NoahNMT at WMT 2021: Dual Transfer for Very Low Resource Supervised Machine Translation

Meng Zhang[1], **Minghao Wu**[2], **Pengfei Li**[1], **Liangyou Li**[1], **Qun Liu**[1]
[1] Huawei Noah's Ark Lab
[1] {zhangmeng92, lipengfei111, liliangyou, qun.liu}@huawei.com
[2] Monash University
[2] minghao.wu@monash.edu

Abstract

This paper describes the NoahNMT system submitted to the WMT 2021 shared task of Very Low Resource Supervised Machine Translation. The system is a standard Transformer model equipped with our recent technique of dual transfer. It also employs widely used techniques that are known to be helpful for neural machine translation, including iterative back-translation, selected finetuning, and ensemble. The final submission achieves the top BLEU for three translation directions.

1 Introduction

In this paper, we describe the NoahNMT system submitted to one of the WMT 2021 shared tasks. The shared task features both unsupervised machine translation and very low resource supervised machine translation. As our core technique is mainly suitable for low resource supervised machine translation, we participated in four translation directions between Chuvash-Russian (chv-ru) and Upper Sorbian-German (hsb-de).

Our core technique is called dual transfer (Zhang et al., 2021), which belongs to the family of transfer learning. It transfers from both high resource neural machine translation model and pretrained language model to improve the quality of low resource machine translation. During the preparation for the shared task, we conducted additional experiments that supplement the original paper, including the choice of parent language, the validation of Transformer big model, and the usage of dual transfer along with iterative back-translation.

In addition, we also applied proven techniques to strengthen the quality of our system, including selected finetuning and ensemble. Our final submission achieves the top BLEU on the blind test sets for three translation directions: chv→ru, ru→chv, and hsb→de.

2 Approach

In this section, we describe the techniques used in our system. Interested readers are encouraged to check out the original papers for further details.

2.1 Dual Transfer

We reproduced the illustration of dual transfer from the original paper (Zhang et al., 2021), as shown in Figure 1. This illustration shows the case of general transfer, where the high resource translation direction is A→B, and the low resource translation direction is P→Q. As discussed in the original paper, in many cases, it is possible to use shared target transfer (B=Q) or shared source transfer (A=P). Taking chv→ru as an example, we can choose en→ru as the high resource translation direction, resulting in an instance of shared target transfer. In this shared task, when training the high resource translation model, we always initialize the shared language side with the pretrained language model BERT (Devlin et al., 2019).

2.2 Iterative Back-Translation

Iterative back-translation (Hoang et al., 2018) is an extension of back-translation (Sennrich et al., 2016a). It can exploit both sides of monolingual data of a language pair, and produces translation models for both directions, which is suitable for this shared task.

The initial models for generating synthetic parallel data are produced by using dual transfer with low resource authentic parallel data. In each iteration of iterative back-translation, we use the latest model to greedily decode a disjoint subset of 4m monolingual sentences[1] to generate synthetic parallel data. Then a new model is trained on a mixture of authentic and synthetic parallel data. With the use of dual transfer, model training can start from

[1] For chv and hsb, all monolingual sentences are used in each iteration.

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 1009–1013
November 10–11, 2021.

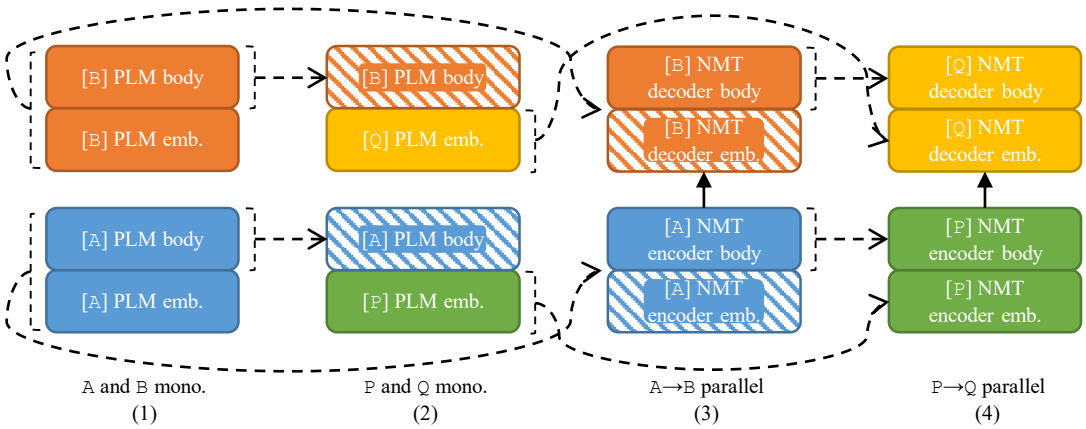

Figure 1: Dual transfer from pretrained language model and high resource A→B neural machine translation to low resource P→Q neural machine translation. Dashed lines represent initialization. Parameters in striped blocks are frozen in the corresponding step, while other parameters are trainable. Different colors represent different languages. Data used in each step is also listed.

language code	# sentence (pair)
cs-de	15m
hsb-de	0.1m
kk-ru	3.9m
en-ru	17m
chv-ru	0.7m
cs	90m
de	100m
hsb	0.8m
kk	17m
en	54m
ru	110m
chv	3m

Table 1: Training data statistics.

parent language	chv→ru BLEU
kk	18.47
en	18.61

Table 2: Test set BLEU for chv→ru, when the parent language is either kk or en (i.e. the parent translation direction is either kk→ru or en→ru). The translation model is Transformer base.

the initial parameters as shown in Step (4) of Figure 1. This has the additional benefit of reducing training time, because convergence is faster than training from random initialization.

2.3 Selected Finetuning

Selected finetuning aims to deal with the domain difference that may exist between the test set and the training set. Given the source side of the test set, we try to select similar source sentences from the training set, and then finetune the translation model on the selected subset of training sentence pairs.

We use BM25 (Robertson and Zaragoza, 2009) to calculate the similarity between two sentences for retrieval. The BM25 score between a query sentence Q and a sentence D in the corpus for retrieval $\mathcal{C}$ is given by

$$s(D,Q) = \sum_{i=1}^{L_Q} \frac{\text{IDF}(q_i) \cdot (k+1) \cdot \text{TF}(q_i, D)}{k \cdot \left(1 - b + b \cdot \frac{L_D}{L_{\text{avg}}}\right) + \text{TF}(q_i, D)},$$

where the query sentence Q is a sequence of L_Q subwords $\{q_i\}_{i=1}^{L_Q}$, $\text{IDF}(q_i)$ is the Inverse Document Frequency for q_i in the corpus $\mathcal{C}$, $\text{TF}(q_i, D)$ is the Term Frequency for q_i in the sentence D, L_D is the length of the sentence D, L_{avg} is the average length of the corpus $\mathcal{C}$, k and b are hyperparameters, which are set as 1.5 and 0.75, respectively.

Based on the BM25 score, we calculate the similarity between a source test sentence (as the query sentence) and the source sentences in the training set to obtain the top 500 sentences. After performing the selection for all the source test sentences, we merge them and remove duplicates to obtain the set for finetuning.

model	chv→ru	ru→chv	hsb→de	de→hsb
Transformer base	18.61	16.18*	55.60	55.98
Transformer big	19.24	17.12	56.10	57.12

Table 3: Test set BLEU for the four translation directions, using either Transformer base or Transformer big for dual transfer. *: The parent translation direction is ru→kk, and we did not train a Transformer base with ru→en as the parent, though the resulting ru→chv BLEU scores should be close based on the experiment in Section 4.1.

	runtime (hours)
BERT_{en}	143
BERT_{chv}	54
$\text{NMT}_{\text{en}\rightarrow\text{ru}}$	52
$\text{NMT}_{\text{chv}\rightarrow\text{ru}}$	14

Table 4: Runtime of each step in dual transfer for $\text{NMT}_{\text{chv}\rightarrow\text{ru}}$ with Transformer big.

3 Experimental Setup

3.1 Data

We collected allowed data for the involved languages and followed the same preprocessing pipeline of punctuation normalization and tokenization, using scripts from Moses[2]. The English monolingual data came from the English original side of ru-en back-translated news[3], but its automatic translation to Russian was discarded. The provided Chuvash-Russian dictionary was not used. Each language was encoded with byte pair encoding (BPE) (Sennrich et al., 2016b). The BPE codes and vocabularies were learned on each language's monolingual data, and then used to segment parallel data. We used 32k merge operations for all languages. After BPE segmentation, we discarded sentences with more than 128 subwords, and cleaned parallel data with length ratio 1.5. Training data statistics is provided in Table 1. Note that we experimented with Kazakh (kk) data (Section 4.1), but did not use it for our final submission. Evaluation on test sets is given by SacreBLEU[4] (Post, 2018), after BPE removal and detokenization.

3.2 Hyperparameters

We use Transformer (Vaswani et al., 2017) as our translation model, but with slight modifications that follow the implementation of BERT[5]. The absolute position embeddings are also learned as in BERT. The encoder and decoder embeddings are independent because each language manages its own vocabulary, but we tie the decoder input and output embeddings (Press and Wolf, 2017). We apply dropout with probability 0.1. We use LazyAdam as the optimizer. Learning rate warms up for 16,000 steps and then follows inverse square root decay. The peak learning rate is 5×10^{-4} for parent translation models, and 1×10^{-4} for child translation models. Early stopping occurs when the validation BLEU does not improve for 10 checkpoints. We set checkpoint frequency to 2,000 updates for parent translation models and 1,000 updates for child translation models. The batch size is 6,144 tokens per GPU and 8 NVIDIA V100 GPUs are used.

Hyperparameters for BERT are the same as in the original paper (Zhang et al., 2021).

For selected finetuning, we use stochastic gradient descent as the optimizer, and the learning rate is 1×10^{-5}. We finetune for 10,000 updates, and save a checkpoint every 100 updates. The checkpoint with the highest validation BLEU is kept.

4 Results

4.1 The Choice of Parent Language

In our preliminary experiments, we found it beneficial to use a closely related language as the parent language. It is clear that there are several factors that should be taken into account, such as the degree of closeness, and the amount of resource for training the parent model. For Upper Sorbian, Czech (cs) is closely related to it, and Czech-German has a good amount of parallel data, so we directly choose Czech as the parent language.

Chuvash, however, is a rather isolated language in the Turkic family. The closest language with usable data is Kazakh (kk), but the amount of parallel data for Kazakh-Russian is relatively small, and we found it to be quite noisy. Therefore, we considered

[2] https://github.com/moses-smt/mosesdecoder

[3] http://data.statmt.org/wmt20/translation-task/back-translation

[4] SacreBLEU signature: BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.4.12.

[5] https://github.com/google-research/bert

iteration	chv→ru	ru→chv	hsb→de	de→hsb
0	19.24	17.12	56.10	57.12
1	19.73	17.45	57.23	56.81
2	**20.42**	17.69	57.12	56.79
3	19.85	**17.81**	**57.72**	**57.47**
4	19.57	17.78	57.40	57.33
5	19.60	17.48	57.66	57.07

Table 5: Test set BLEU for the four translation directions with iterative back-translation. Iteration 0 is the Transformer big model in Table 3. Best BLEU scores are in bold.

method	chv→ru	ru→chv
before selected finetuning	20.42	17.69
after selected finetuning	20.55	18.03

Table 6: Test set BLEU to show the effect of selected finetuning.

model	hsb→de	de→hsb
best single	57.72	57.47
ensemble	58.54	58.28

Table 7: Test set BLEU to show the effect of ensemble.

using English (en) as the parent language of Chuvash. Even though English is unrelated to Chuvash and they use different scripts, English-Russian has more parallel data that can guarantee the quality of the parent model.

We conducted an experiment with Transformer base. Results in Table 2 indicate that English can serve as an eligible parent for Chuvash. Considering that we plan to use Transformer big for which data amount is likely to play a more important role, we decided to use English as the parent language for Chuvash.

4.2 The Effect of Transformer Big

The original paper (Zhang et al., 2021) evaluated dual transfer only with Transformer base. In this shared task, we scale up to Transformer big. We also face a more realistic setting where the monolingual data for the low resource languages (chv and hsb) are quite scarce. Therefore it is worth testing the effect of scaling up. Results in Table 3 show that Transformer big brings consistent improvements. We also report the runtime of each step in dual transfer for $\text{NMT}_{\text{chv}\rightarrow\text{ru}}$ with Transformer big in Table 4 for reference, but the numbers can vary depending on implementation and data size. In the following experiments and our final submission, we use Transformer big models.

4.3 Iterative Back-Translation

We ran five iterations of iterative back-translation. Results are shown in Table 5. The best BLEU scores are attained with two or three iterations. Another observation is that iterative back-translation brings larger improvements for chv→ru and hsb→de than ru→chv and de→hsb. This is probably because the monolingual data for chv and hsb are small in quantity.

4.4 Selected Finetuning

We only use selected finetuning for the chv-ru pair because parallel data for hsb-de is scarce. In order to test the effect of selected finetuning, we start from the models of Iteration 2 in Table 5. Results in Table 6 indicate that selected finetuning gives modest improvements.

4.5 Ensemble

We validate the effectiveness of ensemble on hsb→de and de→hsb, by performing ensemble decoding from the five models from iterative back-translation. Results in Table 7 demonstrate that ensemble gives BLEU improvements of about 0.8.

4.6 Final Submission

For chv→ru and ru→chv, we perform selected finetuning starting from the best models from iterative back-translation (Iteration 2 for chv→ru, Iteration 3 for ru→chv). Note that the selected training subsets are different from those in Section 4.4 because the selection is based on the source side of the blind test sets. We finetune five times

with different random seeds for model ensemble. For hsb→de and de→hsb, we ensemble the five models from iterative back-translation.

5 Conclusion

In this paper, we describe a series of experiments that contribute to our submission to the WMT 2021 shared task of Very Low Resource Supervised Machine Translation. These experiments, as well as the good results of the final submission, show that dual transfer can work in synergy with several widely used techniques in realistic scenarios.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative Back-Translation for Neural Machine Translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24.

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, Brussels, Belgium. Association for Computational Linguistics.

Ofir Press and Lior Wolf. 2017. Using the Output Embedding to Improve Language Models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. 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.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. 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)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

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 *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008.

Meng Zhang, Liangyou Li, and Qun Liu. 2021. Two Parents, One Child: Dual Transfer for Low-Resource Neural Machine Translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2726–2738.