A Novel Two-step Fine-tuning Framework for Transfer Learning in Low-Resource Neural Machine Translation

Yuan Gao , Feng Hou*, **Ruili Wang**
School of Mathematical and Computational Science, Massey University, New Zealand
{y.gao, f.hou, ruili.wang}@massey.ac.nz

Abstract

Existing transfer learning methods for neural machine translation typically use a well-trained translation model (i.e., *a parent model*) of a high-resource language pair to directly initialize a translation model (i.e., *a child model*) of a low-resource language pair, and the child model is then fine-tuned with corresponding datasets. In this paper, we propose a novel two-step fine-tuning (TSFT) framework for transfer learning in low-resource neural machine translation. In the first step, we adjust the parameters of the parent model to fit the child language by using the child source data. In the second step, we transfer the adjusted parameters to the child model and fine-tune it with a proposed distillation loss for efficient optimization. Our experimental results on five low-resource translations demonstrate that our framework yields significant improvements over various strong transfer learning baselines. Further analysis demonstrated the effectiveness of different components in our framework.

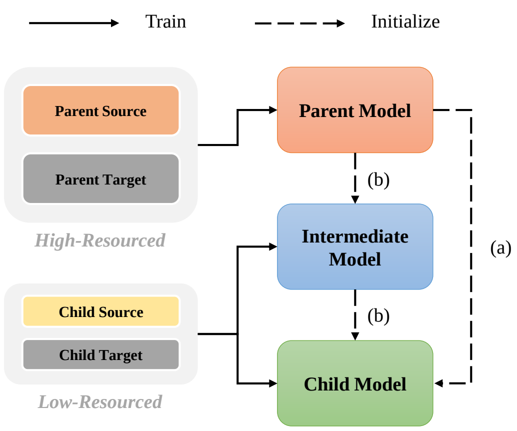


Figure 1: Comparison between vanilla transfer learning framework (a) and TSFT (b). Our proposed TSFT incorporates an intermediate model to pre-fine-tune the parent parameters to fit the child data.

1 Introduction

Neural machine translation (NMT) has achieved superior performance in terms of both fluency and adequacy for high-resource languages (Vaswani et al., 2017; Zhou and Keung, 2020; Cai et al., 2021; Guo et al., 2022). With the introduction of the attention mechanism (Yin et al., 2021; Petrick et al., 2022), NMT has been proven to be efficient and powerful in modeling long-distance dependencies. However, the performance of NMT systems deteriorates dramatically when insufficient parallel data are available for training (Sakaguchi et al., 2017; Michel and Neubig, 2018; Aharoni et al., 2019; Goyal et al., 2022). The scarcity of parallel corpora intensely limits the performance of an NMT system on low-resource languages.

Transfer learning is a learning paradigm for addressing the data scarcity problem (Zoph et al., 2016; Nguyen and Chiang, 2017; Li et al., 2022). For NMT, transfer learning aims to transfer the knowledge from a well-trained high-resource translation model (i.e., a parent model, e.g., English→German) to a low-resource translation model (i.e., a child model, e.g., English→the Māori language). Prior transfer learning methods in NMT (Zoph et al., 2016; Chu et al., 2017) primarily achieve knowledge transfer by initializing the parameters of the *child model* with the *parent model* and fine-tuning the child model on the corresponding data. Such direct transfer of knowledge raises a vocabulary mismatch problem (Lakew et al., 2018; Lin et al., 2019; Kocmi and Bojar, 2020), and results in unsatisfied results for low-resource translations. Some methods have been proposed to alleviate the vocabulary mismatch problem, such as constructing joint dictionaries or employing a cross-lingual token mapping technique (Passban et al., 2017; Kocmi and Bojar, 2018; Kim et al., 2019a). Additionally, Aji et al. (2020) proposed a token matching method that simply duplicates the embed-

*Corresponding author.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 3214–3224
June 16-21, 2024 ©2024 Association for Computational Linguistics

dings of overlapping tokens from the parent model to the child model.

Recently, based on the work of Aji et al. (2020), Li et al. (2022) proposed ConsistTL that uses the predictions of the parent model to continuously provide soft targets during the fine-tuning of the child model. However, given the differences between the source inputs of the parent and the child translation tasks, the parent model is not an optimal starting point for the single-step fine-tuning of the child model using limited parallel child data. Therefore, it is necessary to pre-fine-tune the parent model to fit the child language before initializing the child model with it.

Building upon this insight, we propose a simple yet effective transfer learning framework, named **T**wo-**S**tep **F**ine-**T**uning (TSFT), for low-resource NMT. As shown in Figure 1, we introduce an intermediate (child) model initialized with the parent model to adjust the parent parameters to fit the child language. TSFT involves two fine-tuning steps. In the first step, we feed child source sentences (i.e., monolingual data) and meaning-matched sentences in the parent source language into the intermediate and the parent models, respectively. Then, the intermediate model is fine-tuned with the objective of aligning probability distributions from the parent and intermediate models, aiming to adjust the parameters transferred from the parent model to perform well with child source sentences. Additionally, we propose a regularization-based strategy that can improve the translation performance of the intermediate model and benefit the child model. Note that we apply the token matching method to alleviate the vocabulary mismatch problem in the first step. In the second step, we transfer the adjusted parameters from the intermediate model to the child model and fine-tune the entire child model on the pertinent parallel data, employing both a cross-entropy loss and a proposed distillation loss. Extensive experiments on five low-resource translations show that TSFT surpasses the strongest baseline method with up to 1.2 SacreBLEU points. The ablation study demonstrates the effectiveness of different components within TSFT.

Our contributions can be summarized as follows:

- We propose a novel two-step fine-tuning framework for low-resource NMT, which introduces an intermediate (child) model to fit parent parameters for the data of child languages before initializing the child model with the parent model.
- We propose a regularization-based strategy for fine-tuning the intermediate model and a distillation loss for fine-tuning the child model.
- We validate our method by extensive experiments on various low-resource translations and achieve improved performance compared to various transfer learning methods.

2 Related work

Existing studies have demonstrated the success of transfer learning for low-resource NMT (Lin et al., 2019; Imankulova et al., 2019; Ji et al., 2020; Eronen et al., 2023). Zoph et al. (2016) first introduced transfer learning into the field of NMT and proposed a parent-child framework, where parameters from a pre-trained *parent model* are directly transferred to a new *child model* with a shared target language. Subsequent research largely builds upon the parent-child framework and tends to leverage highly related parent language to perform transfer learning (Passban et al., 2017; Setiawan et al., 2018). However, the languages closely related to low-resource languages are also low-resourced (Nguyen and Chiang, 2017; Xia et al., 2019) and offer only modest performance improvements. Thus, researchers focused on identifying the critical factors for the effectiveness of the parent language. Experimental results from (Lin et al., 2019; Aji et al., 2020) emphasized that linguistic or geographical distance does not appear as important as the size of the parent data (Lin et al., 2019; Aji et al., 2020). This insight expands the range of parent languages available for transfer learning, and alleviates the limitations of highly related parent languages. Consequently, later researchers shifted their attention to parent languages with low relatedness but high-resourced. However, this exacerbates the vocabulary mismatch problem, posing a new challenge to transfer learning.

One solution to the vocabulary mismatch problem is to build a joint dictionary before training a parent model (Kocmi and Bojar, 2018; Kim et al., 2019b). However, this restricts the applicability of a pre-trained parent model to a specific child model only. To overcome this limitation, Kim et al. (2019a) proposed pre-training a language-agnostic cross-lingual word embedding independently from the parent model. Concurrently, token matching methods also show their effectiveness in transfer learning without requiring additional training efforts (Aji et al., 2020; Kocmi and Bojar, 2020). Some other methods introduce highly related intermediate languages to gradually narrow the vocabulary disparity (Luo et al., 2019; Maimaiti et al., 2019). These methods take advantage of both large-scale data sources and syntactic similarity in the intermediate language.

Recently, Li et al. (2022) incorporated the idea of consistency learning into transfer learning based on the work of Aji et al. (2020) and proposed a novel transfer learning method called ConsistTL. This method enables the child model to utilize the parent model during fine-tuning. Subsequently, Liu et al. (2023) proposed kNN-TL, which extends ConsistTL by integrating a k-nearest neighbor (kNN) module, allowing the child model to utilize the parent model during inference. While our method also builds on ConsistTL, we focus on enhancing the child model's performance during fine-tuning. Thus, our work is orthogonal to kNN-TL.

3 Method

In this section, we begin by providing an overview of the basic concepts behind transfer learning and then present our transfer learning framework, TSFT, in detail.

3.1 Transfer Learning Primary

Given a source sentence $x = \{x_1, \ldots, x_I\}$, the objective of an NMT model is to translate it to a new sentence $y = \{y_1, \ldots, y_J\}$ in a target language, where the source sentence and target sentence have lengths I and J, respectively. A typical NMT model is composed of an encoder and a decoder. The encoder is designed to extract high-level semantic information from the source sentences and represent them as hidden states H_e. The decoder generates the output probability $P(y_i|H_e, y_{<i})$ of the next target token y_i. An NMT model is trained on a parallel corpus by minimizing the cross-entropy (CE) loss between the predicted sentence and the ground-truth translation as follows:

$$L_{ce} = -\sum_{i=1}^{J} \log P(y_i|y_{<i}, x, \theta), \qquad (1)$$

where θ is the parameters of the entire NMT model.

Transfer learning has been widely used when only limited training datasets are available for the problem at hand. It transfers the knowledge acquired from large-scale data to enhance the model performance under low-resource conditions. Transfer learning typically follows a parent-child framework (Zoph et al., 2016), where it involves reusing the parameters θ_p from a pre-trained parent model to initialize part or all parameters of a child model. In the field of NMT, the parent model $\mathcal{M}_p$ is initially trained on a high-resourced parallel dataset $D_p = \{X_p, Y_p\}$, while there is only a limited-sized dataset $D_c = \{X_c, Y_c\}$ available to the child model $\mathcal{M}_c$. After the initialization step, the child model can be fine-tuned on D_c, which is also optimized through the minimization of the CE loss.

3.2 Two-step Fine-tuning

For NMT, an ideal transfer learning framework should enable the parent model to exert its complete capabilities on the child task. However, owing to the disparities between the parent and child languages, the current one-step fine-tuning transfer learning framework struggles to adjust the parameters of the parent model to fit the child source language under the constraints of limited child data.

The idea of TSFT is simple: before initializing the child model with the parent model, we first adjust the parameters of the parent model to enhance its congruity with the child source language. In this work, we propose to introduce an intermediate model, denoted as $\mathcal{M}_a$, to make the parameters of the parent model fit for the child data. Specifically, we initialize the intermediate model with the parent model and pre-fine-tune it by using the source side sentences of the child data, then fine-tune the child model with both the source and target child training data. Therefore, we design TSFT as a two-step framework, as shown in Figure 2.

Step 1: Intermediate Fine-tuning After initializing the intermediate model with a well-trained parent model, we aim to equip the intermediate model with the ability to utilize child source sentences as input for target language generation. Since the intermediate model and the parent model share the same target language, it is crucial to retain the generation ability of the parent model. Therefore, we input the source-side sentences of the child data to the intermediate model and the parent model and utilize the predicted distribution of the parent model as the soft label for fine-tuning.

However, it is infeasible to directly input child

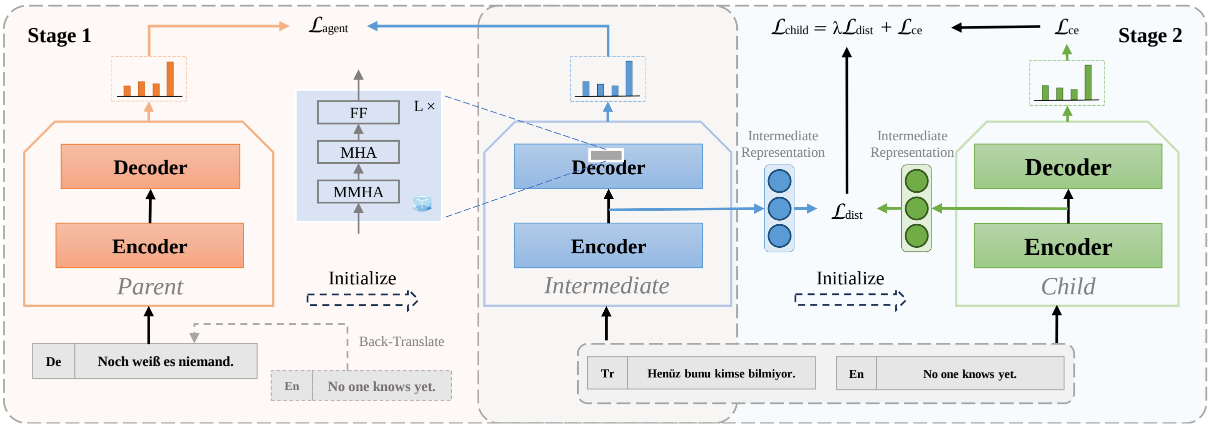


Figure 2: Our proposed transfer learning framework TSFT for low-resource NMT. In Step 1, the loss function L_{inter} is used to optimize the intermediate model. In Step 2, the child model is optimized by L_{child}. The blue icy blocks are initialized with the *parent* model and frozen. The input German sentences are produced through back-translation.

source sentences into the parent model, given that the parent and child models have different source languages. Thus, we need a meaning-matched sentence for each child source sentence in the parent source language. In the context of low-resource translations, parallel data for non-English-centric is often limited in size or entirely absent, making it difficult to meet the requirements for intermediate fine-tuning. Therefore, we adopt the method of Li et al. (2022) to generate pseudo parent data $D_{p*} = \{X_{p*}, Y_c\}$ by using a reversed parent model, where each $x_{p*} \in X_{p*}$ is aligned with $y_c \in Y_c$. Although such a method requires training a reverse parent model, it effectively generates meaning-matched input sentences for the parent model. In addition, we use the following loss function to optimize the intermediate model:

$$L_{inter} = \sum_{i=1}^{J} F_d[P_{inter}(y_i), P_{parent}(y_i)], \quad (2)$$

where F_d is a distribution measurement method, in this work, we choose Jensen-Shannon (JS) divergence (Lin, 1991; Wen et al., 2023) as our F_d. Our preliminary experiments find that JS divergence outperforms using Kullback–Leibler (KL) divergence when taking $P_{inter}(y_i)$ as the first item and $P_{parent}(y_i)$ as the second one. $P_*(y_i)$ represents the prediction distributions of translation models at time step i, which is conditioned on the input sentence and the previous tokens:

$$P_*(y_i) = P_*(y_i|x, y_{<i}). \quad (3)$$

Before fine-tuning the intermediate model, we first apply the token matching method (Aji et al., 2020) that duplicates the embeddings of overlapping tokens from the parent and child vocabularies to alleviate the vocabulary mismatch problem.

Step 2: Child Fine-tuning In the second step, we employ the target-side sentences from the child training data as labels to fine-tune the child model with CE loss, following the general process of transfer learning. Since the encoder of the intermediate model has fine-tuned with the child source sentences, we argue that it encompasses valuable information that can facilitate the child model. Therefore, we extract the encoder outputs, $P_*^e(\cdot)$, from both the intermediate and child models and incorporate a distillation loss L_{dist} as an extra objective to optimize the child model by minimizing the KL divergence between two output representations:

$$L_{dist} = -\sum_{i=1}^{I} P_{inter}^e(x_i) \cdot logP_{child}^e(x_i), \quad (4)$$

$$\begin{aligned} P_*^e(x_i) &= P_*^e(x_i|x, \tau) \\ &= \frac{exp(z_i/\tau)}{\sum_{j \in V} exp(z_j/\tau)}, \end{aligned} \quad (5)$$

where I denotes the sentence length of a child source sentence, z denotes the logits output of encoders before log_softmax is computed, V represents the vocabulary, and τ is a temperate factor used to smooth the prediction distributions. As we only reuse the output of encoders, the process of encoder distillation does not add any extra parameters to models. The overall loss is obtained by a weighted sum of L_{ce} and L_{dist}:

$$L_{child} = L_{ce} + \lambda L_{dist}, \tag{6}$$

where λ is a balancing hyper-parameter.

Partial Decoder Freeze Regularization-based methods are widely used to alleviate the *catastrophic forgetting* issue (Kirkpatrick et al., 2017; Gu and Feng, 2020; Gu et al., 2021). While updating all parameters typically yields good results on a new domain, the data distribution difference between the old and new domains can engender the issue of *catastrophic forgetting*, causing the fine-tuned model to abandon linguistic knowledge learned from previous dataset (Thompson et al., 2019; Bérard, 2021). In this work, we are interested in introducing the regularization-based technique during Step 1 to preserve the predictive capabilities of the parent model. We propose a Partial Decoder Freeze (PDF) strategy to freeze the parameters of the last l decoder layers of the intermediate model and only update the rest parameters. For the selection of parameters l, we conducted empirical experiments in Section 5.1.

4 Experiments

4.1 Settings

Datasets We conduct experiments on five low-resource translation tasks, four of which are from the Global Voices datasets (Tiedemann, 2012; Khayrallah et al., 2020): Polish (Pl), Hungarian (Hu), Indonesian (Id), Catalan (Ca) to English (En), where we use the officially provided training sets, validation sets and test sets in our experiments. The other one is the WMT 2017 Turkish (Tr) to En benchmark. We use *newstest2016* as the validation set and *newstest2017* as the test set. For the parent models training, we use the German-English dataset following the empirical advice of (Aji et al., 2020; Li et al., 2022). We take the WMT 2017 news translation task as our parent dataset containing around 5.8M paired sentences. The detailed statistics of these parallel corpora are presented in Table 1. For fair comparisons, we adopt the same data preprocess techniques as previous research of TL (Li et al., 2022), which only apply normalization and tokenization to parallel sentences by using *Moses* toolkit[1]. Further, we apply Byte Pair Encoding (BPE) (Sennrich et al., 2016) to address the out-of-vocabulary problem and segment words with 16,000 merge operations for Turkish and 8,000 for the rest.

Datasets	# Train	# Valid	# Test
Global Voices Pl - En	39.9K	2,000	2,000
Global Voices Ca -En	15.2K	2,000	2,000
Global Voices Id - En	8.4K	2,000	2,000
Global Voices Hu - En	7.7K	2,000	2,000
WMT 2017 Tr - En	196.6K	3,000	3,007
WMT 2017 De - En	5.8M	3,000	3,003

Table 1: The statistics of parallel corpora.

Model Configuration In our experiments, we implement translation models with *fairseq* [2] toolkit. We choose the Transformer (Vaswani et al., 2017) as the backbone to implement our framework. We use Transformer_base that consists of 6 encoder and decoder layers with 8 attention heads. The number of dimensions of all sub-layers in the model is set to 512, and the inner layers of feed-forward layers have 2048 dimensions. Our models are trained on 2 Nvidia A100 GPUs. We train our models using Adam (Kingma and Ba, 2015) with $(\beta_1, \beta_2) = (0.9, 0.98)$ and use cross-entropy as criterion with *label smoothing* $= 0.1$. In addition, we train the forward and backward parent model (i.e., De→En and En→De) with the initial learning rate $1e^{-7}$ and gradually increase till $1e^{-3}$ within 10,000 warm-up updates. For the models with transfer learning, we set the initial learning rate to $1e^{-7}$, and the peak learning rate is $2e^{-4}$ within 1,000 warm-up steps. Dropout is applied to the output of each sub-layer with a rate of 0.3 to avoid over-fitting. Besides, attention and activation dropouts are also used with a rate of 0.1 and 0.1. We train all models with a maximum of 200 epochs and select the checkpoints with the best BLEU score on the validation set as our final model, where beam search is applied with beam size 5, and the length penalty is 1.

Baselines We use the following baselines to validate our method:

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

[2]https://github.com/facebookresearch/fairseq

Model	Tr→En		Hu→En		Id→En		Ca→En		Pl→En	
	BLEU	**BS**	**BLEU**	**BS**	**BLEU**	**BS**	**BLEU**	**BS**	**BLEU**	**BS**
Vanilla	17.8	51.8	0.9	0.9	1.1	13.2	1.1	15.5	1.5	18.9
TL	17.6	51.9	5.9	27.4	13.5	37.7	21.6	51.8	19.9	55.3
TM-TL	18.6	53.9	10.6	41.2	18.6	49.9	25.3	58.9	21.4	58.2
ConsistTL	19.3	55.9	11.9	43.9	19.7	52.2	26.6	60.0	22.4	59.9
TSFT (ours)	**20.0**	**56.7**	**13.1**	**44.6**	**20.5**	**53.3**	**27.7**	**60.7**	**23.3**	**60.5**

Table 2: The SacreBLEU and BERTScore scores of baselines and ours on various translations. "BS" represents BERTScore. **Blod** indicates the best result. BLEU score reflects that TSFT is significantly better than ConsistTL with t-test $p < 0.05$. The number of bootstrap resamples is set to 1,000 to measure the significant difference between results.

- **Vanilla NMT** (Vaswani et al., 2017): A bilingual NMT model with Transformer architecture directly trained on low-resource child training data from scratch.

- **TL** (Zoph et al., 2016): The first transfer learning work for NMT, initializing the child model with a parent model except for the source word embeddings. Note that the original work employed a two-layer encoder-decoder LSTM model, whereas we replicate TL using Transformer.

- **TM-TL** (Aji et al., 2020): To transfer embeddings across languages with distinct linguistic characteristics, Token Matching (TM) is proposed to assign the child word embeddings with the same tokens in the parent embeddings. The remaining unmatched tokens are assigned random embeddings as TL.

- **ConsistTL** (Li et al., 2022): Based on TM-TL, ConsistTL is proposed to enhance the child model by incorporating the prediction of the parent model during the fine-tuning of the child model.

Metrics To validate the effectiveness of our proposed framework, we use the following two metrics:

- **BLEU** (Papineni et al., 2002): Considering the discrepancy among different tokenization processes, we apply the SacreBLEU score (Post, 2018)[3] for all experiments.

[3]Signature: nrefs:1 + case:mixed + eff:no + tok:13a + smooth:exp + version:2.0.0

Hyper-parameter	Tr→En	Hu→En
(λ = 2.0, τ = 2.0)	19.9	13.0
(λ = 3.0, τ = 2.0)	19.8	12.8
(λ = 4.0, τ = 2.0)	**20.0**	**13.1**
(λ = 5.0, τ = 2.0)	19.9	12.9
(λ = 4.0, τ = 0.5)	19.7	12.9
(λ = 4.0, τ = 1.0)	19.7	**13.1**
(λ = 4.0, τ = 3.0)	19.4	13.0

Table 3: The SacreBLEU scores on the test set of the Tr → En and Hu→En translations with different λ and τ.

- **BERTScore** (Zhang et al., 2020): Leveraging a pre-trained BERT model to evaluate the semantic correctness between the predictions and references by cosine similarity.

4.2 Main Results

The results on five low-resource translation benchmarks are presented in Table 2. In our experiments, we utilize German as the parent language, and the parent models are pre-trained on a German-to-English dataset. As we can see, our method significantly outperforms the vanilla NMT in terms of both SacreBLEU and BERTScore. Compared with TL and TM-TL, TSFT still achieves significant improvements on all translations. Moreover, our proposed TSFT also has demonstrated superior performance compared to the strongest baseline ConsistTL with up to +1.2 SacreBLEU points and +1.1 BERTScore points. Overall, these results prove that our proposed transfer learning framework TSFT can effectively improve the performance of the child model on low-resource translation tasks.

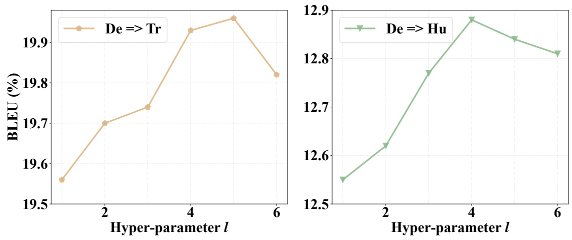


Figure 3: The SacreBLEU scores of TSFT with different hyper-parameter l on Tr $\rightarrow$ En and Hu $\rightarrow$ En. De $\Rightarrow$ Tr / Hu indicates De is the parent language and Tr / Hu is the child language.

Models	Tr→En	Hu→En
TSFT	20.0	13.1
w/o PDF	19.5	12.5
w/o L_{dist}	19.8	12.8
w/o Step 2	18.9	11.2
w/o Step 2 + PDF	18.6	10.6

Table 4: The SacreBLEU scores on the test set of the Tr $\rightarrow$ En and Hu→En translations with PDF, L_{dist}, and Step 2 ablation.

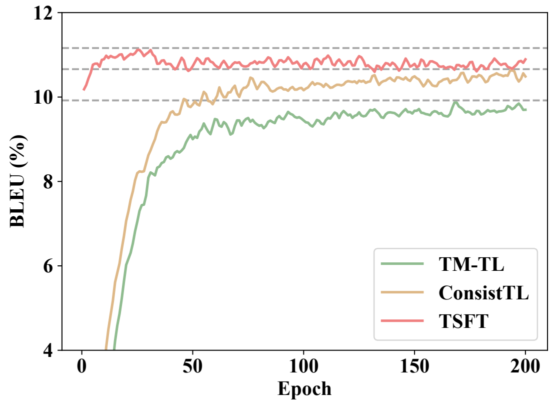


Figure 4: Learning curves of different TL methods.

5 Analysis

5.1 Effect of the Number of Freezing Layers

In Section 3.2, we utilize the PDF strategy in Step 1. However, we do not clearly know the optimal number of freezing layers l that can benefit the child model most. Different numbers of freezing layers would significantly impact the child model performance. Hence, in this section, we conduct a comparative analysis of the impact of different l on the translation performance of the child model.

Concretely, we still use the De→En model as the parent model and select Tr→En and Hu→En translations as child tasks. We tune the hyper-parameter l by performing a grid search on $l \in \{1, 2, 3, 4, 5, 6\}$. Figure 3 illustrates the model performance with different values of l. We can find that the final child models achieve the best performance in Tr→En and Hu→En when l is 5 and 4, respectively. Consequently, we set l as 5 for Tr→En translation and 4 for the rest.

Despite a substantial size difference between the Tr→En and Hu→En datasets, there is not much difference in the choice of the number of layers to freeze. For this phenomenon, we speculate that the distinction between these two child datasets is negligible compared to the size distinctions with the parent dataset, as shown in Table 1. Therefore, when applying our framework to parent models with relatively limited resources, the choice of the number of frozen decoder layers needs to be carefully considered to achieve optimal results.

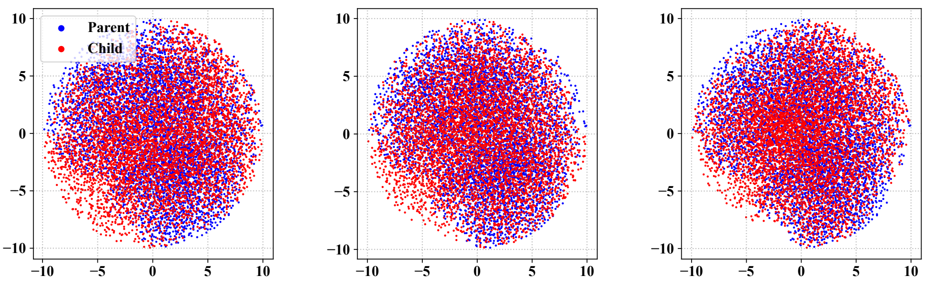
5.2 Effect of Hyper-parameters λ and τ

Hyper-parameter λ is crucial to controlling the influence of the two losses within the L_{child}. In this part, we set λ to $\{2.0, 3.0, 4.0, 5.0\}$ to investigate the impact of different values of λ on the performance of the child model. The corresponding SacreBLEU scores are presented in Table 3. For both Tr→En and Hu→En translations, the best performances are obtained when λ is set to 4.0. Hence, we set λ as 4.0 for all experiments involving L_{dist}.

In addition, we also conduct experiments with varying values of τ during the training process of the child model, while keeping λ fixed at 4.0. As illustrated in Table 3, we can find that the performance of the child model is sensitive to τ and the performance is best when τ is set to 2.0. We argue that this is because minimizing the KL divergence is difficult, but using a larger τ (e.g., 3.0) may diminish the information from the intermediate model, which is not helpful in improving the performance of the child model.

5.3 Ablation Study

We conduct an ablation study of the PDF strategy, L_{dist}, and Step 2 to explore their effects on our framework. We present the performance of four variants of TSFT as follows: 1) w/o PDF. During the training process of Step 1, we do not freeze any layers of the intermediate model, fine-tuning all parameters in every epoch. 2) w/o L_{dist}. In Step 2, we eliminate the distillation loss between the encoders of the intermediate and child models, conducting fine-tuning of the child model using



(a) TM-TL (b) ConsistTL (c) TSFT (Step 1)

Figure 5: Sentence representations after using T-SNE dimensionality reduction. The blue points denote the output from the parent model, and the red points denote the output from the fine-tuned models obtained from different transfer learning methods.

L_{ce} exclusively. 3)w/o Step 2. We evaluate the translation performance of the intermediate model. 4) w/o Step 2 + PDF. Based on 3), we do not freeze any layers of the intermediate model during Step 1. We conduct experiments on Tr→En and Hu→En translations, which correspondingly represent the largest and smallest datasets among those applied in our main experiments. The results are shown in Table 4. It is evident that excluding the PDF strategy, L_{dist}, or Step 2 resulting in a deterioration of the translation quality, underscoring the efficacy of these components within TSFT. The experimental results show that PDF has a greater impact than L_{dist}. Further, we observe that PDF can effectively improve the translation performance of the intermediate model and benefit the child model. This observation shows that retaining the performance of the parent model is crucial for improving the performance of the child model.

5.4 Comparison of Learning Curves

A learning curve represents a model's learning performance throughout the duration of training and is a widely employed diagnostic tool in machine learning (Kambhatla et al., 2022; Bao et al., 2023). In this section, we present the validation learning curve to assess the generalization capabilities of TM-TL, ConsistTL, and TSFT by using the SacreBLEU score as the criterion. Figure 4 illustrates the learning curves of child models trained with three transfer learning methods. Compared with TM-TL and ConsistTL, TSFT exhibits superior initial performance and convergence speed. Note that the TSFT curve delineates the performance of the model fine-tuned after Step 1. This observation emphasizes the effectiveness of fine-tuning the intermediate model in enhancing the final model's performance, which can be attributed to the augmentation of adaptability to child data consequent to the fine-tuning process in Step 1. Besides, as the training progresses into the stable phase, we can find that the performance of the child model under the TSFT framework is consistently higher than that of TM-TL and ConsistTL. It is noteworthy that, similar to TM-TL and ConsistTL, TSFT does not utilize additional data or resources. Thus, the performance improvement of the child model can be attributed to the effectiveness of the pre-fine-tune process.

5.5 Sentence Representation Visualization

In our framework, the intermediate model is used to adjust the parent parameters to perform well when using child source sentences as input (Section 3.2). Thus, in this section, we visualize the target-side sentence representations of the De-En parent model and Hu-En models obtained from different transfer learning methods. We utilize the T-SNE method (Hinton and Roweis, 2002) to project the representations into a 2-dimensional space, as shown in Figure 5. This figure shows that TM-TL struggles to align the child representations with the parent representations. ConsistTL slightly reduces the discrepancy between the parent and child representations, whereas the intermediate model from TSFT makes the representations much more similar. This observation shows that our fine-tuned intermediate model can produce similar outputs to the parent model even with different source languages.

6 Conclusion

In this paper, we propose TSFT: a novel two-step fine-tuning framework for low-resource NMT.

TSFT incorporates an intermediate (child) model to pre-fine-tune the parent model to fit the child data. The intermediate model is initialized with the parent model and then fine-tuned on the child source data in the first step. We propose freezing partial decoder layers when fine-tuning the intermediate model to alleviate catastrophic forgetting. In the second step, TSFT initializes the child model with the intermediate model and fine-tunes the child model on the parallel data using the cross-entropy and proposed distillation losses. Experimental results on five low-resource translations demonstrate the effectiveness of our proposed TSFT.

Limitations

When using our proposed framework, two fine-tuning steps are necessary to obtain the final child model. Therefore, compared to one-step transfer learning methods in NMT, TSFT may require more training time and computation resources to transfer parent knowledge to the child model. Nevertheless, it is important to note that TSFT does not introduce additional time or computing resource consumption during inference. Besides, TSFT is designed for transfer learning scenarios when the target languages of the parent and child models are identical. We will try transferring different target languages in the future.

Ethics Statement

This study uses only publicly accessible datasets and models that permit academic research. The preprocessing tools and model training toolkit are open-sourced without copyright conflicts.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments. This work is supported by the 2020 Catalyst: Strategic New Zealand - Singapore Data Science Research Programme Fund by Ministry of Business, Innovation and Employment (MBIE), New Zealand.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*.

Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, and Rico Sennrich. 2020. In neural machine translation, what does transfer learning transfer? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Guangsheng Bao, Zhiyang Teng, and Yue Zhang. 2023. Target-side augmentation for document-level machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Alexandre Bérard. 2021. Continual learning in multilingual NMT via language-specific embeddings. In *Proceedings of the Sixth Conference on Machine Translation*.

Deng Cai, Yan Wang, Huayang Li, Wai Lam, and Lemao Liu. 2021. Neural machine translation with monolingual translation memory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Juuso Eronen, Michal Ptaszynski, Karol Nowakowski, Zheng Lin Chia, and Fumito Masui. 2023. Improving polish to english neural machine translation with transfer learning: Effects of data volume and language similarity. *arXiv preprint arXiv:2306.00660*.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*.

Shuhao Gu and Yang Feng. 2020. Investigating catastrophic forgetting during continual training for neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Shuhao Gu, Yang Feng, and Wanying Xie. 2021. Pruning-then-expanding model for domain adaptation of neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*.

Dengji Guo, Zhengrui Ma, Min Zhang, and Yang Feng. 2022. Prediction difference regularization against perturbation for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Geoffrey E Hinton and Sam Roweis. 2002. Stochastic neighbor embedding. *Advances in neural information processing systems*.

Aizhan Imankulova, Raj Dabre, Atsushi Fujita, and Kenji Imamura. 2019. Exploiting out-of-domain parallel data through multilingual transfer learning for low-resource neural machine translation. In *Proceedings of Machine Translation Summit XVII*.

Baijun Ji, Zhirui Zhang, Xiangyu Duan, Min Zhang, Boxing Chen, and Weihua Luo. 2020. Cross-lingual pre-training based transfer for zero-shot neural machine translation. In *Proceedings of the AAAI conference on artificial intelligence*.

Nishant Kambhatla, Logan Born, and Anoop Sarkar. 2022. Cipherdaug: Ciphertext based data augmentation for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Huda Khayrallah, Brian Thompson, Matt Post, and Philipp Koehn. 2020. Simulated multiple reference training improves low-resource machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Yunsu Kim, Yingbo Gao, and Hermann Ney. 2019a. 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*.

Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019b. Pivot-based transfer learning for neural machine translation between non-english languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*.

Tom Kocmi and Ondrej Bojar. 2018. Trivial transfer learning for low-resource neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*.

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*.

Surafel M Lakew, Aliia Erofeeva, Matteo Negri, Marcello Federico, and Marco Turchi. 2018. Transfer learning in multilingual neural machine translation with dynamic vocabulary. In *Proceedings of the 15th International Conference on Spoken Language Translation*.

Zhaocong Li, Xuebo Liu, Derek F. Wong, Lidia S. Chao, and Min Zhang. 2022. ConsistTL: Modeling consistency in transfer learning for low-resource neural machine translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

J. Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, et al. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Shudong Liu, Xuebo Liu, Derek F Wong, Zhaocong Li, Wenxiang Jiao, Lidia S Chao, and Min Zhang. 2023. kNN-TL: k-nearest-neighbor transfer learning for low-resource neural machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Gongxu Luo, Yating Yang, Yang Yuan, Zhanheng Chen, and Aizimaiti Ainiwaer. 2019. Hierarchical transfer learning architecture for low-resource neural machine translation. *IEEE Access*.

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*.

Paul Michel and Graham Neubig. 2018. Mtnt: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

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*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*.

Peyman Passban, Qun Lui, and Andy Way. 2017. Translating low-resource languages by vocabulary adaptation from close counterparts. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Frithjof Petrick, Jan Rosendahl, Christian Herold, and Hermann Ney. 2022. Locality-sensitive hashing for long context neural machine translation. In *Proceedings of the 19th International Conference on Spoken Language Translation*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation*.

Keisuke Sakaguchi, Kevin Duh, Matt Post, and Benjamin Van Durme. 2017. Robsut wrod reocginiton via semi-character recurrent neural network. In *Thirty-first AAAI conference on artificial intelligence*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Hendra Setiawan, Zhongqiang Huang, and Rabih Zbib. 2018. BBN's low-resource machine translation for the LoReHLT 2016 evaluation. *Machine Translation*.

Brian Thompson, Jeremy Gwinnup, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 2019. Overcoming catastrophic forgetting during domain adaptation of neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. 2023. f-Divergence minimization for sequence-level knowledge distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

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*.

Pengcheng Yin, Hao Fang, Graham Neubig, Adam Pauls, Emmanouil Antonios Platanios, Yu Su, Sam Thomson, and Jacob Andreas. 2021. Compositional generalization for neural semantic parsing via span-level supervised attention. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*.

Tianyi Zhang, Varsha Kishore, Felix Wu0, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Jiawei Zhou and Phillip Keung. 2020. Improving non-autoregressive neural machine translation with monolingual data. In *Proceedings of the 58th Annual Meeting of the 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*.