Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 6140153, 11 pages
https://doi.org/10.1155/2020/6140153

Research Article

A Joint Back-Translation and Transfer Learning Method for Low-Resource Neural Machine Translation

Gong-Xu Luo,[1,2,3] **Ya-Ting Yang**,[1,2,3] **Rui Dong**,[1,2,3] **Yan-Hong Chen**,[1,2,3] **and Wen-Bo Zhang**[1,2,3]

[1]*The Xinjiang Technical Institute of Physical & Chemistry, Chinese Academy of Sciences, Urumqi 830011, China*
[2]*University of Chinese Academy of Sciences, Beijing 100049, China*
[3]*Xinjiang Laboratory of Minority Speech and Language Information Processing, Urumqi 830011, China*

Correspondence should be addressed to Ya-Ting Yang; yangyt@ms.xjb.ac.cn

Received 16 October 2019; Revised 15 April 2020; Accepted 2 May 2020; Published 31 May 2020

Academic Editor: Vassilios Constantoudis

Neural machine translation (NMT) for low-resource languages has drawn great attention in recent years. In this paper, we propose a joint back-translation and transfer learning method for low-resource languages. It is widely recognized that data augmentation methods and transfer learning methods are both straight forward and effective ways for low-resource problems. However, existing methods, which utilize one of these methods alone, limit the capacity of NMT models for low-resource problems. In order to make full use of the advantages of existing methods and further improve the translation performance of low-resource languages, we propose a new method to perfectly integrate the back-translation method with mainstream transfer learning architectures, which can not only initialize the NMT model by transferring parameters of the pretrained models, but also generate synthetic parallel data by translating large-scale monolingual data of the target side to boost the fluency of translations. We conduct experiments to explore the effectiveness of the joint method by incorporating back-translation into the parent-child and the hierarchical transfer learning architecture. In addition, different preprocessing and training methods are explored to get better performance. Experimental results on Uygur-Chinese and Turkish-English translation demonstrate the superiority of the proposed method over the baselines that use single methods.

1. Introduction

Neural machine translation (NMT) achieves the process of translation between different languages with a neural network [1]. Currently, the NMT systems, which follow the encoder-decoder architecture proposed in [2, 3], have obtained state-of-the-art translation quality for several language pairs [4–8]. The encoder network maps the source sentence to context vectors; the decoder is actually a language model to generate target words with the help of attention mechanism, which gets word-alignment information by calculating weights with context vectors. However, these data-driven NMT systems are not good enough for low-resource languages, which draws widespread concern from both research and industry communications.

Researchers have explored different ways to address the problem of data sparseness for low-resource languages. A straight forward way is to generate more parallel data. In the literature, existing studies for data augmentation are dominated by several directions. On the decoder side, the decoder network, which works like a language model to generate target languages, plays a crucial role in boosting the fluency of the translation. A pioneer work in this field was given by Gulcehre et al. [9]. They integrated monolingual data for pure NMT, which trains an RNN language model separately for the NMT model. In contrast to Gulcehre et al.'s work, Sennrich et al. explored a new strategy to generate parallel data by translating large-scale monolingual data of the target side [10]. Similarly, the source-side monolingual data was fully explored. In an encoder network, Zhang and Zong employed a self-learning algorithm to generate

synthetic parallel data by using large-scale source-side monolingual data, which obtains high-quality context vector representation [11]. In order to make full use of source-side and target-side monolingual data, Cheng et al. [12] used semisupervised learning for training NMT models on parallel data and monolingual data. They reconstructed the monolingual corpus with an autoencoder. In the source autoencoder, the encoder was trained on source-target language pairs and the decoder was trained on target-source language pairs. Luong et al. adopted an autoencoder to exploit large monolingual data of source-side and target-side languages, which shares encoders and decoders for NMT [13]. Besides, in [14], Fadaee et al. utilized the language model to select the contexts that are suitable for substituting common words with rare words to generate new sentence pairs. The target translation was also substituted by automatic word alignments and the language model. In [15], Currey et al. generated a new parallel corpus by copying the target monolingual data.

Another effective way for low-resource problems is transfer learning methods [16]. The common method for NMT is the model-based transfer learning. Inspired by the sharing encoder, decoder, and attention mechanism in multitask learning for NMT [17, 18], Zoph et al. pioneered to adopt the transfer learning method for low-resource NMT. They proposed the parent-child transfer learning architecture and outperformed the strong phrase-based statistic machine translation on Hausa-English [19]. Following Zoph et al.'s work, Nguyen et al. applied the transfer learning architecture cross low-resource-related languages and used byte pair encoding (BPE) [20] for subword segmentation, which significantly improved the performance of the child model [21]. In addition, Dabre et al. studied language relatedness for the parent-child transfer learning method [22]. They concluded that the parent model with similar languages is the best for the child model. In contrast to Dabre et al.'s work, Kocmi and Bojar found that the parent model with higher resource languages is more effective for the improvement of low-resource languages compared with the similar low-resource languages [23]. Inspired by the previous work, we proposed the hierarchical transfer learning architecture for low-resource languages, which effectively combine the data volume advantage of high-resource languages with the language similarity advantage of intermediate languages [24]. Experimental results show that the architecture outperformed other methods that are based on the parent-child architecture.

Despite the success of transfer learning methods and data augmentation methods for the data scarcity problem in machine translation tasks, we argue that they are not effectively enough if separately adopted one of these methods, which limits the capacity of NMT systems. Data augmentation methods strengthen the encoder and decoder by using large monolingual data and generating synthetic parallel data; however, transfer learning methods focused on parameter sharing and better initialization. They improved the performance of low-resource NMT in a diverse angle.

Since the back-translation method need not to change the structure of NMT models, therefore, in this paper, we propose a new joint method, which incorporates the back-translation method into mainstream frameworks of transfer learning for NMT, aiming at further improving the performance for low-resource languages. In order to evaluate the effectiveness of the joint method, we conduct experiments on diverse transfer learning architectures. Specifically, in the parent-child architecture proposed by Zoph et al. [19], we incorporate back-translation [10] into the child model to boost fluency of the target language. Similarly, we also adopt back-translation to generate large synthetic parallel data on the third layer of the hierarchical transfer learning architecture [24]. A demo example for the joint method with a hierarchical transfer learning architecture on Uygur-Chinese is shown in Figure 1. We first train a Chinese-Uygur translation model on parallel data as shown in Table 1. Next, we get mounts of synthesized parallel data by translating large monolingual data and mix it with real data into new mixed data as shown in the right side of Figure 1. Then the NMT model is trained on high-resource language pairs and intermediate language pairs in turn to obtain prior knowledge. Finally, the model is trained on the new mixed low-resource language pairs to converge. The joint method retains the effectiveness of the transfer learning method and exploits large monolingual data for low-resource NMT. Specially, we explore the generalization of the joint method for low-resource NMT in the translation tasks of Uygur-Chinese and Turkish-English. Finally, the quality of translation is evaluated by the general BLEU value [25]. Experimental results show that the joint method, which combines transfer learning methods with back-translation, significantly improves the performance for low-resource languages compared to using only one of the methods. In summary, our contributions are as follows:

(i) In order to effectively improve the capacity of NMT for low-resource problems, we propose a new method that incorporates data augmentation methods into diverse transfer learning architectures, which is more effective compared with each single methods.

(ii) Furthermore, different preprocessings and training methods are explored for better performance.

(iii) The generalization of the method is verified by experimenting it on Uygur-Chinese and Turkish-English. Empirical experimental results show that the method achieved significant results in both low-resource languages.

The remainder of this paper is organized as follows. The NMT model of our method is introduced in Section 2. We then present the methodology for low-resource languages in Section 3. Section 4 reports the experiments. Results and analysis are reported in Section 5. Section 6 introduces the related work. Finally, we conclude this paper in Section 7.

2. Neural Machine Translation

Our approach on incorporating data augmentation methods into transfer learning architectures can be applied to any

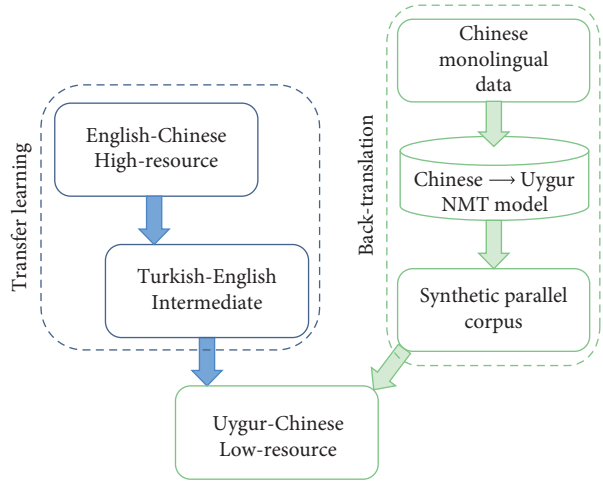

Figure 1: A demo example of the joint method with a hierarchical transfer learning architecture on Uygur-Chinese.

Table 1: The statistics of datasets.

Corpus	Dataset	Sentences (M)	Test set
English-Chinese	Union parallel data	15	Newtest2017
Turkish-English	WMT16 parallel data	0.2	Newtest2016
Uygur-Chinese	CWMT2017 parallel data	0.35	CWMT-2017
English	Monolingual data	0.75	N/A
Chinese	Monolingual data (organized)	0.9	N/A

NMT models as long as it can be used for transfer learning architectures. Without loss of generality, we follow the self-attention-based NMT proposed by Vaswani et al. which substitutes a recurrent neural network by fully utilizing the self-attention mechanism as illustrated in Figure 2 [8].

The encoder of self-attention-based NMT first represents the source sentence $\chi = (\chi_1, \chi_2, \ldots, \chi_n)$ into the context vector $C = (h_1, h_2, \ldots, h_n)$ whose size is the same as with the source sentence. Then, the decoder generates the target translation $Y = (y_1, y_2, \ldots, y_m)$ with the help of the attention mechanism, in which the language model is conditioned on a previous word to generate the translation by maximizing the probability of $P(y_j \mid y_{<j}, C)$. Next, we briefly introduce the encoding process of the encoder and the translation process of the decoder.

The encoder consists of six identical layers. Each layer has two sublayers: one is the multihead attention and the other is a full connected feed-forward network. The input embedding matrix is re-represented to the context vector C by six identical layers. The multihead attention is

$$\text{multihead}(Q, K, V) = \text{concat}(\text{head}_1, \ldots, \text{head}_i)W^{\star}, \quad (1)$$

where Q is the query matrix, K is the key matrix, V is the value matrix, they are the same with the input word embedding matrix, and the dimension of $W^{\star}$ is $hd_{\text{head}} \times d_{\text{input}}$, which is to maintain the same dimension of the input embedding matrix. In the multihead attention mechanism, each head_i is an independent self-attention mechanism. The calculation process is as follows:

$$\begin{aligned} \text{head}_i &= \text{attention}\left(QW_i^q, KW_i^k, VW_i^v\right), \\ \text{attention}(Q, K, V) &= \text{soft max}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \end{aligned} \quad (2)$$

where different heads have different parameters, which makes each head learn different semantics. Therefore, the encoder based on the self-attention mechanism can solve the problem of anaphora resolution. In order to ensure the calculation of weights, W_i^q has the same dimension with W_i^k and then the weights of V are calculated by a dot product. W_i^q, W_i^k, and W_i^v transform the input word embedding into d_q, d_k, and d_v, respectively, for each head_i, which projects the word embedding to different representation subspaces. We describe the process of calculation in Figure 3 for easy understanding.

The full-connected feed-forward network is

$$\text{FFN}(a) = \max(0, W_1 a + b_1)W_2 + b_2, \quad (3)$$

where a is the output of the multihead attention and W_1, b_1 and W_2, b_2 are the parameters of the two liner transformation. Particularly, residual connection and layer norm are added to each sublayer to make the model converge faster. The layer norm is calculated as follows:

$$\widehat{x} = \frac{x_{ij} - \mu_i}{\sqrt{\delta^2 + \varepsilon}}, \quad (4)$$

where x_{ij} is an element of the input embedding vector, μ_i and δ are the mean and variance of the input embedding

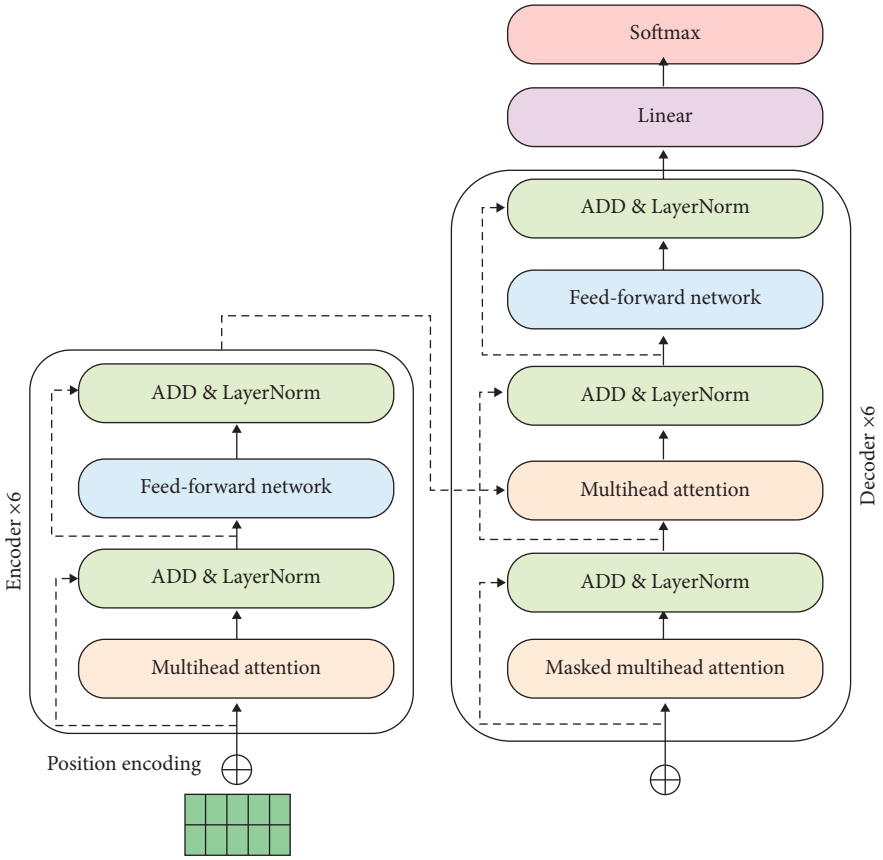

Figure 2: The model structure of the transformer.

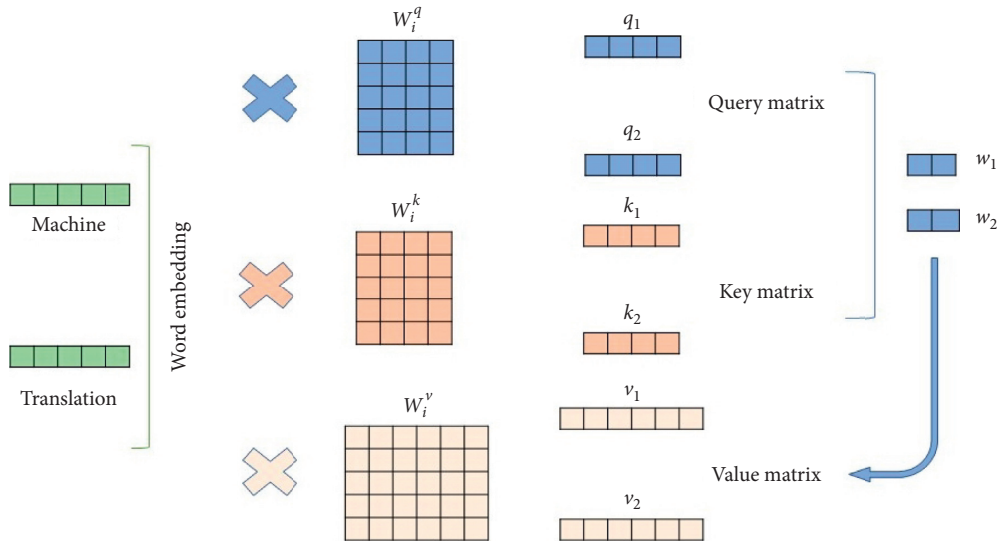

Figure 3: The calculation process of the attention mechanism.

vector, and ε is set to prevent the denominator from being 0. The active function for transformer is RELU:

$$\mathrm{RELU}(x) = \begin{cases} 0, & x < 0, \\ x, & x \geq 0. \end{cases} \tag{5}$$

The decoder also consist of six identical layers, but the difference is that each layer is composed by three sublayers. The first sublayer is the masked multihead attention, which focus only on the location of the input sequence earlier. The second sublayer is the multihead attention whose function is the same with the attention mechanism of attention-based

NMT. In this multihead attention, a query matrix is from the masked multihead attention, and key matrix and value matrix are the outputs of an encoder, which help a decoder get sequence information and alignment information of the source sequence. The third sublayer is the same feed-forward network. Residual connection and layer norm are also utilized by a decoder. The decoder starts with the beginning tag of the sentence. Then, target translation is generated by maximizing the conditional probability of $p(y_j \mid y_{<j}, C)$ through the final linear and softmax layer. The calculation of softmax is as follows:

$$\text{softmax}(a_i) = \frac{e^{a_i}}{\sum_j^n e^{a_j}}. \tag{6}$$

The softmax layer is utilized to normalize the output vector, where a_i is the ith item of the vector. For back-propagation, the loss function is a cross entropy loss function as follows:

$$\text{loss} = -[y \log \hat{y} + (1 - y)\log(1 - \hat{y})], \tag{7}$$

where y is the label of the real sample and $\hat{y}$ is the probability that the model predicts the sample. The optimizer is Adam, in which the parameter update process is

$$w_t = w_{t-1} - \mu^* \frac{\widehat{m}_t}{\sqrt{\widehat{v}_t} + \varepsilon}, \tag{8}$$

where w_t represents the parameters at time t, μ is the learning rate, ε is set to $10^{\wedge} - 8$ to avoid the denominator being 0, and the momentum $\widehat{m}_t$ that is used to speed up training is computed as

$$\begin{aligned} \widehat{m}_t &= \frac{m_t}{1 - \gamma_1} \\ &= \frac{\gamma_1 m_{t-1} + (1 - \gamma_1) g_t}{1 - \gamma_1}, \end{aligned} \tag{9}$$

where γ_1 is the exponential decay rate, g_t is the gradient at time t, and m_0 is initialized to 0. The gradient squared exponential moving average for the adjusting learning rate is

$$\begin{aligned} \widehat{v}_t &= \frac{v_t}{1 - \gamma_2} \\ &= \frac{\gamma_2 v_{t-1} + (1 - \gamma_2) g_t^2}{1 - \gamma_2}, \end{aligned} \tag{10}$$

where γ_2 is the exponential decay rate, g_t^2 is the square of the gradient at time t, and v_0 is initialized to 0. All the parameters of the self-attention-based NMT are optimized to the conditional log-likelihood of

$$L(\theta) = \frac{1}{K} \sum_{j=1}^{K} \log P\left(Y^j \mid X^j; \theta\right) = \frac{1}{K} \sum_{j=1}^{K} \sum_{i=1}^{n} \log P\left(y_i^j \mid y_{1:\, i-1}^j; \theta\right), \tag{11}$$

where K is the number of sentences in training data, n is the length of a sentence, y_i^j means the ith word of the jth sentence, and θ represents the parameters of the NMT model.

3. Methodology

We propose the joint method, which incorporates back-translation into diverse transfer learning architectures, for low-resource problems in this section. First of all, we give a description of mainstream transfer learning architectures for low-resource problems. Afterwards, we introduce the back-translation method for generating large-scale synthetic parallel language pairs. We then introduce the integration of the back-translation and the transfer learning methods.

3.1. Parent-Child Architecture. The parent-child architecture was proposed by Zoph et al. [19], which pioneer to exploit transfer learning methods to solve low-resource problems for neural machine translation. The parent model is trained on high-resource language pairs, which provides a strong prior distribution for the child model. All parameters of the child model are initialized by transferring the parameters of the parent model. Then, the parameters are fine-tuned on the low-resource language pairs.

3.2. Hierarchical Transfer Learning Architecture. We proposed a more effective hierarchical transfer learning architecture for low-resource languages [24]. Its core idea about transfer learning is the same with the parent-child architecture. However, the difference is that the hierarchical transfer learning architecture adds the intermediate layer to combine the data volume advantage of the high-resource language and the syntactic similarity advantage of the intermediate language. The first layer is trained on high-resource language pairs to get a prior distribution. Then, the second layer is trained on intermediate language pairs that are syntactically similar with the low-resource language. The third layer is trained on low-resource language pairs to achieve the ultimate goal. Models of the last two layers are initialized by transferring the parameters layer by layer, and the parameters are fine-tuned on corresponding language pairs after initialization.

3.3. Back-Translation. The back-translation was proposed by Sennrich et al. which translates large-scale monolingual data of target side to generate synthetic parallel language pairs, where large-scale target-side monolingual data boost the fluency of target translation [10]. This method first needs to train a target-to-source NMT model on the small-size parallel corpus. Then, large-scale monolingual target data are translated by the NMT model to generate synthetic parallel data. Finally, new training data are generated by mixing the original parallel corpus and the synthetic parallel corpus.

3.4. Method Integration. In order to improve the capacity of the NMT model for low-resource problems, we incorporate the back-translation into diverse transfer learning architectures, which not only can boost the fluency of target translations in the decoding process, but also can effectively initialize the parameters by transfer learning methods. To start with, we apply the back-translation method to generate

synthetic parallel data for low-resource languages and then mix it with real parallel data into mixed data as low-resource language pairs for diverse transfer learning architectures.

4. Experiment

In this section, we conduct experiments to evaluate the effectiveness of the joint method with diverse transfer learning architectures and to evaluate the generalization on different low-resource languages. Experimental settings and datasets are first described for the joint method. Afterwards, we introduce the three baselines. We then experiment the joint method on Uygur-Chinese and Turkish-English for generalization.

4.1. Datasets. We conduct experiments on two low-resource languages to verify the generalization of the method, one is Uygur-Chinese parallel corpus of news, which is used for the CWMT2017 Uygur-Chinese translation evaluation task. The other is a news-parallel corpus of Turkish-English, which is published on WMT 2016 [26]. For Uygur-Chinese, the validation set and test set come from the evaluation task of CWMT 2017. The target-side monolingual data are collected and organized from web pages and news broadcast. For Turkish-English, we divide the 2 K parallel corpus as the validation set and choose newtest-2016 as the test set, and the English monolingual corpus of the target side is taken from the UN corpus. The high-resource language is the English-Chinese parallel corpus that is published on the union corpus [27]. The validation set and test set are from newsdev-2017 and newtest-2017 [28]. The statistics of datasets are shown in Table 1.

4.2. Data Preprocessing. In data preprocessing process, high-resource language pairs (English-Chinese) and similar intermediate language pairs (Turkish-English) are processed by using BPE to segment words into subwords for word embedding presentation. For Uygur-Chinese, we utilize character-level embedding for the target Chinese to eliminate out-of-vocabulary words. It is suitable for Chinese to use character-level embedding, which not only can eliminate out-of-vocabulary words, but also can represent the semantic of sentences. However, since English, Turkish, and Uygur are composed of basic letter units whose number is particularly small, the character-level embedding will result in ambiguous representation.

4.3. Experiment Settings. All experiments are conducted on the NMT system Transformer, which is implemented by tensor2tensor version 1.11.0 [29]. We first prepare an experimental corpus and build an NMT system for the low-resource problems. In data preprocessing process, we utilize BPE and character-level embedding for word representation and vocabulary generation. In training process, each NMT model is trained on GPU Tesla K80 with 11 GB RAM. In order to get the best results, we set the hyperparameters of the transformer according to the training tips that were experimented by Popel and Bojar [30]. The batch size is 2048, the max-length of sentences is 256, and the dimension of the source and the target embedding is 1024. For the transformer-big model, the number of encoder and decoder layers is 6, the number of head is 16 for multihead attention and masked multihead attention, and the dimension of the hidden layer of the feed-forward network is 4096. In backpropagation process, the feed-forward network uses Adam as an optimizer. We set the learning rate as 2.0 and momentum term as 0.9 to adjust the converge rate. The dropout is set to 0.2 to get training out of the local optimal and to avoid overfitting. In transfer learning architectures, the hyperparameters of the transformer of each layer are set to be the same to maintain the same model structure for transferring parameters of models. Finally, the beam size N is set to 8 to get the best candidate translation.

4.4. Baselines. We set the following three baselines for the joint method. The methods of the three baselines are described in Section 3:

(i) Baseline 1 is the NMT system that uses the back-translation method to generate synthetic parallel data for low-resource languages

(ii) Baseline 2 is the parent-child transfer learning architecture for low-resource problems

(iii) Baseline 3 is the hierarchical transfer learning architecture, which combines the data volume advantage and the syntactic similarity advantage for low-resource languages

4.5. Uygur-Chinese. We first experiment with the three baselines on Uygur-Chinese. For baseline 1, we train a Chinese-Uygur NMT model on the 0.35 M parallel corpus. Afterwards, the monolingual data of Chinese, which are approximately three times the parallel corpus, are translated to Uygur to generate a synthetic parallel corpus. Subsequently, the new parallel corpus is generated by mixing 0.35 M parallel corpus and 0.9 M synthetic parallel corpus. The NMT model trains 0.5 M steps to converge on the new parallel corpus. In baseline 2, the parent model trains 0.5 M steps on the 15 M English-Chinese parallel corpus to get simple prior to distribution. The child model is initialized by the parameters of the parent model, and then the model trains 0.22 M steps on Uygur-Chinese to converge. In baseline 3, the difference is that we add the second layer and choose Turkish-English as an intermediate language. We train 50 K steps on the second layer and 0.25 M steps on the third layer to get the best result.

For the joint method with a parent-child transfer learning architecture, we first get a synthetic Uygur-Chinese parallel corpus by the back-translation method and then generate the new mixed parallel corpus. Afterward, the training process of the parent model is the same as with baseline 2, and the child model trains 0.5 M steps to converge on the new mixed parallel corpus.

For the joint method with a hierarchical transfer learning architecture, the first two layers are the same with baseline 3,

and the difference is that the NMT model has trained 0.45 M steps to converge on the new mixed parallel corpus in the third layer. Furthermore, we also experiment the word-level and character-level shared vocabulary for the joint method on Uygur-Chinese.

4.6. Turkish-English. For baseline 1, we train a English-Turkish NMT model on the 0.2 M parallel corpus (wmt 16). After that, the synthetic parallel corpus is generated by translating English monolingual data from the union corpus. The ratio of monolingual data and parallel data is 3 : 1. We train the NMT model 0.15 M steps to converge on the new mixed Turkish-English parallel corpus. The high-resource language of baseline 2 is the English-Chinese parallel corpus. The parent model is trained on English-Chinese for 0.5 M steps, and the child model is trained on Turkish-English for 50 K step to converge. In baseline 3, the Uygur-Chinese corpus is set as the intermediate layer. We separately train the NMT model 0.5 M steps, 0.1 M steps, and 50 K steps on the three layers.

For the joint method with the parent-child transfer learning architecture, like the baseline 1, we first get the new mixed Turkish-English parallel corpus. Next, the parent model is trained on English-Chinese for 0.5 M steps. Afterwards, the child model is trained on the new Turkish-English parallel corpus for 0.12 M steps to converge.

For the joint method with a hierarchical transfer learning architecture, the difference with baseline 3 is the third layer. We get the new mixed parallel corpus and train 0.12 M steps for the NMT model to converge.

5. Results and Analysis

In this section, we separately compare the experimental results of the joint methods with three baselines on Uygur-Chinese and Turkish-English. According to the results we give specific analysis.

5.1. Uygur-Chinese. For Uygur-Chinese, experimental results of three baselines and the joint method with diverse transfer learning architectures are given in Table 2. Form Table 2, we can see that compared to pure back-translation and single parent-child transfer learning architecture, the joint method with parent-child architecture separately improves **0.4** BLEU scores and **2.08** BLEU scores. The reason lies in that because the back-translation method need not to change the structure of NMT models for low-resource problems, and the joint method can perfectly combine the back-translation method with transfer learning architectures, which not only can initialize the parameters of multihead attention, feed-forward network, and masked multihead attention by transferring parameters from the NMT model trained on English-Chinese, but also can generate a large-scale synthetic parallel corpus for Uygur-Chinese by translating large-scale Chinese monolingual data. For the joint method with hierarchical transfer learning architecture, we can see that the joint method separately improves **0.8** BLEU scores and **1.9** BLEU scores compared with the pure back-translation method and the hierarchical transfer learning method. Figure 4 shows the samples of Uygur-Chinese translation and the corresponding English translation version. From this sample, we can see that the translation of the joint method is not only accurate but also more fluent, which demonstrates the effectiveness of the joint method for the hierarchical transfer learning architecture. The reasons are the same with previous analysis. Specially, compared with the pure back-translation method, the improvements of the joint method with hierarchical transfer learning architecture is more obvious than that with the parent-child architecture. The reasons are that the hierarchical transfer learning architecture combines the data volume advantage of English and the syntactic similarity advantage of Turkish for Uygur in an encoder.

Table 2: The BLEU score of the joint method and three baselines on Uygur-Chinese.

Method	BLEU score
Back-translation (BT)	36.61
Parent-child architecture	34.93
Hierarchical transfer learning architecture	35.51
Parent-child architecture + BT	37.01
Hierarchical transfer learning architecture + BT	37.41

Besides, the joint method with hierarchical transfer learning architecture gets the best results on Uygur-Chinese. The reason is that the advantages of back-translation and transfer learning methods, which have been testified to be helpful for low-resource problems, are superimposed in the joint method. The BLEU scores of the joint method with transfer learning architectures are shown in Figure 5.

Furthermore, form Table 3, we can find that the character-level shared vocabulary is more suitable for Chinese in low-resource NMT compared with the word-level shared vocabulary. The reason lies in that the character-level embedding, which can completely eliminate OOV words, is more effective for the data scarcity problem. In addition, for the joint method with hierarchical transfer learning architecture, the more convergent the first layer is trained, the better the transfer learning method effects, which provides a stronger prior distribution model.

5.2. Turkish-English. We compare the joint method with several baselines on Turkish-English to explore the generalization of the joint method. All experimental results are given in Table 4. The parent-child architecture with the back-translation method separately outperforms the pure back-translation method and the transfer learning method with parent-child architecture **2.91** BLEU scores and **1.58** BLEU scores. The hierarchical transfer learning architecture with the back-translation method separately improves **3.51** BLEU scores, **1.45** BLEU scores, and **0.6** BLEU scores compared with pure back-translation method, the single hierarchical transfer learning architecture, and the joint method with parent-child architecture. The joint method with diverse transfer learning architectures all have achieved significant improvement on Turkish-English and Uygur-Chinese which

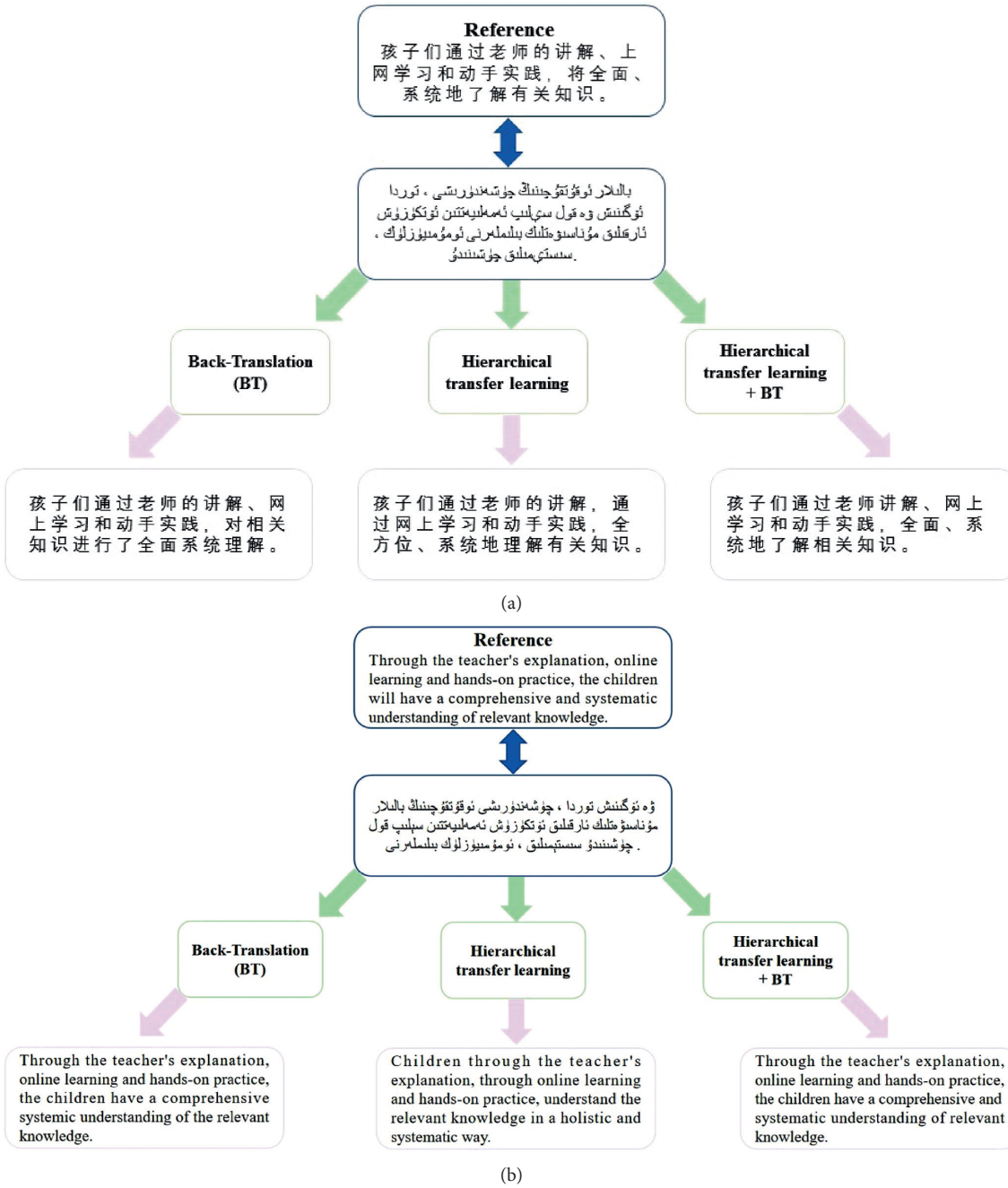

Figure 4: The translations for the joint method with hierarchical transfer learning architecture and baselines. (a) Uygur-Chinese translation results. (b) English translation of Uygur-Chinese translation results.

shows that the joint method has excellent generalization performance in low-resource languages.

Besides these, Figure 6 shows the degree to which the joint method improves the different transfer learning architectures. It is obvious that the back-translation method is more effective in improving the effect of transfer learning architectures in Uygur-Chinese. We speculate that this is due to the quality of the synthetic parallel data generated by the back-translation method. Since Uygur-Chinese has more parallel data compared with Turkish-English, the reverse translation model trained on Chinese-Uygur can generate higher quality translations, which improves the quality of the synthetic parallel corpus.

6. Related Work

Low-resource language problems in the field of machine translation have drawn more and more attention from both research and industry communicates in recent years. To our knowledge, for the data scarcity problem of low-resource languages, researchers have done a lot of studies. A straightforward way is to generate more parallel data for low-

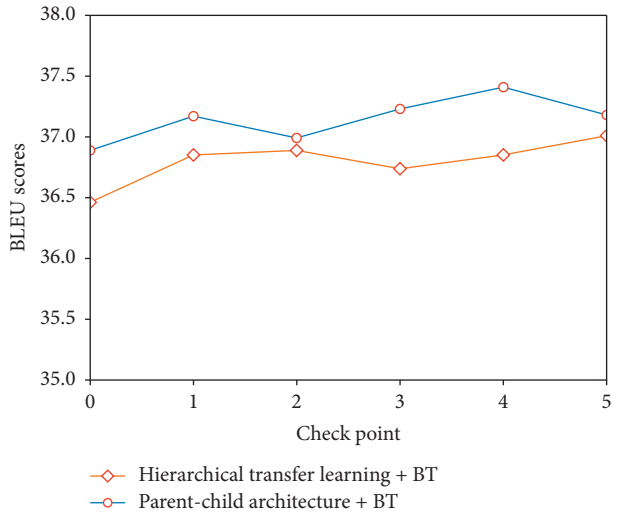

Figure 5: The BLEU scores of the joint method with diverse transfer learning architectures.

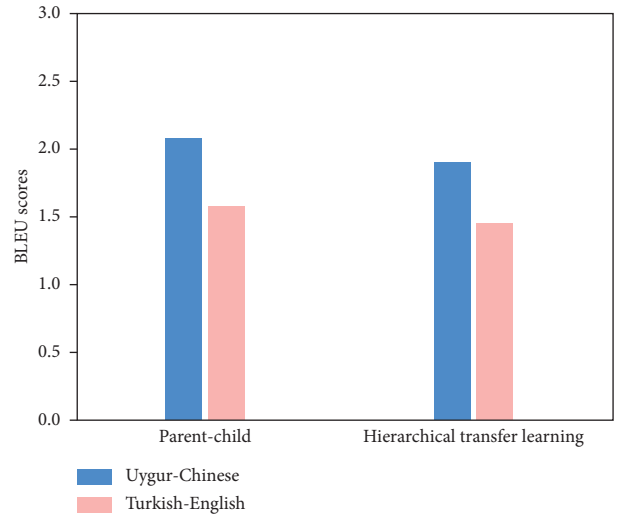

Figure 6: The BLEU scores to which the joint method improves the two transfer learning architectures.

Table 3: The BLEU scores of the joint method with hierarchical transfer learning architecture separately on character-level and word-level shared vocabulary for Uygur-Chinese.

Shared vocabulary	Steps (first layer) (M)	BLEU scores
Word level	0.5	36.65
Word level	1	37.05
Character level	0.5	37.31
Character level	1	37.64

Table 4: The BLEU score of the joint method and three baselines on Turkish-English.

Method	Newtest-2016
Back-translation (BT)	16.40
Parent-child architecture	17.73
Hierarchical transfer learning architecture	18.46
Parent-child architecture + BT	19.31
Hierarchical transfer learning architecture + BT	19.91

resource languages by using large monolingual data in both encoder and decoder [9–14]. The other effective way is the transfer learning method, which shares an encoder and decoder or transfers the parameters of pretrained models for initialization [17, 19, 21–23]. Besides the two mainstream methods, in [31], Ren et al. proposed a TA-NMT model to improve the translation performance for low-resource languages by using the unified bidirectional EM algorithm. For zero-resource languages, in [32], Chen et al. proposed a teacher-student architecture to avoid the accumulation of errors in the pivot-based method [33], which bases on the assumption that parallel sentences have close probabilities of generating a sentence in a third language. Inspired by multilingual NMT, Gu et al. improved translation quality for tiny even zero-resource parallel corpus by sharing a universal word-level representation and sentence-level representation [34]. For zero-resource parallel corpus, Lakew et al. generated new synthetic data by the training-inference-training scheme, which is based on a multi-NMT system [35]. Xia et al. proposed a dual-learning algorithm called dual-NMT to tackle the training data bottleneck, which teach each other by giving feedback signals [36]. In [37], Gu et al. treat low-resource translation as a metal-earning problem, which is solved by the model-agnostic metalearning algorithm.

In order to further improve the translation performance of low-resource languages, we proposed the joint method, which incorporates data augmentation method into transfer learning architectures. In addition, we carefully investigate the effectiveness of the joint method.

7. Conclusion

In this paper, we propose a joint method to further improve the translation performance for low-resource languages, which incorporates data augmentation methods into diverse transfer learning architectures. Unlike exiting single methods that improve the quality of translations from different angles, which is not effective enough for low-resource problems, we find that the data augmentation methods can be perfectly integrated into diverse transfer learning architectures. Therefore, the proposed method simultaneously makes full use of the advantages of data augmentation methods and transfer learning methods. In order to evaluate the effectiveness of the joint method, we compared the joint method with the baseline methods on the parent-child transfer learning architecture and the hierarchical transfer learning architecture separately. Furthermore, the generalization of the method is testified by conducting experiments on Uygur-Chinese and Turkish-English. Experimental results show that the joint method significantly improves the translation performance compared with the baseline methods and has excellent generalization for other low-resource languages.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (U1703133); Western Light of the Chinese Academy of Sciences (2017-XBQNXZ-A-005); Subsidy of the Youth Innovation Promotion Association of the Chinese Academy of Sciences (2017472); Major Science and Technology Project of Xinjiang Uygur Autonomous Region (2016A03007-3); Xinjiang Uygur Autonomous Region Level Talent Introduction Project (Y839031201); and the Xinjiang Key Laboratory Fund under grant no. 2018D04018.

References

[1] P. Koehn, "Neural machine translation," 2017, http://arxiv.org/abs/1709.07809.

[2] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proceedings of the Empirical Methods in Natural Language Processing*, Washington, DC, USA, 2013.

[3] K. Cho, B. van Merrienboer, C. Gulcehre et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, Doha, Qatar, October 2014.

[4] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 3104–3112, Bangkok, Thailand, November 2014.

[5] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the International Conference on Learning Representations*, San Diego, CA, USA, May 2015.

[6] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421, Lisbon, Portugal, September 2015.

[7] Y. Wu, M. Schuster, Z. Chen et al., "Google's neural machine translation system: bridging the gap between human and machine translation," 2016, http://arxiv.org/abs/1609.08144.

[8] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *Proceedings of the Conference on Neural Information Processing Systems*, Long Beach, CA, USA, December 2017.

[9] C. Gulcehre, O. Firat, K. Xu et al., "On using monolingual corpora in neural machine translation," 2015, http://arxiv.org/abs/1503.03535.

[10] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, August 2016.

[11] J. Zhang and C. Zong, "Exploiting source-side monolingual data in neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, TX, USA, 2016.

[12] Y. Cheng, W. Xu, Z. He et al., "Semi-supervised learning for neural machine translation," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, August 2016.

[13] M.-T. Luong, Q. Le, L. Sutskever, O. Vinyals, and L. Kaiser, "Multi-task sequence to sequence learning," in *Proceedings of the ICLR*, San Juan, PR, USA, 2016.

[14] M. Fadaee, A. Bisazza, and C. Monz, "Data augmentation for low-resource neural machine translation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 567–573, Vancouver, Canada, July 2017.

[15] A. Currey, A. V. M. Barone, and K. Heafield, "Copied monolingual data improves low-resource neural machine translation," in *Proceedings of the Second Conference on Machine Translation*, Copenhagen, Denmark, September 2017.

[16] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[17] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, "Multi-task learning for multiple language translation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Taipei, Taiwan, November 2015.

[18] O. Firat, K. Cho, and Y. Bengio, "Multi-way, multilingual neural machine translation with a shared attention mechanism," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, CA, USA, June 2016.

[19] B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1568–1575, Austin, TX, USA, 2016.

[20] R. Sennrich, B. Haddow, and A. Birch, "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)*, pp. 1715–1725, Berlin, Germany, August 2016.

[21] T. Q. Nguyen and D. Chiang, "Transfer learning across low-resource related languages for neural machine translation," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pp. 296–301, Taipei, Taiwan, November 2017.

[22] R. Dabre, T. Nakagawa, and H. Kazawa, "An empirical study of language relatedness for transfer learning in neural machine translation," in *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, Cebu City, Philippines, November 2017.

[23] T. Kocmi and O. Bojar, "Trivial transfer learning for low-resource neural machine translation," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, Brussels, Belgium, October 2018.

[24] G. Luo, Y. Yang, Y. Yuan, Z. Chen, and A. Ainiwaer, "Hierarchical transfer learning architecture for low-resource neural machine translation," *IEEE Access*, vol. 7, pp. 154157–154166, 2019.

[25] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for*

Computational Linguistics, pp. 311–318, Stroudsburg, PA, USA, 2002.

[26] J. Tiedemann, "Parallel data, tools and interfaces in OPUS," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, Istanbul, Turkey, May 2012.

[27] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen, "The united nations parallel corpus v1. 0," in *Proceedings of the 2016 International Conference on Language Resources and Evaluation*, Portorož, Slovenia, May 2016.

[28] O. Bojar, R. Chatterjee, and C. Federmann, "Findings of the 2017 conference on machine translation (WMT17)," in *Proceedings of the Second Conference on Machine Translation*, Copenhagen, Denmark, September 2017.

[29] A. Vaswani, S. Bengio, E. Brevdo et al., "Tensor2tensor for neural machine translation," in *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pp. 193–199, Boston, MA, USA, 2018.

[30] M. Popel and O. Bojar, "Training tips for the transformer model," *The Prague Bulletin of Mathematical Linguistics*, vol. 110, no. 1, pp. 43–70, 2018.

[31] S. Ren, W. Chen, S. Liu, M. Li, M. Zhou, and S. Ma, "Triangular architecture for rare language translation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia, July 2018.

[32] Y. Chen, Y. Liu, Y. Cheng, and V. O. K. Li, "A teacher-student framework for zero-resource neural machine translation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, July 2017.

[33] Y. Cheng, Y. Liu, Q. Yang, M. Sun, and W. Wu, "Neural machine translation with pivot languages," 2016, http://arxiv.org/abs/1611.04928.

[34] J. Gu, H. Hassan, J. Devlin, and V. O. K. Li, "Universal neural machine translation for extremely low resource languages," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, LA, USA, January 2018.

[35] S. M. Lakew, Q. F. Lotito, M. Negri, M. Turchi, and M. Federico, "Improving zero-shot translation of low-resource languages," in *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, Tokyo, Japan, December 2017.

[36] Y. Xia, D. He, T. Qin et al., "Dual learning for machine translation," in *Proceedings of the Advances in Neural Information Processing Systems*, Barcelona, Spain, December 2016.

[37] J. Gu, Y. Wang, Y. Chen, K. Cho, and V. O. K. Li, "Meta-learning for low-resource neural machine translation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, October 2018.