Two Parents, One Child: Dual Transfer for Low-Resource Neural Machine Translation

Meng Zhang, Liangyou Li, Qun Liu
Huawei Noah's Ark Lab
{zhangmeng92, liliangyou, qun.liu}@huawei.com

Abstract

Neural machine translation suffers when parallel data for training is scarce. Previous works have explored transfer learning to assist training in low-resource scenarios. However, they transfer either from high-resource parallel data, or from monolingual data. In this work, we propose a framework to transfer multiple sources of auxiliary data, including both high-resource parallel data and monolingual data of involved languages. Knowledge in those sources is respectively encoded in a high-resource translation model and pretrained language models, and dually transferred to the low-resource translation model by our approach. Extensive experiments show that our approach yields consistent improvements over strong competitors for multiple translation directions. Furthermore, our approach still exhibits benefit on top of back-translation, making it a useful addition to practitioners' toolbox.

1 Introduction

Neural machine translation (NMT) has achieved remarkable success in recent years, but its quality critically hinges on large-scale parallel data. In the low-resource scenarios for most world languages and many domains, its performance usually deteriorates dramatically.

Although parallel data for some translation tasks may be difficult to obtain, monolingual data is usually within reach, and often comes in much larger quantity. Besides, parallel data for several high-resource languages is readily available. These corpora have been used in various methods to help training low-resource NMT. The most relevant method to our work is transfer learning.

Transfer learning starts with training a source task and then initializes the target task with the parameters. Recent advances in pretrained language models (PLM) like BERT (Devlin et al., 2019) can be seen as transfer learning, where language modeling is the source task for downstream target tasks. In low-resource NMT, pretrained language models have also provided noticeable improvements (Clinchant et al., 2019; Imamura and Sumita, 2019). As another source of transfer, high-resource NMT models have also been used for transfer learning low-resource NMT. Zoph et al. (2016) pioneered this direction with NMT based on recurrent neural networks, and coined the high-resource and low-resource models as parent and child models, respectively.

However, it is non-trivial to transfer from both PLMs and NMT models. This limitation constrains most existing transfer-learning-based low-resource NMT to a single source of auxiliary data, either monolingual or parallel.

In this paper, we propose a framework for transfer learning low-resource NMT that utilizes both monolingual data and high-resource parallel data (Table 1). Our approach encodes monolingual

approach	H		L	
	M	P	M	P
no transfer				✓
(Zoph et al., 2016)		✓		✓
(Kim et al., 2019)		✓	✓	✓
BERT2RND			✓	✓
BERT2BERT			✓	✓
(Kocmi and Bojar, 2018)		✓		✓
BBERT2BBERT			✓	✓
BBERT transfer	✓	✓	✓	✓
dual transfer (ours)	✓	✓	✓	✓

Table 1: An overview of data usage by approaches considered in this work (Section 4.3). H/L: high/low-resource language pair; M: monolingual; P: parallel. BBERT transfer checks all the boxes but uses data in a different way from ours.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2726–2738

knowledge in parent PLMs and translation knowledge in parent NMT models, and transfers both types of models to the child NMT model. Despite its simplicity, our approach shows consistent gains for multiple translation directions. Furthermore, it possesses several desirable features:

- It performs reasonably well even with little or no parallel data in the language pair of interest, alleviating the data issue for low-resource language pairs.
- It is complementary to back-translation, a strong data augmentation approach.
- It is agnostic to network architectures and thus applicable to any translation models.
- It is widely applicable to low-resource languages and can be applied to domain adaptation.
- The same high-resource NMT model can be used to transfer to future low-resource languages, saving computation.

2 Background

2.1 Transfer from Pretrained Language Models

The "pretraining-finetuning" paradigm has been highly successful for various natural language processing tasks. It first pretrains a language model through self-supervised learning, and then finetunes the model along with additional task-specific layers on downstream task data. Here, we exclude pretrained language models trained by sequence-to-sequence learning to simplify discussion[1]. Common pretrained language models include BERT and GPT (Brown et al., 2020).

In NMT with the encoder-decoder architecture (Sutskever et al., 2014; Bahdanau et al., 2015), the direct application of the "pretraining-finetuning" paradigm would be initializing the encoder with PLM and treating the decoder as task-specific layers. However, it is also possible to initialize the compatible modules in the decoder, leaving the cross attention module randomly initialized. Although initializing the decoder does not appear as useful, especially for high-resource language pairs (Rothe et al., 2020), it is not harmful either.

[1]Examples of such models include MASS (Song et al., 2019) and BART (Lewis et al., 2020). If desired, pretrained encoders in these models can be used in our approach.

2.2 Transfer from High-Resource Translation Models

Even though the Transformer model (Vaswani et al., 2017) has become more popular than recurrent neural networks for NMT, the transfer procedure proposed by Zoph et al. (2016) still applies as long as the parent model and the child model share the same architecture, which is typically the case. However, one problem still persists. Because the high-resource languages have different vocabularies from the low-resource ones, directly transferring the word embedding layer is not possible.

One way to circumvent this issue is to prepare a joint vocabulary of the involved languages that is shared between the parent and child NMT models (Kocmi and Bojar, 2018). Known as warm-start transfer (Neubig and Hu, 2018), this type of methods need to prepare a new joint vocabulary whenever a new low-resource model is on demand, and retrain both parent and child models. In contrast, cold-start transfer (Kocmi and Bojar, 2020) trains a universal parent NMT model that does not depend on child languages.

Kim et al. (2019) addressed the vocabulary mismatch for cold-start transfer by matching word embeddings across languages. They first learn monolingual word embeddings of the child language with e.g. skip-gram (Mikolov et al., 2013), and then learn a cross-lingual linear mapping to connect child monolingual word embeddings and pretrained parent NMT word embeddings. The child monolingual word embeddings can then be mapped to the parent word embedding space, and be used to initialize the child NMT word embeddings. The cross-lingual linear mapping relies on a bilingual lexicon to learn, which can be induced from parent and child language monolingual data by unsupervised methods like (Lample et al., 2018a).

Our approach also belongs to cold-start transfer in its usage of the parent NMT model. It addresses the vocabulary mismatch by design, without relying on monolingual word embeddings and bilingual lexica.

3 Approach

Our approach is a general framework for transferring from any high-resource language pair to any low-resource language pair, as long as data condition permits. Generally speaking, monolingual data and high-resource parallel data are available in large quantity. We first present the gen-

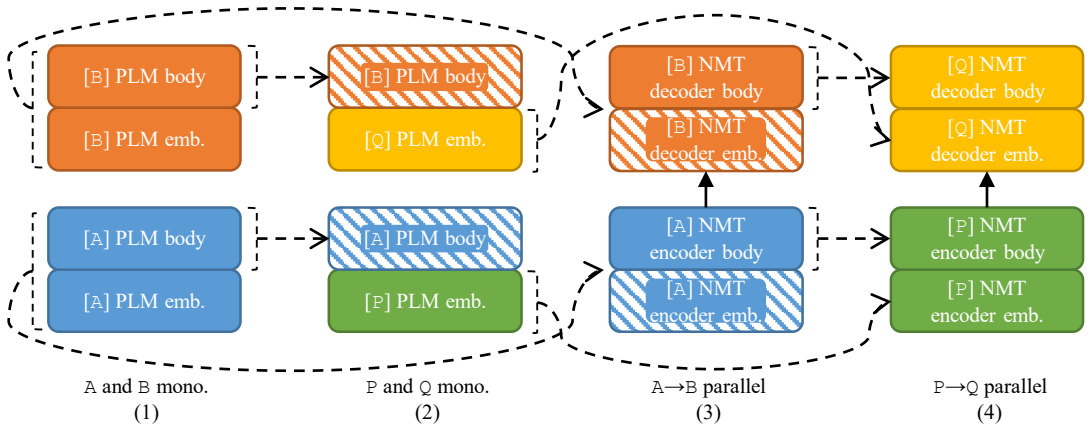

Figure 1: Dual transfer from PLM and high-resource A→B NMT to low-resource P→Q NMT. Dashed lines represent initialization. Parameters in striped blocks are frozen in the corresponding step, while other parameters are trainable. Different colors represent different languages. Data used in each step is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

eral case where we would like to transfer from the high-resource A→B to the low-resource P→Q, where capital letters denote languages. Then we discuss specific cases where some of the involved languages are the same.

3.1 General Transfer

Figure 1 shows the pipeline of our approach, consisting of four major steps, as detailed below.

(1) Train PLM_A and PLM_B on monolingual data of A and B separately.

(2) Train PLM_P and PLM_Q on monolingual data of P and Q as follows.

- Initialize PLM_P with PLM_A (except word embeddings); freeze parameters other than word embeddings.
- Initialize PLM_Q with PLM_B (except word embeddings); freeze parameters other than word embeddings.

(3) Train $\text{NMT}_{\text{A}\to\text{B}}$ on A→B parallel data as follows: Initialize NMT encoder with PLM_A, and decoder with PLM_B; freeze word embeddings during training.

(4) Replace word embeddings as follows to initialize $\text{NMT}_{\text{P}\to\text{Q}}$, and finetune on P→Q parallel data.

- Replace $\text{NMT}_{\text{A}\to\text{B}}$ encoder word embeddings with those in PLM_P.
- Replace $\text{NMT}_{\text{A}\to\text{B}}$ decoder word embeddings with those in PLM_Q.

Note that Steps (2) and (3) are independent of each other, and therefore can be done in parallel.

Intuitively, Step (2) learns word embeddings of P and Q that lie in the same semantic space of A and B, respectively. Because only word embeddings are trainable, they are forced to align with pretrained A and B body parameters to do language modeling (e.g. masked language model). In Step (3), $\text{NMT}_{\text{A}\to\text{B}}$ needs to learn translation based on the frozen A and B word embeddings space. With P and Q word embeddings swapped in place in Step (4), the body and embedding parameters can cooperate in a close semantic space, allowing finetuning to proceed smoothly.

Like (Kim et al., 2019), our approach solves the vocabulary mismatch issue by manipulation in the embedding space, allowing transfer between arbitrary languages, even with different scripts[2]. Each language now manages its own independent vocabulary. We also tie input and output embeddings of the decoder (Press and Wolf, 2017), so a single decoder embedding block is shown in Figure 1.

We can further generalize our approach by defining *transfer parameters* as those responsible for transforming input into continuous representations shared across languages. In Figure 1, the transfer parameters are simply word embeddings, but we may also use other sets of transfer parameters, e.g. word and position embeddings, or even lower layers of the body. In Step (2), only transfer parameters are trainable, while in Step (3), only non-transfer parameters are trainable, and initialization changes accordingly.

[2]We verified the effectiveness of our approach for transferring from fr→en to ru→en on in-house data.

Our approach defines a framework for transfer learning, which can be applied to various network architectures. For example, if we would like to train a low-resource RNN-based NMT, we can prepare RNN-based PLMs and a high-resource RNN-based NMT. In our experiments, we use Transformer for PLMs and NMT models.

3.2 Shared Target Transfer and Shared Source Transfer

In practice, it is a rare need to train on a low-resource language pair where both languages are low-resource. Typically one of the two languages would be high-resource, e.g. English. In this case, we can choose a high-resource language pair that shares this language on the same side, thereby simplifying our approach.

If the target language (Q) of the low-resource language pair (P→Q) is high-resource, we can choose a high-resource language pair (A→B) with that language as the target, i.e. B=Q. In this case, there is no vocabulary mismatch on the target side, so PLM_Q is no longer needed, and decoder word embeddings can be adjusted when training $\text{NMT}_{\text{A}\to\text{B}}$ in Step (3). PLM_B also becomes optional, and the randomly initialized decoder of $\text{NMT}_{\text{A}\to\text{B}}$ may learn sufficiently from abundant A→B parallel data.

Likewise, if the source language (P) is high-resource, we can let A=P. Then PLM_P is not needed, and encoder word embeddings are trainable in Step (3). PLM_A may also be dispensed with and the encoder of $\text{NMT}_{\text{A}\to\text{B}}$ is randomly initialized.

3.3 Domain Adaptation

By viewing a certain domain as a special language, our approach can also be applied to domain adaptation. In this case, A→B is a high-resource source domain, and P→Q is a low-resource target domain. By definition, this setting is general transfer, because neither B=Q nor A=P is possible due to domain difference, but typically they will be the same language, respectively.

4 Experimental Setup

We mainly verify our approach in the more realistic shared target and shared source transfer scenarios. We take German-English (de-en) as the high-resource language pair, while Estonian-English (et-en) and Turkish-English (tr-en) are the low-resource language pairs. Previous works mainly consider shared target transfer (Dabre et al., 2020), and we make extensive comparison in the experiment that transfers from de→en to et→en. We then verify on other translation directions, including shared source transfer, as well as general transfer, in which we consider an artificial setting of transferring from de→en to French→Spanish (fr→es). For domain adaptation we work on de→en, transferring from news domain to medical domain. We report SacreBLEU[3] (Post, 2018). Further details about data and hyperparameters can be found in Appendices B and C, respectively.

language code	# sentence (pair)
de-en	5.9m
et-en	1.9m
tr-en	207k
fr-es	10k
de-en medical	347k
en	94m
de	147m
et	139m
tr	100m
fr	4.1m
es	4.2m
en medical	4.0m
de medical	3.6m

Table 2: Training data statistics.

4.1 Data

We mainly use data from WMT 2018[4]. We use preprocessed parallel data for training NMT models. The provided development data includes multi-parallel data for several languages, which we use for fr→es. We collect monolingual data for the involved languages and follow the same preprocessing pipeline. Training data statistics is provided in Table 2. Each language is encoded with byte pair encoding (BPE) (Sennrich et al., 2016b). The BPE codes and vocabularies are learned on each language's monolingual data, and then used to segment parallel data. Following (Kim et al., 2019), we use 50k merge operations for English, and 20k for other languages. Sentences with more than 150 subwords are removed from NMT training.

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

[4] http://statmt.org/wmt18/translation-task.html

4.2 Hyperparameters

We use Transformer base as our NMT model, but with slight modifications that follow the implementation of BERT[5]. The absolute position embeddings are also learned as in BERT. We apply dropout with probability 0.1. Learning rate warms up for 16,000 steps and then follows inverse square root decay. The peak learning rate is 7×10^{-4} for the high-resource de–en. For other translation tasks, we grid search over $\{1, 3, 5\} \times 10^{-4}$ for each approach in every experiment, and keep the best model based on development BLEU. We use 8 GPUs for de–en, and 1 GPU otherwise. Other hyperparameters follow (Kim et al., 2019).

We train BERT as the PLM in our experiments, with the same number of layers and hidden size as Transformer base. The absolute position embeddings are learned up to 128. We only train with masked language modeling and dispense with next sentence prediction as in (Liu et al., 2019). We train for 480k steps with batch size 180 on 8 GPUs. The peak learning rate is 1.8×10^{-4}, and the number of warmup steps is 18,000.

Rothe et al. (2020) found that for the high-resource de–en pair, initializing the decoder with PLM has no advantage over random initialization. Therefore, we only used PLM_{de} for de→en, but for en→de, we used both PLM_{en} and PLM_{de} because the vocabulary mismatch is on the target side.

4.3 Baselines

We compare with the following approaches.

No transfer This baseline trains directly on the low-resource parallel data.

(Zoph et al., 2016) This approach transfers from the high-resource language pair. In the original paper, random parent word embeddings are used to initialize child word embeddings. We simply initialize child word embeddings with the truncated normal initializer.

(Kim et al., 2019) This approach transfers from the high-resource language pair and utilizes cross-lingual word embeddings. The authors also proposed other orthogonal data augmentation techniques, but we do not include them in our experiments.

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

BERT2RND This approach transfers from the source language PLM trained on monolingual data. By comparing with BERT2BERT, we can see if the finding in (Rothe et al., 2020) holds for low-resource language pairs.

BERT2BERT This approach transfers from the source and target language PLMs trained on monolingual data. Note that PLMs for BERT2BERT and BERT2RND are directly trained on monolingual data of P and Q, different from those obtained by Step (2) of our approach.

As discussed in (Kim et al., 2019), managing independent vocabularies for each language has the advantage of flexibility. However, many approaches rely on shared vocabulary. We nevertheless report their performance for reference.

(Kocmi and Bojar, 2018) This approach uses joint vocabulary of all the involved languages. It first trains the NMT model on the high-resource parallel data, and then finetunes it on the low-resource parallel data. It can be seen as a multilingual NMT in which high-resource performance does not matter. We experiment with transferring from de→en to et→en, thus involving three languages. We learn joint BPE with 90k merge operations.

BBERT2BBERT Multilingual PLMs usually rely on shared vocabulary, and bilingual BERT (BBERT) is an example trained on non-parallel data of two languages. We learn joint BPE with 70k merge operations for the source and target languages of the low-resource language pair, and the same vocabulary is used for the source and target sides of NMT. Otherwise this approach is the same as BERT2BERT. This is equivalent to XLM (Conneau and Lample, 2019) used for bilingual PLM and MT.

BBERT transfer Multilingual PLMs are often used for cross-lingual transfer. We apply this approach to NMT, transferring from de→en to et→en. First we train a de–et BBERT, and use it to initialize the encoder of $\text{NMT}_{\text{de}\to\text{en}}$. Then we train on de→en and finally finetune on et→en. We learn joint BPE with 40k merge operations for the de–et pair. This approach uses exactly the same data as ours because we do not use

approach	V	BLEU
no transfer	✓	21.76
(Zoph et al., 2016)	✓	21.07
(Kim et al., 2019)	✓	22.25
BERT2RND	✓	22.89
BERT2BERT	✓	23.44
(Kocmi and Bojar, 2018)	✗	23.58
BBERT2BBERT	✗	23.90
BBERT transfer	✗	24.03
dual transfer (word)	✓	**24.81**
dual transfer (word+position)	✓	24.28

Table 3: BLEU on et→en, with the best in bold. "✓" in the "V" column indicates independent vocabulary, while "✗" means the approach relies on shared vocabulary. Our approach (dual transfer) has two variants, with or without position embeddings in the transfer parameters.

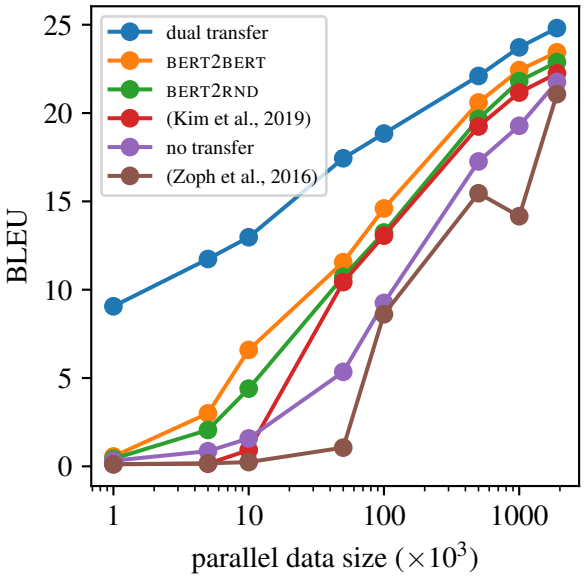

Figure 2: BLEU of different approaches with respect to the number of parallel et→en sentence pairs for training. We plot our approach with word embeddings as transfer parameters; additionally transferring position parameters performs similarly.

PLM_{en} when transferring from de→en to et→en.

In their experiments, Zoph et al. (2016) and Kim et al. (2019) only considered shared target transfer, and they found that freezing certain components of the decoder during finetuning can be beneficial. In our et→en experiment, we tried freezing the decoder word and position embeddings, and optionally self attention parameters, for their approaches, our approach, and BERT2BERT, but development set results revealed that the only setting which brought improvement was freezing word and position embeddings and self attention parameters for (Kim et al., 2019), possibly due to the relatively large size of et→en data. Therefore we only use it for (Kim et al., 2019) in our experiments.

5 Results

In this section, we first report extensive experiments on et→en before generalizing to other translation directions. We then present the performance of our approach when used in conjunction with back-translation and self training. Finally we demonstrate that our approach can be used for domain adaptation.

5.1 Results on et→en

Table 3 shows the BLEU scores for et→en. We report the following findings for this translation direction.

The approach in (Zoph et al., 2016) only uses high-resource parallel data for transfer, and the approach in (Kim et al., 2019) additionally uses low-resource monolingual data; their BLEU scores are close to the "no transfer" baseline. The approach in (Kocmi and Bojar, 2018) shows positive transfer from high-resource parallel data by forgoing the vocabulary flexibility and relying on joint vocabulary.

Using monolingual data, BERT2RND and BERT2BERT show notable improvement on the "no transfer" baseline. In this relatively low-resource setting, it appears useful to initialize the decoder with BERT, in contrast to de-en experiments in (Rothe et al., 2020).

We expected additionally transferring position embeddings to better deal with word order divergence across languages, but after comparing the two variants of our approach, we find no benefit in including position embeddings in the transfer parameters. Our approach with word embeddings as transfer parameters achieves best BLEU, which is a 3.05 improvement over the "no transfer" baseline, and 1.37 over BERT2BERT. Note that we did not use monolingual English data for our approach when the target language is English.

parallel data size ($\times 10^3$)	0	1	5	10	50	100	500	1000
dual transfer (word)	0.43	9.06	11.74	12.97	17.44	18.84	22.10	23.72
+freezing parent NMT encoder	**6.20**	8.82	11.58	12.76	16.62	18.50	21.69	23.59

Table 4: BLEU on et→en. Freezing the parent NMT encoder helps our approach to perform zero-shot translation.

approach	tr→en	en→et	en→tr	fr→es
no transfer	15.44	16.29	9.63	10.59
BERT2BERT	19.73	17.36	11.78	18.26
dual transfer (word)	**21.12**	**19.41**	**13.18**	**22.28**
dual transfer (word+position)	20.29	18.79	13.16	-

Table 5: BLEU on translation directions shown in columns, grouped by shared target transfer, shared source transfer, and general transfer. "-" means the experiment was not carried out.

approach	BLEU
no transfer	21.63 (-0.13)
dual transfer (word)	22.53 (-2.28)
dual transfer (word+position)	23.08 (-1.20)

Table 6: BLEU on et→en augmented with 4m self training data. Numbers in parentheses indicate differences from the corresponding approach trained on authentic parallel data.

5.2 Effect of Low-Resource Parallel Data Size

Arguably, the parallel training data for et→en is not quite low-resource. But it provides a good test bed for manually adjusting the data size to simulate various degrees of resource scarcity. We sample subsets of $\{1, 5, 10, 50, 100, 500, 1000\} \times 10^3$ parallel sentence pairs, and show BLEU of different approaches in Figure 2. We observe roughly monotonic trend of BLEU with respect to parallel data size, as expected. Our approach performs consistently better than baselines, and the gap is larger with fewer parallel sentence pairs. In the extremely low-resource setting of one thousand pairs, our approach still achieves BLEU close to 10, while all other approaches fail with BLEU close to 0.

5.3 Zero-Shot Translation

Our approach can also be modified slightly to perform zero-shot translation. We conjecture that in Step (3) of our approach, freezing the embeddings alone is insufficient to prevent encoder body parameters from drifting too far away. Therefore we try freezing the entire encoder in Step (3). This technique helps our approach to achieve a zero-shot BLEU score of 6.20, as shown in Table 4. However, it does not have advantage when parallel data is available.

5.4 Other Translation Directions

Table 5 shows the results that include shared target transfer, shared source transfer, and general transfer, comparing our approach with no transfer and BERT2BERT. Our approach consistently outperforms baselines. Previous works (Zoph et al., 2016; Kim et al., 2019) typically conducted experiments on shared target transfer only, and shared source transfer is considered more difficult (Kocmi, 2020), but our approach works well for shared source transfer, as well as general transfer. Also note that, we use the same de-en pair for all child languages from diverse language families, which demonstrates the robustness of our approach. It also highlights the advantage of independent vocabularies: We can prepare $\text{NMT}_{\text{de}\to\text{en}}$ and $\text{NMT}_{\text{en}\to\text{de}}$ for any future child language, while approaches like (Kocmi and Bojar, 2018) and BBERT transfer have to retrain with the high-resource language every time a new low-resource language is needed.

5.5 Back-Translation and Self Training

Back-translation (BT) (Sennrich et al., 2016a) and self training (ST) (Zhang and Zong, 2016) are data augmentation techniques that generate synthetic parallel data, using target language monolingual data and source language monolingual data respectively. We first experiment with ST for et→en. We use the "no transfer" $\text{NMT}_{\text{et}\to\text{en}}$ to translate 4m et monolingual data into en by greedy decoding, and merge with authentic parallel data. Results in Table 6 show that self training is not helpful for this experiment, and considerably lowers the BLEU of our approach.

approach	BT data size	BLEU
no transfer	4m	19.78 (+3.49)
dual transfer (word)	4m	21.74 (+2.33)
dual transfer (word+position)	4m	**22.34** (+3.55)
no transfer	130m	20.52 (+4.23)
dual transfer (word+position)	130m	22.23 (+3.44)

Table 7: BLEU on en→et augmented with 4m or 130m back-translation data. Numbers in parentheses indicate differences from the corresponding approach trained on authentic parallel data.

approach	BLEU
no transfer (child)	62.94
BERT2BERT (child)	64.33
finetuning (parent)	64.91
dual transfer (parent)	65.14
dual transfer (child)	**65.40**

Table 8: Domain adaptation results. The transfer parameters are word embeddings for dual transfer. "Parent" indicates using source domain (news) vocabulary, and "child" indicates using target domain (medical) vocabulary.

We then use the same synthetic parallel data for en→et, turning to the case of BT. The upper rows in Table 7 show that BT is highly beneficial for both the baseline and our approach. Encouraged by this, we further try using all 130m et monolingual data with the maximum of 80 tokens and 100 subwords per line. We upsample authentic data to have a 1:4 ratio with synthetic data, following (Caswell et al., 2019). The lower rows in Table 7 show that more BT data can further improve the "no transfer" baseline, though the small improvement appears unattractive considering the cost. As for our approach, going from 4m to 130m yields no gain. Besides, our approach with 4m BT still surpasses no transfer with 130m BT. We conjecture that our approach can work complementarily with a manageable amount of BT data, reducing the need to decode and train on a huge data size.

Finally, note that we use the "no transfer" $\text{NMT}_{\text{et}\rightarrow\text{en}}$ to generate all synthetic parallel data in our experiments. In practice, the model produced by our approach can be used for decoding, which should result in higher-quality synthetic data. This might also be the reason that ST hurts our approach more than the "no transfer" baseline.

5.6 Domain Adaptation

A simple and effective approach to domain adaptation is finetuning source domain NMT on target domain data (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016). This approach is possible because directly inheriting parent NMT vocabulary is acceptable for domain adaptation. In other words, this is a special case of (Kocmi and Bojar, 2018) where child vocabulary largely overlaps with parent vocabulary. However, our approach allows using a dedicated vocabulary for the target domain. In this case, we learn BPE with the same number of merge operations as the source domain on target domain monolingual data. Table 8 shows that our approach can surpass the baselines, especially with the child (medical domain) vocabulary.

6 Related Work

Low-resource NMT has been researched from many perspectives. Exploiting auxiliary data has been verified to be helpful by various approaches, including data augmentation like back-translation (Sennrich et al., 2016a; Xia et al., 2019), transfer learning as focused in our work, meta-learning (Gu et al., 2018), semi-supervised learning (Cheng et al., 2016), or even unsupervised NMT (Artetxe et al., 2018; Lample et al., 2018b; Chronopoulou et al., 2020).

Transfer learning usually utilizes a single source of knowledge. When multiple sources are available, transfer learning may be applied in a cascaded fashion (Lakew et al., 2018), but catastrophic forgetting may need to be addressed. Maimaiti et al. (2019) proposed multi-round transfer by performing transfer learning for several rounds on multiple high-resource language pairs.

Multilingual NMT (Johnson et al., 2017; Dabre et al., 2019) aims to perform translation for multiple translation pairs in a single model, and positive transfer towards low-resource language pairs

typically occurs. In our experiment, we have considered a variant that solely focuses on the low-resource pair (Kocmi and Bojar, 2018; Nguyen and Chiang, 2017).

Outside NMT, Artetxe et al. (2020) proposed a similar partial freezing approach to transferring BERT cross-lingually. As they worked on BERT (Transformer encoder) for natural language understanding tasks, several differences from our work arise. First, we need to consider the initialization of decoder for NMT, and for the shared source case, we need to deal with vocabulary mismatch on the decoder side. Second, we find that additionally transferring position embeddings is not helpful in our experiments. Third, our approach can outperform BBERT transfer, whereas they observe slightly lower performance in their experiments.

7 Conclusion and Future Work

In this work, we propose a framework for transferring from both pretrained language models and neural machine translation models, so that both monolingual data and high-resource parallel data can be used to assist low-resource training. Our approach shows consistent usefulness in a variety of experiments, while also enjoying the flexibility of independent vocabulary.

Recently, a deep encoder and shallow decoder architecture is shown to have comparable translation quality with faster decoding speed (Kasai et al., 2020). While our approach can be applied to such architectures, a shallow decoder means that transferring on the decoder side will be limited by the shallow PLM, which is particularly severe for shared source transfer. In future work we would like to investigate how to work around this issue.

Acknowledgments

We thank Wenyong Huang for his help with the BERT code. We also thank anonymous reviewers for their helpful feedback. We thank MindSpore[6] for the partial support of this work, which is a new deep learning computing framework.

Ethical Considerations

In this paper we have reported experiments with Transformer base, and the computation cost is shown in Table 11 of Appendix A. Scaling up the model may bring further improvement at the cost of computation. However, we have highlighted the benefit of cold-start transfer: Trained high-resource NMT models can be used for future transfer. For example, we can reuse $\text{NMT}_{\text{de}\to\text{en}}$ and $\text{NMT}_{\text{en}\to\text{de}}$ for a future low-resource language X translating to and from English, and PLM_{X} can be used for both directions if the encoder and the decoder have the same number of layers. We hope such reuse can amortize the cost of preparing parent models. We release the code to facilitate future transfer at https://github.com/huawei-noah/noah-research/tree/master/noahnmt/dual-transfer. Besides, our experiment indicates that our approach can reduce the need of back-translation data size, while producing back-translation data and training on augmented data are both costly.

[6] https://www.mindspore.cn

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised Neural Machine Translation. In *International Conference on Learning Representations*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the Cross-lingual Transferability of Monolingual Representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged Back-Translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-Supervised

Learning for Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974, Berlin, Germany. Association for Computational Linguistics.

Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. 2020. Reusing a Pretrained Language Model on Languages with Limited Corpora for Unsupervised NMT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2703–2711, Online. Association for Computational Linguistics.

Stephane Clinchant, Kweon Woo Jung, and Vassilina Nikoulina. 2019. On the use of BERT for Neural Machine Translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 108–117, Hong Kong. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual Language Model Pretraining. In *Advances in Neural Information Processing Systems 32*, pages 7059–7069. Curran Associates, Inc.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A Comprehensive Survey of Multilingual Neural Machine Translation. *arXiv:2001.01115 [cs]*. ArXiv: 2001.01115.

Raj Dabre, Atsushi Fujita, and Chenhui Chu. 2019. Exploiting Multilingualism through Multistage Fine-Tuning for Low-Resource 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 1410–1416, Hong Kong, China. Association for Computational Linguistics.

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.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast Domain Adaptation for Neural Machine Translation. *arXiv:1612.06897 [cs]*. ArXiv: 1612.06897.

Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. 2018. Meta-Learning for Low-Resource Neural Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3622–3631, Brussels, Belgium. Association for Computational Linguistics.

Kenji Imamura and Eiichiro Sumita. 2019. Recycling a Pre-trained BERT Encoder for Neural Machine Translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 23–31, Hong Kong. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Jungo Kasai, Nikolaos Pappas, Hao Peng, James Cross, and Noah Smith. 2020. Deep Encoder, Shallow Decoder: Reevaluating Non-autoregressive Machine Translation. In *International Conference on Learning Representations*.

Yunsu Kim, Yingbo Gao, and Hermann Ney. 2019. Effective Cross-lingual Transfer of Neural Machine Translation Models without Shared Vocabularies. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1246–1257, Florence, Italy. Association for Computational Linguistics.

Tom Kocmi. 2020. Exploring Benefits of Transfer Learning in Neural Machine Translation. *arXiv:2001.01622 [cs]*. ArXiv: 2001.01622.

Tom Kocmi and Ondřej Bojar. 2018. Trivial Transfer Learning for Low-Resource Neural Machine Translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 244–252, Brussels, Belgium. Association for Computational Linguistics.

Tom Kocmi and Ondřej Bojar. 2020. Efficiently Reusing Old Models Across Languages via Transfer Learning. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 19–28, Lisboa, Portugal. European Association for Machine Translation.

Surafel M. Lakew, Aliia Erofeeva, Matteo Negri, Marcello Federico, and Marco Turchi. 2018. Transfer Learning in Multilingual Neural Machine Translation with Dynamic Vocabulary. In *IWSLT*.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018a. Word translation without parallel data. In *International Conference on Learning Representations*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-Based & Neural Unsupervised Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. In *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *The 12th International Workshop on Spoken Language Translation*.

Mieradilijiang Maimaiti, Yang Liu, Huanbo Luan, and Maosong Sun. 2019. Multi-Round Transfer Learning for Low-Resource NMT Using Multiple High-Resource Languages. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 18(4):38:1–38:26.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, volume 26, pages 3111–3119.

Graham Neubig and Junjie Hu. 2018. Rapid Adaptation of Neural Machine Translation to New Languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880, Brussels, Belgium. Association for Computational Linguistics.

Toan Q. Nguyen and David Chiang. 2017. Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

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.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging Pre-trained Checkpoints for Sequence Generation Tasks. *Transactions of the Association for Computational Linguistics*, 8(0):264–280.

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.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *International Conference on Machine Learning*, pages 5926–5936. PMLR.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

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.

Mengzhou Xia, Xiang Kong, Antonios Anastasopoulos, and Graham Neubig. 2019. Generalized Data Augmentation for Low-Resource Translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5786–5796, Florence, Italy. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting Source-side Monolingual Data in Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Austin, Texas. Association for Computational Linguistics.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer Learning for Low-Resource Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

A Runtime

We report the runtime of each step in our dual transfer (word) for $\text{NMT}_{\texttt{et}\rightarrow\texttt{en}}$ in Table 11. PLMs for other languages take similar time because we run a fixed number of steps. The runtime of the last finetuning step varies depending on low-resource parallel data size and learning rate. We also report the runtime of the "no transfer" baseline for this language pair as reference.

language code	train	dev	test
de-en	preprocessed - ParaCrawl	newstest2017	-
et-en	preprocessed	newsdev2018	newstest2018
tr-en	preprocessed	newsdev2016	newstest2018
fr-es	newstest2008-2011	newstest2012	newstest2013
de-en medical	EMEA - dev - test	random 3k of EMEA	random 3k of EMEA

Table 9: Parallel data source. "Preprocessed" means the preprocessed data provided in WMT 2018 news translation task.

language code	data source
en	News Crawl 2014-2017
de	News Crawl 2014-2017
et	News Crawl 2014-2017, BigEst Estonian corpus, Common Crawl
tr	News Crawl 2016-2017, Common Crawl
fr/es	News Crawl 2012
en medical	EMEA, PatTR, Wikipedia articles, AACT
de medical	EMEA, PatTR, Wikipedia articles

Table 10: Monolingual data source.

	# GPU	runtime (hours)
PLM_{de}	8	32
PLM_{et}	8	33
$\text{NMT}_{\text{de}\to\text{en}}$	8	39
$\text{NMT}_{\text{et}\to\text{en}}$	1	8
no transfer	1	20

Table 11: Runtime of each step in dual transfer (word) for $\text{NMT}_{\text{et}\to\text{en}}$. The runtime of the "no transfer" baseline for this language pair is also listed.

B Data Source and Preprocessing

We list the data source in Tables 9 and 10. Most of the data is from WMT 2018, unless otherwise noted. Medical data is from WMT 2014 medical translation task[7]. The French and Spanish monolingual data is from WMT 2013 news translation task[8].

All data sets are deduplicated. The Turkish monolingual data is further cleaned by removing lines with more than half non-Turkish characters, and we only use a subset with 100m lines.

C Hyperparameters and Development Performance

As we grid search learning rates in $\{1,3,5\}\times 10^{-4}$, we report the best found learning rate and the corresponding development BLEU in Tables 12, 13, and 14. The development BLEU is calculated by tokenized `multi-bleu.perl`. Due to the large scale of the 130m BT experiment, we directly use the best learning rates for 4m BT, and set other hyperparameters as in high-resource NMT.

[7] http://statmt.org/wmt14/medical-task/

[8] http://statmt.org/wmt13/translation-task.html

approach	et→en		tr→en		en→et		en→tr		fr→es	
	lr	BLEU	lr	BLEU	lr	BLEU	lr	BLEU	lr	BLEU
no transfer	5	22.37	5	17.56	3	15.32	5	14.03	3	11.62
(Zoph et al., 2016)	5	21.67	-	-	-	-	-	-	-	-
(Kim et al., 2019)	3	23.21	-	-	-	-	-	-	-	-
BERT2RND	3	22.84	-	-	-	-	-	-	-	-
BERT2BERT	3	23.98	1	22.06	1	16.44	1	16.27	3	21.57
(Kocmi and Bojar, 2018)	5	24.42	-	-	-	-	-	-	-	-
BBERT2BBERT	1	24.52	-	-	-	-	-	-	-	-
BBERT transfer	1	25.05	-	-	-	-	-	-	-	-
dual transfer (word)	1	25.33	1	23.34	3	18.31	1	17.84	1	26.06
dual transfer (word+position)	1	25.20	1	22.33	3	17.91	1	17.88	-	-

Table 12: Best found learning rate ($\times 10^{-4}$) and the corresponding development BLEU for various translation directions.

approach	4m ST		4m BT		130m BT	
	lr	BLEU	lr	BLEU	lr	BLEU
no transfer	3	22.48	3	19.45	3	19.97
dual transfer (word)	1	23.36	1	21.67	-	-
dual transfer (word+position)	1	23.67	1	21.73	1	21.27

Table 13: Best found learning rate ($\times 10^{-4}$) and the corresponding development BLEU for et→en ST and en→et BT experiments.

approach	lr	BLEU
no transfer (child)	3	63.39
BERT2BERT (child)	3	64.84
finetuning (parent)	1	65.26
dual transfer (parent)	3	65.13
dual transfer (child)	3	65.41

Table 14: Best found learning rate ($\times 10^{-4}$) and the corresponding development BLEU for domain adaptation on de→en.