting of the Association for Computational Linguistics, Proceedings of System Demonstrations. 2015.

[18] Post M. A Call for Clarity in Reporting BLEU Scores[C]//Proceedings of the Third Conference on Machine Translation: Research Papers. 2018: 186-191.