Tencent Translation System for the WMT21 News Translation Task

Longyue Wang[*] **Mu Li** **Fangxu Liu** **Shuming Shi** **Zhaopeng Tu**
Xing Wang **Shuangzhi Wu** **Jiali Zeng** **Wen Zhang**

Tencent AI Lab & Cloud Xiaowei

Abstract

This paper describes Tencent Translation systems for the WMT21 shared task. We participate in the news translation task on three language pairs: Chinese⇒English, English⇒ Chinese and German⇒English. Our systems are built on various Transformer models with novel techniques adapted from our recent research work. First, we combine different data augmentation methods including back-translation, forward-translation and right-to-left training to enlarge the training data. We also apply language coverage bias, data rejuvenation and uncertainty-based sampling approaches to select content-relevant and high-quality data from large parallel and monolingual corpora. Expect for in-domain fine-tuning, we also propose a fine-grained "one model one domain" approach to model characteristics of different news genres at fine-tuning and decoding stages. Besides, we use greed-based ensemble algorithm and transductive ensemble method to further boost our systems. Based on our success in the last WMT, we continuously employed advanced techniques such as large batch training, data selection and data filtering. Finally, our constrained Chinese⇒English system achieves 33.4 case-sensitive BLEU score, which is the highest among all submissions. The German⇒English system is ranked at second place accordingly.

1 Introduction

In this year's news translation task, our translation team at Tencent AI Lab & Cloud Xiaowei participated in three shared tasks, including Chinese⇒English, English⇒Chinese and German⇒English. We used the same data strategies, model architectures and corresponding techniques for all tasks.

We hypothesized that different models have their own strengths and characteristics, and they can complement each other. Thus, we built various advanced NMT models which mainly differ in training data and model architectures. These models (i.e. DEEP, LARGE and LARGE-FFN) are empirically designed based on Transformer-Deep which has proven more effective than the Transformer-Big models (Li et al., 2019). In addition to the original multi-head self-attention, we also proposed a mixed attention strategy by combining relative position with the original one, which extends the self-attention to efficiently consider representations of the relative positions. We use a variation of relative position, the random attention (RAN) (Zeng et al., 2021). As a results, we combined these models at transductive fine-tuning stage.

In terms of data augmentation, we adapt back-translation (BT) (Sennrich et al., 2016a), forward-translation (FT) (Zhang and Zong, 2016) and right-to-left (R2L) (Zhang et al., 2019) techniques to generate large-scale synthetic training data. Different from the standard back-translation, we add noise to the synthetic source sentence in order to take advantage of large-scale monolingual text. In addition, we used tagged BT mechanism (i.e. add a special token to the synthetic source sentence) to help the model better distinguish the originality of data. All the parallel data and a large amount of monolingual data are used in corresponding data augmentation methods, and finally we combine them together to build strong baseline models.

To enhance the domain-specific knowledge, we introduced approaches at both data and model levels. First, we employed a hybrid data selection method (Wang et al.) to produce different fine-tuning datasets. More specifically, we apply language coverage bias (Wang et al., 2021a), data rejuvenation (Jiao et al., 2020) and uncertainty-based sampling (Jiao et al., 2021) to select content-

[*] Corresponding author: vinnylywang@tencent.com. The other authors are in alphabetical order of last name.

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 216–224
November 10–11, 2021. ©2021 Association for Computational Linguistics

relevant and high-quality data from parallel and monolingual corpora. The news texts contain a number of sub-genres such as COVID-19 and government report. Thus, we fine-tuned a domain-specific model translate each sub-genre of text in the test set (i.e. "one domain one model").

We take advantage of the combination methods to further improve the translation quality. The "greedy search ensemble algorithm" (Li et al., 2019) is used to select the best combinations from single models. Furthermore, we propose an multi-model & multi-iteration transductive ensemble (m^{2}TE) method based on the translation results of the ensemble models. First, we divided models into two parts. Second, each part produced syntactic parallel testsets which is used to fine-tune another part of models. We repeated this procedure for N times.

This paper is structured as follows: Section 2 describes our advanced model architectures. We then present the data statistics and processing methods in Section 3. The methods and ablation study are detailed in Section 4 followed by final experimental results in Section 5. Finally, we conclude our work in Section 6.

2 Model Architecture

In this section, we mainly introduced three model architectures, which are empirically adapted from Transformer (Vaswani et al., 2017).

2.1 General Configurations

All models are implemented on top of the open-source toolkit Fairseq (Ott et al., 2019). Each single model is carried out on 8∼16 NVIDIA V100 GPUs each of which have 32 GB memory. We use the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. The gradient accumulation is used due to the high GPU memory consumption. We also employed large batching (Ott et al., 2018), which has significantly outperformed models with regular batch training. To speed up the training process, we conduct training with half precision floating point (FP16). We set max learning rate to 0.0007 and warmup-steps to 16000. All the dropout probabilities are set to 0.3. The detailed hyper-parameters of each model are summarized in Table 1.

2.2 Deep Model

Deep transformer has shown more effective performance than the TRANSFORMER-BIG models (Dou et al., 2018; Wang et al., 2019). We mainly modified the TRANSFORMER-BASE model by using a 40-layer encoder. To stabilize the training of deep model, we use the Pre-Norm strategy (Li et al., 2019), which is applied to the input of every sub-layer. The layer normalization was applied to the input of every sub-layer which the computation sequence could be expressed as: normalize → Transform → dropout → residual-add. The batch size is 5120 with 16 GPUs and "update-freq" is 1. We totally train models with 400K updates.

Module	DEEP	LARGE	LARGE-FFN
Encoder Layer	40	24	20
Attention Heads	8	16	16
Embedding Size	512	1024	1024
FFN Size	2048	4096	8192
Model Size	232M	514M	652M

Table 1: Hyper-parameters and model sizes of different models used in our systems.

2.3 Large Model

The large model is empirically designed based on TRANSFORMER-BIG models (Vaswani et al., 2017; Yang et al., 2020) with 24 encoder layers. More specifically, the batch size is 4096 with 8 GPUs and the "update-freq" is 4. We totally train models with 400K updates.

2.4 Large-FFN Model

We train Larger Transformers, the inner FFN dimension of which is twice as big as that of large Transformer. Specifically, in this setting, the FFN dimension is set to 8192. The number of encoder and decoder layers are 20 and 6 respectively. The number of head is 16. In addition to the original multi-head self-attention, we use a mixed attention strategy, where the random attention (Zeng et al., 2021) is combined with the original attention. In this way, the self-attention mechanism can efficiently consider representations of the relative positions, or distances between sequence elements. In training Large-FFN models, we set the batch size to 8192 toknes per GPU and the "update-freq" parameter is set to 8. The models are trained on 8 GPUs for about 3 days.

3 Data and Processing

3.1 Overview

Table 2 lists statistics of parallel and monolingual data we used in training our systems. The details

D.	L.	Parallel Data		Monolingual Data	
		# Sent.	# Word	# Sent.	# Word
In.	En	6.7M	128.2M	641.3M	13.1B
	Zh		116.0M	18.4M	466.0M
Out.	En	24.8M	613.8M	1.8B	35.5B
	Zh		550.3M	1.1B	28.4B
In.	En	58.5M	1.1B	641.3M	13.1B
	De		1.1B	353.8M	7.2B
Out.	En	3.4M	78.0M	1.8B	35.5B
	De		74.4M	417.0M	1.5B

Table 2: Data statistics of parallel and monolingual data. We combine sub-corpora according to in-domain (*In.*) and out-of-domain (*Out.*).

are as follows.

Chinese ⇔ English The bilingual data include all the available corpora provided by WMT2021: CCMT Corpus, News Commentary v16, ParaCrawl v7.1, Wiki Titles v3, UN Parallel Corpus V1.0 and WikiMatrix (except for Back-translated news). The monolingual English data consist of News crawl, News discussions, Common Crawl. The Chinese data consist of News crawl, News Commentary, Common Crawl and Extended Common Crawl.

English ⇒ German The bilingual data includes News Commentary v16, Europarl v10, ParaCrawl v7.1, Common Crawl, Wiki Titles v3, Tilde Rapid and WikiMatrix. For monolingual German data, we used News Crawl, News Commentary, Common Crawl and Extended Common Crawl. The monolingual English data are same as Chinese⇔English.

3.2 Pre-Processing

To process raw data, we applied a series of open-source/in-house scripts (Wang et al., 2014; Lu et al., 2014), including non-character filter, punctuation normalization, and tokenization/segmentation. The English and German languages are tokenized by Moses toolkit,[1] while the Chinese sentences are segmented by Jieba.[2] Furthermore, we generated subwords via BPE (Sennrich et al., 2016b) with 35K merge operations. The BPE models are trained on all the data in corresponding parallel and monolingual corpora instead of only parallel data. The vocabulary sizes of Chinese⇔English are 59100 and 48772, respectively. The vocabulary sizes of English⇒German are 41812 and 40948.

3.3 Filtering

To improve the quality of data, we filtered noisy sentences (pairs) according to their characteristics in terms of language identification, duplication, length, invalid string and traditional-simplified Chinese conversation. First, we filtered sentences whose language identification is invalid especially for English⇒German. Second, we removed similar sentences by comparing MD5 values of skelectons (i.e. removing stop words from sentences). About length, we filter out the sentences with length longer than 150 words. For more noisy corpora (e.g. ParaCrawl), we added hard filtering rules on special symbol, digital number, word length, punctuation number, HTML tags. Regarding bingling data, we further considered source-target ratio. For instance, the word ratio between the source and the target must not exceed 1:1.3 or 1.3:1. According to our observations, our method can significantly reduce noise issues including misalignment, translation error, illegal characters, over-translation and under-translation.

After filtering noisy training data, we used several data manipulation approaches to further improve the quality of the training data. We first followed Wang et al. (2021a) to identify the original languages of the bilingual sentence pairs, and explicitly distinguished between the source- and target-original training data using the bias-tagging strategy. We also identified the inactive training examples which contribute less to the model performance, rejuvenated them with self-training (Jiao et al., 2020). For the data augmentation with back-translation and forward-translation, we selected the most informative monolingual sentences by computing the uncertainty of monolingual sentences using the bilingual dictionary extracted from the parallel data (Jiao et al., 2021).

3.4 Evaluation

We regarded the WMT2019 test set as the validation set, and WMT2020 test set as the test set for all experiments. We ranked checkpoints according to either loss or BLEU on validation set. We used sacreBLEU score[3] as our evaluation metrics which is officially recommended. We also con-

[1] https://github.com/moses-smt/mosesdecoder/tree/master/scripts/tokenizer/tokenizer.perl.

[2] https://github.com/fxsjy/jieba.

[3] https://github.com/mjpost/sacrebleu.

#	Method	WMT20 Data			WMT21 Data		
		Data	*WMT19*	*WMT20*	*Data*	*WMT19*	*WMT20*
1	TRANSFORMER-DEEP	12.4M	30.1	30.2	31.5M	32.3	32.3
2	+ Forward-Translation	22.8M	33.1	31.8	49.9M	34.5	33.2
3	+ Back-Translation	32.4M	29.6	28.4	61.5M	32.4	32.5
4	+ Right-to-Left Training	24.8M	33.2	31.6	81.4M	34.4	33.1
5	+ 2 + 4	45.6M	33.9	32.2	99.8M	35.3	34.0
6	+ 2 + 3 + 4	58.0M	33.6	32.3	129.8M	35.5	34.3

Table 3: Effects of data augmentation methods on Chinese⇒English translation task. We used generally the same amount of monolingual data with the bilingual corpus. We used the DEEP model trained on the original bilingual data to construct the synthetic data, which is used together with the bilingual data to train the NMT models.

ducted post-processing such as *detokenizer.perl* on system output before sacreBLEU.

4 Method and Ablation Study

In this section, we conducted a comprehensive ablation study of the techniques used in this competition. We reported results on the Chinese⇒English task using the constrained data.

4.1 Data Augmentation

In this evaluation, we used three commonly-used data augmentation methods, namely *back-translation* (BT), *forward-translation* (FT) and *right-to-left training* (R2L), to exploit the useful monolingual data. All the synthetic parallel data is used together with the original parallel data to train NMT models.

Back-Translation This method first trains an intermediate target-to-source NMT system, which is used to translate monolingual target sentences into source language. Then the synthetic parallel corpus is used to train models together the bilingual data. In this work, we apply the noise back-translations method as introduced in Lample et al. (2018). When translating monolingual data we use an ensemble of two models to get better source translations. We follow Edunov et al. (2018) to add noise to the synthetic source data. Furthermore, we use a tag at the head of each synthetic source sentence as Caswell et al. (2019) does. To filter the pseudo corpus, we translate the synthetic source into target and calculate a Round-Trip BLEU score, the synthetic pairs are dropped if the BLEU score is lower than 30.

Forward-Translation This method is similar to BT but performs in a reverse manner. Recent studies showed that back-translation harms the translation performance, while forward-translation improves the performance (Edunov et al., 2020; Marie et al., 2020). Our preliminary experiments reconfirm their findings. Accordingly, we use forward-translation to construct the synthetic parallel data by translating the monolingual source sentences by the source-to-target NMT model, which is trained on the original bilingual data.

Right-to-Left Training The approach is proposed to address the error propagation problem in autoregressive generation task (Zhang et al., 2019). The main idea is to improve the agreement between translations generated by Right-to-Left (R2L) models and Left-to-Right (L2R) models. Following this work, we translate the source-side sentences in both parallel and monolingual corpora with both a R2L model and a L2R model, and use the translated pseudo corpus to improve the L2R model. For the right-to-left training, we trained another DEEP model on the bilingual data, whose target side is reversed. We drop the pseudo parallel data if the BLEU score lower than 15.

Experimental Results As shown in Table 3, we systematically investigated effects of 1) WMT20/WMT21 training data and 2) individual/combined data augmentation methods on Chinese⇒English translation task. For a comparison between the different training corpora of WMT20 and WMT21, we also reported results on the WMT20 training data ("WMT20 Data") used in last year (Wu et al., 2020b), and it consists of 12.4M sentence pairs after filtering. As seen, WMT21 extended around 19M sentence pairs, which improves the baseline model by +2.1 BLEU points. About data augmentation methods,

Source	Description	# Sent.
WMT	17-, 18-, 19-Dev/Test	7,466
	Source-Original	3,963
	Data Selection	1,000
	Data Augmentation	10,000
CWMT	08-, 09-, 11-Test	19,658
	Source-Original	8,036

Table 4: Statistics of data used for fine-tuning. “Source-Original” (SO) and “Data Selection” (DS) means respectively selecting source-original and domain-relevant examples from the whole WMT test sets. “Data Augmentation” indicates selecting data from the whole training corpus as extended data.

we selected domain-relevant and high-quality sentences from all available monolingual data as listed in Table 2. To construct the new training data (i.e. combining authentic and synthetic data), we selected the same amount of monolingual data with the bilingual corpus. As seen, individually using FT and R2L can significantly improve the baseline model by around +1 BLEU point. About BT, we fount that it failed to outperform baseline in “WMT20 Data” while performs slightly better than baseline in “WMT21 Data”. Finally, we trained the NMT models on the WMT21 training data augmented with the synthetic data generated by different data augmentation methods (up to 99.8M sentence pairs in total). We can further improve the performance by combining them together, demonstrating complementarity of different methods.

4.2 Fine-Tuning

We use in-domain finetune to further improve the model performance, which has proven effective on the WMT19∼20 news translation tasks (Sun et al., 2019; Meng et al., 2020; Li et al., 2020; Wu et al., 2020b). We construct different types of finetune data with the following approaches. Table 4 lists the statistics of data used for fine-tuning.

Previous Test Sets We follow the common practices to use WMT test sets in previous years as the finetune data. Specifically, we use WMT2017 development set, WMT2017 test set, WMT2018 and WMT19 test set.[4] Previous studies have shown that current NMT models suffer from the *language coverage bias* problem, which indicates the content-dependent differences between sentence pairs originating from the source and target languages, because the target-original data[5] can not improve translation performance (Wang et al., 2021a). Accordingly, we select the source-original examples (SO) from the test sets as the finetune data. Besides the WMT test sets, we also use the test sets from the CWMT competitions, which are available in the released data of WMT21 competition. In the CWMT testsets, each source sentence has four references, therefore we construct four sentence pair for each instance in the CWMT test sets.

Finetune	W19	W20
None	35.3	34.0
WMT	44.3	35.5
+ CWMT	42.3	34.9
WMT (SO)	43.4	35.7
+ CWMT (SO)	43.3	35.1
+ DS	44.9	35.4
+ DA	42.5	35.8
+ ODOM	n/a	36.1

Table 5: Finetune results on the corresponding datasets.

In-Domain Training Data We employed data selection and data augmentation methods to select in-domain data from WMT/CWMT test sets and training corpus, respectively. More specifically, we employed BM25 algorithm to select relevant sentence pairs by regarding source-side of WMT20 test set as queries. As shown in Table 4, the “Data Selection” is a subset of WMT test sets. On the other hand, we extend the finetuning set by selecting in-domain data from the tranining corpus. We further use the RT and R2L approaches in Section 4.1 to augment the finetune data with the Transformer-Deep model. Since the data augmentation approaches only require source-side sentences, we also construct the synthetic data for the WMT19 and WMT20 test sets.[6] We finetune the NMT model on the mixture of the additional synthetic corpus and the selected previous test sets.

One Domain One Model Li et al. (2020) argued that low-frequency words contain more domain information than high-frequency words, since low-

[4] In our final submission, we include WMT2019 and WMT2020 test sets in the fine-tune data.

[5] Target-original data are sentence pairs that are translated from the target language into the source language.

[6] In the final submission, we augment the WMT21 test set.

frequency words are mostly domain-specific nouns, etc., which may indicate the topic directly. Therefore, they adapt the TF-IDF algorithm to search and filter on the whole training set and then use them to train domain-specific models. We automatically to assigned domain labels to each source-side document in the test set. First, we used K-means clustering to obtain keywords of each document. Then, we proposed a rule-based method to classify each document in three categories: COVID-19, government report and other. In this experiment, we only focused on two specific domains and thus we trained two domain-specific models to translate COVID-19 and government report documents, respectively. The other documents are still dealt with a general-domain model.

Experimental Results As shown in Table 5, we investigated effects of different fine-tuning methods on Chinese⇒English translation task. As seen, source-original data is more effective than combining non-source-original one into finetuning dataset (35.5 vs. 35.7 BLEU). However, the CMWT dataset instead decrease the BLEU scores (-0.6 BLEU). The data reduction ("+DS") and expansion ("+DA") methods can not further improve the performance of baseline model (-0.3 and + 0.1 BLEU). Encouragingly, the "One Domain One Model" method can significantly improve the baseline model by +0.4 BLEU point.

4.3 Model Ensemble

Model ensemble is a widely used technique in previous WMT shared tasks, which can boost the performance by combining the predictions of several models at each decoding step (Li et al., 2019; Sun et al., 2019; Wang et al., 2018). In our work, we use two kinds of ensemble methods and finally the two are combined for further improvements.

Checkpoint Average For one model (same architecture and training data), we stored checkpoints according to their BLEU scores (instead of PPL or training time) on validation set. Then we combined top-L checkpoints (generate a final checkpoint) by averaging their weights to avoid stochasticity. To combine different models, we further ensembled the averaged checkpoint of each model. In our empirical experiments (Wang et al., 2020a), we find that this hybrid combination method outperforms solely combining checkpoints or models in terms of robustness and effectiveness.

Algorithm 1: Multi-Model & Multi-Iteration Transductive Ensemble

```
Input: Single Model $M_n$,
       In-domain Seed $D$={$D_s$, $D_t$},
       Ensemble $N$ models $E_N$.
Output: New Model $M_n^{'}$
t := 0
while not convergence do
    Translate $D_s$ with $E_N$ and get $D_t^{E_N}$
    Train $M_n$ on $D \cup D^{E_N}$ and get $M_n^{'}$,
      then $M_n = M_n^{'}$
    $t := t + 1$
end
```

Greedy Based Ensemble This method is proposed by Li et al. (2019), which adopts an easy operable greedy-base strategy to search for a better single model combinations on the development set. For more detail, please refer to the original paper. We also train single models with different hyper parameters to ensure the diversity. We refer to this method as Ensemble in the following.

Multi-Model & Multi-Iteration Transductive Ensemble Transductive ensemble (TE) is proposed by Wang et al. (2020b). The key idea is that source input sentences from the validation and test sets (in-domain seed) are firstly translated to the target language space with multiple different well-trained NMT models, which results in a pre-translated synthetic dataset. Then individual models are finetuned on the generated synthetic dataset. We propose an variation of TE, namely Multi-Model & Multi-Iteration TE (m^{2}TE) which is shown in Algorithm 1. The main difference from Iterative Transductive Ensemble (Wu et al., 2020b) is that E_N can be different groups of ensembled models (Deep, Large and Large-FFN models).

5 Final Results

In this section, we combined all the presented methods and techniques (detailed in Section 4) together and showed the final results in Table 6.

5.1 Chinese⇔English Translation Tasks

We train multiple single models in each settings. We found that the R2L method can significantly improve the baseline by about 1 BLEU score. It is surprising to find a gain of 2 BLEU improvement when combining all data augmentation meth-

System	Method	Zh⇒En		En⇒Zh		De⇒En	
		W19	*W20*	*W19*	*W20*	*W19*	*W20*
WMT2020 Competition Systems							
Meng et al.	*KD+Fine.+Ens.*	39.9	36.9	-	-	-	-
Li et al.	*XLM+Doc+Ens.+Fine.+Rerank*	-	-	40.5	49.1	-	-
Wu et al.	*KD+iteBT+Ens.*	-	-	-	-	43.8	43.5
Shi et al.	*KD+Ens.+Fine.+Rerank*	-	-	-	-	42.2	-
Wu et al.	*FT+R2L*	31.5	-	39.1	-	-	-
	FT+R2L+Fine.+Ens.	39.0	36.8	42.3	48.0	-	-
Our System	*BT+FT+R2L+Fine.+Ens.+Domain*	40.3	37.2	42.9	48.8	43.5	43.2

Table 6: Translation quality when combining all methods and techniques together.

ods. After we boost the in-domain corpus, we can further achieve 1∼2 more BLEU points on the different models, illustrating the effectiveness of fine-tuning. Specifically, we used corresponding development and test datasets and selected parallel data as in-domain corpus D. After training an NMT model M with the above methods, we finetune W on D with the same hyper parameters of training M. When testing on the WMT2020 test set, we achieve about 1.5 BLEU improvement. As the in-domain corpus is very limited, we propose a boosted finetune method by using the R2L training method to boost the finetune process. In our final submission, we add the WMT2020 test set to D, the batch size is set to 2048, the finetune finished after 3K training steps.

In our experiments, the ensemble models consists of 5 single models: 1 Deep, 2 Large, 2 Larger-Fnn models. The simple ensembled model can outperform the best single model by 0.5∼2.0 BLEU scores. We then apply transductive ensemble to each group of models and the performance achieves 36.8 BLEU on Chinese⇒English task. Finally, we employed two fine-grained domain-specific models to translate COVID-19 and government report texts, respectively. This can further improve the model by +0.5 BLEU point. We also find that the single models that applied TE cannot bring further improvement to ensemble results. We do not apply re-ranking to this task, as we find that the improvement is insignificant.

5.2 German⇔English Translation Tasks

The baseline model are trained on bilingual data and R2L data. This boosts the BLEU score from 41.6 to 42.1. After adding BT and FT, we further improve the BLEU score by 1.3 BLEU scores.

For finetuning English⇒German models, we select the document whose source side is originally in German from all previous development and test dataset as in-domain corpus D. Single models are trained with the above methods are then fine-tune on D for one epoch with a fixed learning rate of 1e-4. In our final submission, the WMT2020 test set is added to D for better performance improvement. The fine-tuning can further achieve 0.93 BLEU improvement on the Deep model.

In this task, the ensemble models consists of 3 single models: 1 Deep, 1 Large, 1 Larger-Fnn models. The ensemble models outperform the best single model by 1.5 BLEU scores. Furthermorem we apply a rule-based post-processing procedure on punctuation and this can improve the BLEU score on development set by 0.5 point.

5.3 Official Results

The official automatic results (in terms of sacreBLEU) of our submissions for WMT 2021 are presented in Table 7. Among participated teams, our primary systems achieve the first and the second BLEU scores on Chinese⇔English and German⇔English, respectively. The experimental results demonstrates that our models can achieve the state-of-the-art performance.

In the future, we will integrate these useful techniques in the Tencent TranSmart (Huang et al., 2021), Mr. Translator (`https://fanyi.qq.com`), and Tencent Simultaneous Translation systems.

6 Conclusion

This paper presents the Tencent Translation systems for WMT2021 news translation tasks. We investigate various deep architectures to build strong baseline models. Then popular data augmentation

System	Zh-En	En-Zh	De-En
Best Official	33.4	36.9	35.0
Our System	33.4	36.5	34.9

Table 7: Official sacreBLEU scores of our submissions for WMT21 news task. The "Best Official" denotes the best performance among all participant teams.

methods such as BT, FT and R2L are combined to improve their performances. We demonstrate that in-domain fine-tuning and fine-grained domain modelling are effective to further improve domain-specific quality. Besides, our proposed greed-based ensemble algorithm and transductive ensemble method play key roles in our systems. Among participated teams, our primary systems achieve the first and the second BLEU scores on Zh⇒En and De⇒En, respectively. In the future, we will adopt useful methods to our advanced non-autoregressive translation models (Ding et al., 2021b,a) and investigate the effects of pre-training on NMT (Liu et al., 2021a,b).

It is worth mentioning that most advanced technologies reported in this paper are also adapted to our systems for biomedical translation task (Wang et al., 2021b), which achieve three 1st ranks in German/French/Spanish⇒English tasks.

References

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *WMT*.

Liang Ding, Longyue Wang, Xuebo Liu, Derek F Wong, Dacheng Tao, and Zhaopeng Tu. 2021a. Rejuvenating low-frequency words: Making the most of parallel data in non-autoregressive translation. In *ACL*.

Liang Ding, Longyue Wang, Xuebo Liu, Derek F. Wong, Dacheng Tao, and Zhaopeng Tu. 2021b. Understanding and improving lexical choice in non-autoregressive translation. In *ICLR*.

Zi-Yi Dou, Zhaopeng Tu, Xing Wang, Shuming Shi, and Tong Zhang. 2018. Exploiting deep representations for neural machine translation. In *EMNLP*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *EMNLP*.

Sergey Edunov, Myle Ott, Marc'Aurelio Ranzato, and Michael Auli. 2020. On the evaluation of machine translation systems trained with back-translation. In *ACL*.

Guoping Huang, Lemao Liu, Xing Wang, Longyue Wang, Huayang Li, Zhaopeng Tu, Chengyan Huang, and Shuming Shi. 2021. Transmart: A practical interactive machine translation system. *arXiv preprint arXiv:2105.13072*.

Wenxiang Jiao, Xing Wang, Shilin He, Irwin King, Michael Lyu, and Zhaopeng Tu. 2020. Data rejuvenation: Exploiting inactive training examples for neural machine translation. In *EMNLP*.

Wenxiang Jiao, Xing Wang, Zhaopeng Tu, Shuming Shi, Michael R Lyu, and Irwin King. 2021. Self-training sampling with monolingual data uncertainty for neural machine translation. In *ACL*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *ICLR*.

Bei Li, Yinqiao Li, Chen Xu, Ye Lin, Jiqiang Liu, Hui Liu, Ziyang Wang, Yuhao Zhang, Nuo Xu, Zeyang Wang, Kai Feng, Hexuan Chen, Tengbo Liu, Yanyang Li, Qiang Wang, Tong Xiao, and Jingbo Zhu. 2019. The niutrans machine translation systems for WMT19. In *WMT*.

Zuchao Li, Hai Zhao, Rui Wang, Kehai Chen, Masao Utiyama, and Eiichiro Sumita. 2020. Sjtu-nict's supervised and unsupervised neural machine translation systems for the WMT20 news translation task. In *WMT*.

Xuebo Liu, Longyue Wang, Derek F Wong, Liang Ding, Lidia S Chao, Shuming Shi, and Zhaopeng Tu. 2021a. On the complementarity between pre-training and back-translation for neural machine translation. In *EMNLP*.

Xuebo Liu, Longyue Wang, Derek F Wong, Liang Ding, Lidia S Chao, Shuming Shi, and Zhaopeng Tu. 2021b. On the copying behaviors of pre-training for neural machine translation. In *ACL*.

Yi Lu, Longyue Wang, Derek F Wong, Lidia S Chao, and Yiming Wang. 2014. Domain adaptation for medical text translation using web resources. In *WMT*.

Benjamin Marie, Raphael Rubino, and Atsushi Fujita. 2020. Tagged back-translation revisited: Why does it really work? In *ACL*.

Fandong Meng, Jianhao Yan, Yijin Liu, Yuan Gao, Xianfeng Zeng, Qinsong Zeng, Peng Li, Ming Chen, Jie Zhou, Sifan Liu, and Hao Zhou. 2020. Wechat neural machine translation systems for WMT20. In *WMT*.

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. *NAACL*.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *WMT*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *ACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *ACL*.

Tingxun Shi, Shiyu Zhao, Xiaopu Li, Xiaoxue Wang, Qian Zhang, Di Ai, Dawei Dang, Xue Zhengshan, and Jie Hao. 2020. Oppo's machine translation systems for WMT20. In *WMT*.

Meng Sun, Bojian Jiang, Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Baidu neural machine translation systems for WMT19. In *WMT*.

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 *NIPS*.

Longyue Wang, Yi Lu, Derek F Wong, Lidia S Chao, Yiming Wang, and Francisco Oliveira. 2014. Combining domain adaptation approaches for medical text translation. In *WMT*.

Longyue Wang, Zhaopeng Tu, Xing Wang, Li Ding, Liang Ding, and Shuming Shi. 2020a. Tencent AI Lab machine translation systems for the WMT20 chat translation task. In *WMT*.

Longyue Wang, Derek F Wong, Lidia S Chao, Yi Lu, and Junwen Xing. A systematic comparison of data selection criteria for smt domain adaptation. *The Scientific World Journal*, 2014.

Mingxuan Wang, Li Gong, Wenhuan Zhu, Jun Xie, and Chao Bian. 2018. Tencent neural machine translation systems for WMT18. In *WMT*.

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 *ACL*.

Shuo Wang, Zhaopeng Tu, Zhixing Tan, Shuming Shi, Maosong Sun, and Yang Liu. 2021a. On the language coverage bias for neural machine translation. In *Findings of ACL*.

Xing Wang, Zhaopeng Tu, and Shuming Shi. 2021b. Tencent ai lab machine translation systems for the WMT21 biomedical translation task. In *WMT*.

Yiren Wang, Lijun Wu, Yingce Xia, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. 2020b. Transductive ensemble learning for neural machine translation. In *AAAI*.

Liwei Wu, Xiao Pan, Zehui Lin, Yaoming Zhu, Mingxuan Wang, and Lei Li. 2020a. The volctrans machine translation system for WMT20. In *WMT*.

Shuangzhi Wu, Xing Wang, Longyue Wang, Fangxu Liu, Jun Xie, Zhaopeng Tu, Shuming Shi, and Mu Li. 2020b. Tencent neural machine translation systems for the WMT20 news translation task. In *WMT*.

Yilin Yang, Longyue Wang, Shuming Shi, Prasad Tadepalli, Stefan Lee, and Zhaopeng Tu. 2020. On the sub-layer functionalities of transformer decoder. In *EMNLP*.

Jiali Zeng, Shuangzhi Wu, Yongjing Yin, Yufan Jiang, and Mu Li. 2021. Recurrent attention for neural machine translation. In *EMNLP*.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *EMNLP*.

Zhirui Zhang, Shuangzhi Wu, Shujie Liu, Mu Li, Ming Zhou, and Tong Xu. 2019. Regularizing neural machine translation by target-bidirectional agreement. In *AAAI*.