DUTNLP Machine Translation System for WMT22 General MT Task

Ting Wang **Huan Liu** **Junpeng Liu** **Degen Huang***
School of Computer Science, Dalian University of Technology
{Wting_1513577,liuhuan4221,liujunpeng_nlp}@mail.dlut.edu.cn
huangdg@dlut.edu.cn

Abstract

This paper describes DUTNLP Lab's submission to the WMT22 General MT Task on four translation directions: English to/from Chinese and English to/from Japanese under the constrained condition. Our primary system are built on several Transformer variants which employ wider FFN layer or deeper encoder layer. The bilingual data are filtered by detailed data pre-processing strategies and four data augmentation methods are combined to enlarge the training data with the provided monolingual data. Several common methods are also employed to further improve the model performance, such as fine-tuning, model ensemble and post-editing. As a result, our constrained systems achieve 29.01, 63.87, 41.84, and 24.82 BLEU scores on Chinese → English, English → Chinese, English → Japanese, and Japanese → English, respectively.

1 Introduction

DUTNLP Lab participates in the WMT22 General MT Task on four translation directions: English ↔ Chinese and English ↔ Japanese. Our translation system is trained on the officially provided bilingual and monolingual data under the constrained condition. Several strategies such as fine-grained data pre-processing, large-scale synthetic data augmentation, diverse model architectures and domain fine-tuning are utilized to enhance the performance of the final ensemble model.

Since the quality of the training data is crucial to the translation performance, all the training sets are filtered by the off-the-shelf toolkits and some manual rules. Details will be discussed in Section 2. Those data pre-processing strategies are also employed to filter out the synthetic data generated by different data augmentation methods.

To generate synthetic parallel data, four data augmentation methods including back-translation (Sennrich et al., 2016), forward-translation (Wu et al., 2019), knowledge distillation (Freitag et al., 2017) and R2L training (Liu et al., 2016) are employed in our experiments. Specifically, we leverage source-side monolingual data by exploring forward-translation, knowledge distillation and R2L training, while target-side monolingual data by back-translation. These strategies increase the data size to a large extent. The generated data and the original parallel data are combined to train NMT models.

For model architectures, starting from Transformer-Big (Vaswani et al., 2017) settings, several Transformer variants are used to improve the model capacity and diversity. Previous studies (Bapna et al., 2018; Li et al., 2020) have shown that the translation performance can be significantly improved by increasing the model capacity. Therefore, we build different model architectures with either wider FFN layers (Ng et al., 2019) or deeper transformer encoder (Sun et al., 2019). Moreover, the Pre-Norm (Wang et al., 2019) is also adopted in all our experiments as its performance and training stability are better than the Post-Norm counterpart.

Domain fine-tuning is the most effective method in our experiments, which greatly improves the translation performance. We first employ previous WMT test sets as the domain data to fine-tune several models with different architectures. Then we ensemble those fine-tuned models and translate the test sets to construct pseudo parallel data. Finally, the original and the pseudo test sets are merged for further domain fine-tuning.

This paper is structured as follows: Section 2 describes the data pre-processing strategies. We present the details of our systems in Section 3 and show the experiment settings and results in Section 4. We draw the conclusion in Section 5.

*Corresponding author

Proceedings of the Seventh Conference on Machine Translation (WMT), pages 397–402
Abu Dhabi, December 7–8, 2022.

2 Data Pre-processing

For each language pair, we follow the constrained data requirements and make full use of the provided bilingual and monolingual data. Table 1 lists the data we used in our experiments.

Language Pair	Filtered Bilingual	Monolingual
En-Zh	34.5M	En:15M Zh:15M
En-Ja	20.1M	En:20M Ja:20M

Table 1: Statistics of the training dataset.

As the quality of the parallel training data is crucial to the final translation performance, we perform fine-grained data filtering with the off-the-shelf toolkits and some manual rules. For both language pairs, the pre-processing strategies are as follows:

- Normalize punctuation with Moses scripts (Koehn et al., 2007) for English. Chinese and Japanese text are separately segmented by jieba[1] and MeCab[2] toolkits.
- Filter out the duplicated sentence pairs.
- Filter out sentences containing html tags, illegal characters and invisible characters.
- Filter out sentences with the character-to-word ratio higher than 12 or lower than 1.5 following (Wei et al., 2021).
- Filter out sentences with the source-to-target token ratio higher than 3 or lower than 0.3 following (Wei et al., 2021).
- Filter out sentences in other languages by applying language identification (Joulin et al., 2016).
- Filter out sentence pairs with low alignment score by using fast-align (Dyer et al., 2013).
- For Chinese, we convert full-width format to half-width format and convert traditional Chinese characters to simplified ones.

3 System Overview

3.1 Model Architectures

Previous studies (Bapna et al., 2018; Li et al., 2020; Wei et al., 2021; Li et al., 2021) have shown that the translation performance can be significantly improved by increasing the model capacity. Considering the model performance, we adopt the Deep Encoder and Shallow Decoder architecture with wider FFN layer. For En-Zh pair, we adopt the Deep 35-6 big model as baseline model following (Wei et al., 2021). For En-Ja pair, in view of the training cost we choose the Deep 24-6 big model as baseline model following (Subramanian et al., 2021; Zhou et al., 2021). The details about the models are as follows:

- **Deep 24-6 model**: This model features 24-layer encoder, 6-layer decoder, 512 dimensions of word vector, 4096 dimensions of FFN, 16-head self-attention and uses Pre-Norm strategy(Wang et al., 2019).
- **Deep 35-6 big model**: This model features 35-layer encoder, 6-layer decoder, 768 dimensions of word vector, 3076 dimensions of FFN, 16-head self-attention and uses Pre-Norm strategy(Wang et al., 2019).

3.2 Data Augmentation

In this task, four data augmentation strategies are utilized to generate synthetic data, which have shown their effectiveness on improving the performance of NMT model in previous works (Wei et al., 2021; Zhou et al., 2021; Wang et al., 2021; Zeng et al., 2021).

Back-Translation (Sennrich et al., 2016) is the most commonly used data augmentation technique which generates pseudo parallel data by translating the target monolingual sentences into source language with a pre-trained target-to-source NMT model. Our back-translation is divided into three stages:

- Training an ensemble target-to-source NMT model with the provided bilingual parallel data.
- Translating the target monolingual sentences to source language with the pre-trained target-to-source NMT model to generate synthetic parallel data.
- Training models with the bilingual and synthetic parallel data in a ratio of 1:1.

[1] https://github.com/fxsjy/jieba
[2] http://taku910.github.io/mecab/

Forward-Translation (Wu et al., 2019) is another data generation technique. Different from back-translation, forward-translation translating the source monolingual corpus into target corpus with a pre-trained source-to-target NMT model. Here, the forward-translation is only applied to Ja → En direction.

Knowledge Distillation (Freitag et al., 2017) is a powerful technique to improve a student model by distilling knowledge from a group of teacher models. In our experiments, we first train several teacher models on the original bilingual data and generate synthetic training corpus with the ensemble teacher models. Then the student model is trained on the combination of the original and synthetic training set.

R2L Training Previous work (Liu et al., 2016) has shown that R2L training is an effective way to boost translation quality by addressing the error propagation problem in auto-regressive generation tasks. Following this strategy, we train an R2L model with the original source sentences and inverse target sentences and translate the source monolingual sentences into target sentences. In our experiment, we mix the synthetic data generated by both R2L and L2R models to for iterative joint training.

3.3 Domain Fine-tuning

Domain fine-tuning plays a key role in improving the model performance. Following Sun et al. (2019), we take previous development and test sets as in-domain data and fine-tune the models. For En ↔ Ja task, since previous development and test sets are too small to use, we search for additional in-domain data which are similar to the development sets. Specifically, we obtain the low-frequency domain-specific words in the development/test sets by employing the TF-IDF algorithm and filter sentences in the training set which contain those words.

3.4 Model Ensemble

Model ensemble is a widely used method in previous WMT shared tasks (Garmash and Monz, 2016), which can enhance the translation performance by combining the predictions of several models at each decoding step. In our work, we employ two kinds of ensemble methods, namely, checkpoint average and voting based ensemble. For checkpoint average, we average the top-5 checkpoints of each model according to their BLEU performance on the development set. While for model ensemble, we train several models with different architectures to increase the model diversity.

3.5 Post-editing

We apply post-editing to obtain the final translation outputs. For En → {Zh, Ja}, the post-editing includes removing the redundant spaces, converting punctuation to the language-specific format and replacing some of the English in the translation (such as the person name) with the English in the source sentence. For {Ja, Zh} → English, we de-tokenize the sentences with the Moses toolkit.

4 Experiments and Results

4.1 Settings

The implementation of our models is based on open-source fairseq (Ott et al., 2019) and we use sacreBLEU (Post, 2018) to measure system performances which is officially recommended. We select Transformer-big as the baseline for all tasks. The Zh ↔ En models are carried out on single NVIDIA 3090 GPU which has 24GB of memory and the Ja ↔ En models are carried out on 8 RTXA6000 GPUs each of which has 48GB of memory. For all tasks, the dropout probabilities are set to 0.1. We use the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.997$ (Zhou et al., 2021) during training. Table 2 lists the fairseq parameter setting in training.

Parameter	Zh ↔ En	Ja ↔ En
batch size	4096	8192
update-freq	4	2
learning rate	0.0005	0.002
warmup steps	4000	8000
save-interval-updates	4000	2000

Table 2: Fairseq parameter setting in training.

4.2 Zh ↔ En

For Zh ↔ En tasks, the training data consists of ParaCrawl v9, News Commentary v16, Wiki Titles v3, WikiMatrix, UN Parallel Corpus V1.0 (Ziemski et al., 2016) and CCMT Corpus. We take newsdev2017 as the development set and newstest2021 as the test set to tune the hyper-parameters. The training data is filtered by aforementioned methods and obtain the training data of 34.5M. The joint

System	En → Zh	Zh → En
Baseline	32.1	23.4
+ Back Translation	32.3(+0.2)	23.5(+0.1)
+ Checkpoint Average	32.8(+0.5)	24.2(+0.7)
+ Domain Fine-tuning	34.1(+1.3)	26.9(+2.7)
+ Ensemble	34.5(+0.4)	27.4(+0.5)
+ Post-edit	37.9(+3.4)	27.4

Table 3: The experimental result of En ↔ Zh task.

vocabulary with 32K words is generated by using sentencepiece(Kudo and Richardson, 2018). The officially provided back-translation data are not used in our experiments since no obvious improvements are obtained when adding it to the training set. The results of En ↔ Zh on newstest2021 are shown in Table 3.

We perform back-translation with the deep 35-6 big model in the target-to-source direction to generate the synthetic parallel data. Comparing with the baseline model, the back-translation technique leads to an improvement of 0.2 and 0.1 BLEU in En → Zh and Zh → En directions, respectively. The checkpoint average method brings another BLEU improvements of 0.5 and 0.7.

In the fine-tuning stage, we use previous WMT test sets as the in-domain data. We first perform fine-tuning on several different models with the combination of newstest2017-2019. Then we translate the in-domain data by the ensemble model to obtain pseudo parallel data and perform further fine-tuning on both the original and pseudo data. In our final submission, we add the newstest2020 and newstest2021 test set to the in-domain data. Domain fine-tuning is the most effective method in our experiment, which achieve an improvement of 1.3 and 2.7 BLEU scores in En → Zh and Zh → En directions, respectively.

We ensemble several models with better performance on the test set, in order to obtain more robust translation system. In our work, model ensemble further lead to a 0.4 and 0.5 BLEU improvement, respectively. Moreover, we apply post-editing to the translation outputs. It should be noted that post-editing can mainly improve the BLEU of En → Zh, which is about 3.4 BLEU. The punctuation format of Chinese translation has a great impact on BLEU. Finally, we obtain 37.9 BLEU scores in En → Zh direction and 27.4 BLEU scores in Zh → En direction.

4.3 Ja ↔ En

For Ja ↔ En tasks, we choose ParaCrawl v9, News Commentary v16, Japanese-English Subtitle Corpus (Pryzant et al., 2018), The Kyoto Free Translation Task Corpus (Neubig, 2011) and TED Talks as the training bilingual corpus. The final training bilingual corpus we used to train the model is about 20.1M. The source and target side each has a vocabulary with 32K words. We use the combination of newsdev2020 and newstest2020 as the development set and newstest2021 as the test set, respectively. Table 4 summarizes our results on newstest2021.

As shown in Table 4, all the four data augmentation methods improve the translation performance in both translation directions. We apply the deep 24-6 model to implement four data augmentation methods and We find that back-translation contributes the largest BLEU improvements (+4.1 BLEU) of the four data augmentation methods on En → Ja direction, while knowledge distillation performs best in the opposite direction (+2.1 BLEU). Moreover, we also evaluate the combination of the four data augmentation methods. In En → Ja direction, we combine the synthetic data from back-translation and R2L model with the original parallel data in a ratio of 0.5:0.5:1. By contrast, in Ja → En direction, we mix the synthetic data from back-translation, forward-translation and knowledge distillation with the original parallel data in a ratio of 0.5:0.5:0.5:1. The combination of multiple data augmentation methods brings 4.5 and 2.0 BLEU gains in En → Ja and Ja → En directions.

We further use newsdev2020, newstest2020 and selected in-domain data to fine-tune the model and achieve another 3.6 and 1.8 BLEU improvement in En → Ja and Ja → En directions, respectively. Then, the model ensemble further bring 1.1 and 0.6 BLEU improvement. Finally, we apply post-editing to the translation outputs and it further bring 0.2 and 0.1 BLEU improvement in En → Ja and Ja → En directions.

5 Conclusion

This paper presents the DUTNLP Translation systems for WMT22 General MT Task. Our main exploration is to improve the translation performance with the fine-grained data filtering, diverse model architectures, large-scale data augmentation and domain fine-tuning. The effectiveness of each method is demonstrated in our experiments. Model

System	En → Ja	Ja → En
Baseline	36.8	22.3
+ Back Translation	40.9	23.8
+ Forward Translation	39.5	24.2
+ Knowledge Distillation	38.9	24.3
+ R2L Training	39.7	23.4
+ BT+R2L	41.3(+4.5)	-
+ BT+FT+KD	-	24.3(+2.0)
+ Domain Fine-tuning	44.9(+3.6)	26.1(+1.8)
+ Ensemble	46.0(+1.1)	26.7(+0.6)
+ Post-edit	46.2(+0.2)	26.8(+0.1)

Table 4: The experimental result of En ↔ Ja task.

ensemble and post-editing are also used to further improve the performance of our system. Our constrained systems achieve 29.01, 63.87, 41.84, and 24.82 BLEU scores on Chinese → English, English → Chinese, English → Japanese, and Japanese → English, respectively.

Acknowledgements

We sincerely thank the reviewers for their insightful comments and suggestions to improve the paper. The authors gratefully acknowledge the financial support provided by the National Key Research and Development Program of China (2020AAA0108004) and the National Natural Science Foundation of China under (No.U1936109).

References

Ankur Bapna, Mia Chen, Orhan Firat, Yuan Cao, and Yonghui Wu. 2018. Training deeper neural machine translation models with transparent attention. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3028–3033, Brussels, Belgium. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Markus Freitag, Yaser Al-Onaizan, and Baskaran Sankaran. 2017. Ensemble distillation for neural machine translation. *CoRR*, abs/1702.01802.

Ekaterina Garmash and Christof Monz. 2016. Ensemble learning for multi-source neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1409–1418, Osaka, Japan. The COLING 2016 Organizing Committee.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hervé Jégou, and Tomás Mikolov. 2016. Fasttext.zip: Compressing text classification models. *CoRR*, abs/1612.03651.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

B. Li, Z. Wang, H. Liu, Q. Du, T. Xiao, C. Zhang, and J. Zhu. 2021. Learning light-weight translation models from deep transformer. In *National Conference on Artificial Intelligence*.

Bei Li, Ziyang Wang, Hui Liu, Yufan Jiang, Quan Du, Tong Xiao, Huizhen Wang, and Jingbo Zhu. 2020. Shallow-to-deep training for neural machine translation. *CoRR*, abs/2010.03737.

Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Agreement on target-bidirectional neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 411–416, San Diego, California. Association for Computational Linguistics.

Graham Neubig. 2011. The Kyoto free translation task. http://www.phontron.com/kftt.

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)*, pages 314–319, 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 NAACL-HLT 2019: Demonstrations*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

R. Pryzant, Y. Chung, D. Jurafsky, and D. Britz. 2018. JESC: Japanese-English Subtitle Corpus. *Language Resources and Evaluation Conference (LREC)*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Sandeep Subramanian, Oleksii Hrinchuk, Virginia Adams, and Oleksii Kuchaiev. 2021. NVIDIA nemo neural machine translation systems for english-german and english-russian news and biomedical tasks at WMT21. *CoRR*, abs/2111.08634.

Meng Sun, Bojian Jiang, Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Baidu neural machine translation systems for WMT19. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 374–381, Florence, Italy. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Longyue Wang, Mu Li, Fangxu Liu, Shuming Shi, Zhaopeng Tu, Xing Wang, Shuangzhi Wu, Jiali Zeng, and Wen Zhang. 2021. Tencent translation system for the WMT21 news translation task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 216–224, Online. Association for Computational Linguistics.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. 2019. Learning deep transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1810–1822, Florence, Italy. Association for Computational Linguistics.

Daimeng Wei, Zongyao Li, Zhanglin Wu, Zhengzhe Yu, Xiaoyu Chen, Hengchao Shang, Jiaxin Guo, Minghan Wang, Lizhi Lei, Min Zhang, Hao Yang, and Ying Qin. 2021. HW-TSC's participation in the WMT 2021 news translation shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 225–231, Online. Association for Computational Linguistics.

Lijun Wu, Yiren Wang, Yingce Xia, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2019. Exploiting monolingual data at scale for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4207–4216, Hong Kong, China. Association for Computational Linguistics.

Xianfeng Zeng, Yijin Liu, Ernan Li, Qiu Ran, Fandong Meng, Peng Li, Jinan Xu, and Jie Zhou. 2021. WeChat neural machine translation systems for WMT21. In *Proceedings of the Sixth Conference on Machine Translation*, pages 243–254, Online. Association for Computational Linguistics.

Shuhan Zhou, Tao Zhou, Binghao Wei, Yingfeng Luo, Yongyu Mu, Zefan Zhou, Chenglong Wang, Xuanjun Zhou, Chuanhao Lv, Yi Jing, Laohu Wang, Jingnan Zhang, Canan Huang, Zhongxiang Yan, Chi Hu, Bei Li, Tong Xiao, and Jingbo Zhu. 2021. The NiuTrans machine translation systems for WMT21. In *Proceedings of the Sixth Conference on Machine Translation*, pages 265–272, Online. Association for Computational Linguistics.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).