Effective Data Augmentation Methods for CCMT 2022

Jing Wang and Lina Yang✉

School of Computer and Electronic Information
Guangxi University, Nanning, China
lnyang@gxu.edu.cn

Abstract. The purpose of this paper is to introduce the specific situation in which Guangxi University participated in the 18th China Conference on Machine Translation (CCMT 2022) evaluation tasks. We submitted the results of two bilingual machine translation (MT) evaluation tasks in CCMT 2022. One is Chinese-English bilingual MT tasks from the news field, the other is Chinese-Thai bilingual MT tasks in low resource languages. Our system is based on Transformer model with several effective data augmentation strategies which are adopted to improve the quality of translation. Experiments show that data augmentation methods have a good impact on the baseline system and aim to enhance the robustness of the model.

Keywords: Machine Translation · CCMT 2022 · Transformer · Data Augmentation

1 Introduction

In the context of the rapid development of deep learning, neural machine translation (NMT) has attracted more and more attention from the academic community. We participated in four directions of machine translation evaluation tasks. And we built our translation systems based on Google's Transformer [11] model in all directions.

The reason why we selected the Transformer model is that it solved the problem of long-distance information loss. In addition, we applied BPE algorithm which Sennrich [9] proposed in 2016 to word-segmented texts to deal with the out-of-vocabulary (OOV) problem. Finally, we used several data augmentation strategies to generate pseudo data, enrich the diversity of data, and make up for the lack of training data. Data augmentation is defined by many authors as a solution to a data distribution mismatch problem [13]. In short, data augmentation has been used in low resource tasks due to the requirement of large amounts of training data.

The remaining part of the paper proceeds as follows. Chapter Two briefly describes the Transformer model. After that, Chapter Three is concerned with the data augmentation methodologies used for our model. Chapter Four then describes the experimental settings and discusses the results our model obtained. This paper ends with a conclusion and future work.

2 System Architecture

In CCMT 2022 translation evaluation tasks, we adopted our neural machine translation model based on the Transformer model. The Transformer model adapts the encoder-decoder architecture which is one of the most popular architectures. In recent years, the attention mechanism has been widely used to solve the insufficient dependency when modeling long sequences. Meanwhile, the attention mechanism is the most significant part of the Transformer model. And it also is an important reason why Transformer has achieved great success in many fields.

The decoder and encoder of Transformer have a similar structure consisting of n identical layers. Each layer contains two main modules: an attention mechanism module and a feed-forward neural network module. Scaled dot-product attention mechanism performs the following operations on the input query, key, and value as follows:

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

where $\sqrt{d_k}$ is the dimension of the hidden layer state. Based on scaled dot-product attention, the calculation method of the multi-head attention mechanism can be expressed as:

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{2}$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$. Multi-head attention enriches the representation of semantic information. When calculating the attention score, the input is divided into multiple parts on average, then each part is calculated as attention score independently. Finally, all the obtained attention scores are concatenated together as the output of the multi-head attention layer. After calculating the self-attention, the following feed-forward neural network is used to transform the input.

The residual connection [2] is also important in Transformer architecture. It can prevent the problem of vanishing gradients and increase the network depth further. The residual connection is employed around each of the two sub-layers, followed by layer normalization.

3 Methods

3.1 Data Augmentation

Data augmentation is an important machine learning method nowadays. It is based on the existing training sample data to generate more training data. Its purpose is to make the expanded training data as close to the real distributed data as possible and improve the translation quality further. In addition, data augmentation can force the model to learn more robust features and effectively improve the generalization ability of the model. Figure 1 shows the process of

generating pseudo parallel corpus. In CCMT 2022, we use the data augmentation methods as follows.

Swap Randomly select two words in the target sentence and exchange their order until words of $\alpha \cdot n$ sentence length are exchanged.

BPE-Dropout BPE-Dropout [6] algorithm was proposed in 2020 by Provilkov et al. BPE-Dropout stochastically corrupts the segmentation procedure of BPE, leading to different subword segmentation with the same BPE vocabulary.

Synonym Replacement Synonym Replacement is to replace a word with its synonym. And this word is randomly selected in the target sentence. We believe that synonym replacement can enrich the diversity of training data.

Word-Replacement Use mgiza++[1] toolkit to obtain bi-directional alignment lexicon from the training data. $\alpha \cdot t$ source-target aligned words are selected at random and replaced by random entries in the bi-directional alignment lexicon.

Back Translation Back Translation [8] is the process of translating the target language into the source language. On the one hand, it can augment the pseudo parallel pairs. On the other hand, it can improve the generalization ability of the model.

Fine Tuning Fine tuning [1] is an effective method which can bring improvements to neural network. Our translation systems trained with data augmentation method were fine-tuned on the training set.

3.2 CE Task and EC Task

In Chinese-English machine translation tasks, our baseline system was developed in base Transformer model. Besides, we built two contrast systems which both use data augmentation strategies. One contrast system made the use of word-replacement operation. The other applied several data augmentation methods to enhance the performance of the model.

Specifically, we generated 1M synthetic data by the word-replacement operation. Word-Replacement makes use of statistical machine translation to generate a bilingual aligned lexicon. Sánchez-Cartagena et al. [7] used each of the multi-task learning data augmentation auxiliary tasks to stress the fact that the augmented data. In addition, we mixed several data augmentation methods, such as swap, insert, delete, and so on. We augmented 4M pseudo parallel data by using this method. The paper of [12] motivates this idea. The data and model settings of the CE task and EC task are consistent.

[1] https://github.com/moses-smt/mgiza

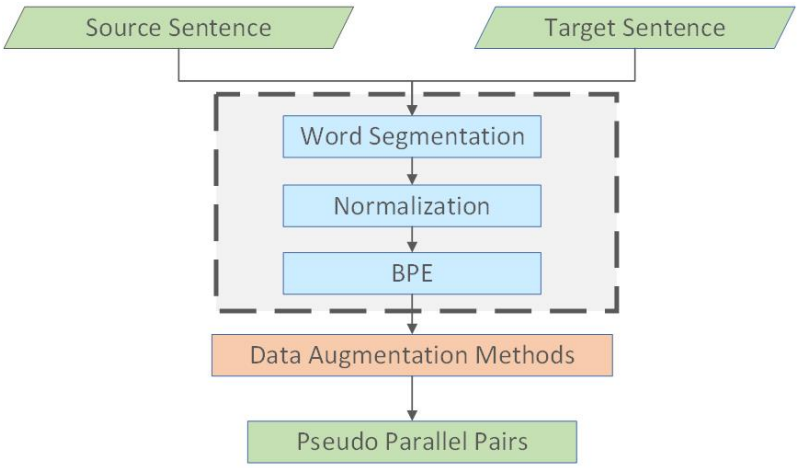

Fig. 1. Overall flow chart for data augmentation.

3.3 CThai Task and ThaiC Task

In Chinese-Thai low resource translation tasks, our systems applied data augmentation methods which included back translation and swap to improve the performance of translation. We used Tencent AI Lab Embedding Corpus for Chinese Words and Phrases [2] [10] to do synonym replacement. At the same with Chinese-English translation tasks, fine tuning was used in Chinese-Thai data augmentation systems.

For back translation, we applied only for the Thai-Chinese direction. And we conducted experiments to evaluate the impact of fine tuning technology on the model.

4 Experiments

4.1 System Settings

We use fairseq[3] [3] open-source framework to implement our translation systems. The toolkit fairseq was implemented in 2019. In bilingual Chinese-English directions, our Transformer model includes six layers for the encoder and six layers for the decoder, respectively. Each layer has the size of 512 hidden units. We also set the size of embedding layers to 512. The dimension of the feed-forward layer is 2048. And the multi-head self-attention mechanism has 8 heads. However, we set the encoder layer number and decoder layer number to 5 in the low resource translation tasks. The multi-head self-attention mechanism only has 4 heads. Table 1 shows the main parameters of our Transformer model. The parameter

[2] https://ai.tencent.com/ailab/nlp/zh/embedding.html
[3] https://github.com/pytorch/fairseq

patience stands for early cease training if valid performance doesn't improve for N consecutive validation runs.

Table 1. Model Settings

Parameter	Chinese-English	Chinese-Thai
Embedding Size	512	512
Encoder Layer	6	5
Decoder Layer	6	5
Dropout	0.3	0.3
Encoder Attention Heads	8	4
Decoder Attention Heads	8	4
Warm-up Steps	16000	8000
Patience	20	6

In the low resource translation tasks, the parallel corpus is limited so that we selected a slightly smaller Transformer model. Specifically, we apply layer normalization before each encoder block. The same goes for each decoder block. A major advantage of these settings is to prevent the model from over-fitting during training.

4.2 Data Pre-processing

As we all know, data pre-processing is an especially important part of machine translation. Data pre-processing is also the first step to solving practical problems by deep learning. It mainly includes duplicate removal, symbol normalization, word segmentation, and so on. Next, we will introduce our data pre-processing steps.

In the CE and EC task, the evaluation organizers provide about 9M Chinese-English parallel corpus and 11M Chinese monolingual corpus. We only use the NEU2017 corpus, the Datum2015, and the Datum2017 as the training set in Chinese-English MT tasks in our submitted systems. The test set consists of newstest2019.

In the CThai low resource translation task, the only bilingual parallel corpus is released. Therefore, we sample 195K randomly as the training set. The rest of the parallel data is divided into validation set and test set in a ratio of 2:3.

After splitting the training set, validation set, and test set, we begin to process the sentences. For English sentences, we tokenize the English word by the space first. Then, we need to learn the most suitable case form for English words due to the problems of different cases of the same word. For Thai sentences, we use pythainlp[4] to do word segmentation. It is a useful toolkit for us to split the Thai sentences. Finally, we use jieba[5] for Chinese sentences to tokenize the Chi-

[4] https://github.com/PyThaiNLP/pythainlp
[5] https://github.com/fxsjy/jieba

nese texts, whose advantages are fast and high accuracy compared with other Chinese word segmentation tools.

In all languages, we use Moses scripts[6] to normalize the texts from digits, punctuations and special symbols. Additionally, we use Subword-NMT[7] toolkit to learn and apply Sennrich's BPE from the tokenized texts. Lastly, those sentence pairs are removed which are less than 5 BPE tokens or more than 100 BPE tokens in the training set. We completed all our experiments on a single RTX3090.

4.3 Experimental Results

BLEU [4] is one of the most commonly used automatic evaluation methods for machine translation. We use sacrebleu[8] [5] to calculate the score of BLEU for our submitted results in CCMT2022.

In the CE translation evaluation task, we submit three translation systems, which include one baseline system and two contrast systems. Chinese and English do not share the alphabet, so we learn 16K BPE operations separately on Chinese and English texts by using Subword-NMT toolkit. Mixture DA is a data augmentation method that performs random swap, random insert, and random delete of words in sentences. Table 2 reports the performance of our CE translation systems. We can infer that data augmentation is effective due to the increment of 1.04 BLEU points in the validation set and 0.64 BLEU points in the test set. And we find that it is obvious that data augmentation methods increase the accuracy of the model.

Table 2. The BLEU Scores of CE Task

System	Valid Set	Test Set
Baseline	19.73	17.56
Mixture DA	19.67	16.66
Word-Replacement	20.77	18.20

In the EC translation evaluation task, we also submit three translation systems. It indicates the results of our submitted translation systems in Table 3. The methods of EC translation systems used are exactly the same as the CE task. We know clearly that word-replacement receives the best results among all EC translation systems from Table 3.

In the CThai translation evaluation task, we find that BPE-Dropout algorithm does not perform well on the low resource dataset through experiments. Then, we use synonym replacement method to generate Chinese-Thai pairs. Table 4 reports the results of our translation systems. Synonym Replacement

[6] https://github.com/moses-smt/mosesdecoder/tree/master/scripts
[7] https://github.com/rsennrich/subword-nmt
[8] https://github.com/mjpost/sacrebleu

Table 3. The BLEU Scores of EC Task

System	Valid Set	Test Set
Baseline	14.49	23.20
Mixture DA	16.49	25.10
Word-Replacement	18.42	26.26

method obtains an obvious improvement. And it can perform better when it combines with fine tuning method.

Table 4. The BLEU Scores of CThai Task

System	Valid Set	Test Set
Baseline	11.63	11.97
Synonym Replacement	14.39	15.18
Synonym Replacement + Fine Tuning	15.12	15.64

In the ThaiC translation evaluation task, four translation systems are submitted. As shown in Table 5, we discover that reverse has a positive effect on the baseline system. In addition, it does not work for the model to only use back translation. When we add fine tuning to back translation method, the model obtains a better result. What's more, combining swap and fine tuning realizes the best performance in the ThaiC task.

Table 5. The BLEU Scores of ThaiC Task

System	Valid Set	Test Set
Baseline	12.13	16.27
Back Translation	10.95	15.02
Back Translation + Fine Tuning	12.87	17.21
Swap + Fine Tuning	14.00	18.28

5 Conclusion

In this paper, we described our translation systems in four translation evaluation tasks including Chinese to English, English to Chinese, Chinese to Thai, and Thai to Chinese. In all directions, our experiments proved that our translation systems have been improved on data augmentation methods. Our data augmentation strategies bring good performance to the baseline system in these translation evaluation tasks. In the future, we expect that we explore more data augmentation approaches, especially in some fields where parallel data is scarce. And we hope that our proposed data enhancement methods can be applied to different neural network models and datasets.

References

1. Chu, C., Dabre, R., Kurohashi, S.: An empirical comparison of domain adaptation methods for neural machine translation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 385–391 (2017)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
3. Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., Auli, M.: fairseq: A fast, extensible toolkit for sequence modeling. arXiv preprint arXiv:1904.01038 (2019)
4. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: 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 (2002)
5. Post, M.: A call for clarity in reporting bleu scores. In: Proceedings of the Third Conference on Machine Translation: Research Papers. pp. 186–191 (2018)
6. Provilkov, I., Emelianenko, D., Voita, E.: Bpe-dropout: Simple and effective subword regularization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 1882–1892 (2020)
7. Sánchez-Cartagena, V.M., Esplà-Gomis, M., Pérez-Ortiz, J.A., Sánchez-Martínez, F.: Rethinking data augmentation for low-resource neural machine translation: A multi-task learning approach. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 8502–8516 (2021)
8. Sennrich, R., Haddow, B., Birch, A.: Improving neural machine translation models with monolingual data. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 86–96 (2016)
9. Sennrich, R., Haddow, B., Birch, A.: 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 (2016)
10. Song, Y., Shi, S., Li, J., Zhang, H.: Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). pp. 175–180 (2018)
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
12. Wei, J., Zou, K.: Eda: Easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196 (2019)
13. Wei, X., Yu, H., Hu, Y., Weng, R., Xing, L., Luo, W.: Uncertainty-aware semantic augmentation for neural machine translation. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 2724–2735 (2020)