High-resource Language-specific Training for Multilingual Neural Machine Translation

Jian Yang1*, Yuwei Yin2*, Shuming Ma2, Dongdong Zhang2, Zhourun Li1†, Furu Wei2
1State Key Lab of Software Development Environment, Beihang University
2Microsoft Research
{jiaya, lizj}@buaa.edu.cn, {v-yuweiyin, shumma, dozhang, fuwei}@microsoft.com

Abstract
Multilingual neural machine translation (MNMT) trained in multiple language pairs has attracted considerable attention due to fewer model parameters and lower training costs by sharing knowledge among multiple languages. Nonetheless, multilingual training is plagued by language interference degeneration in shared parameters because of the negative interference among different translation directions, especially on high-resource languages. In this paper, we propose the multilingual translation model with the high-resource language-specific training (HLT-MT) to alleviate the negative interference, which adopts the two-stage training with the language-specific selection mechanism. Specifically, we first train the multilingual model only with the high-resource pairs and select the language-specific modules at the top of the decoder to enhance the translation quality of high-resource directions. Next, the model is further trained on all available corpora to transfer knowledge from high-resource languages (HRLs) to low-resource languages (LRLs). Experimental results show that HLT-MT outperforms various strong baselines on WMT-10 and OPUS-100 benchmarks. Furthermore, the analytic experiments validate the effectiveness of our method in mitigating the negative interference in multilingual training.

1 Introduction
Recent advances in multilingual neural machine translation (MNMT) aim to build and deploy a single universal model in real industrial scenarios, which supports multiple translation directions by sharing model parameters [Firat et al., 2016; Johnson et al., 2017; Aharoni et al., 2019; Fan et al., 2020; Lin et al., 2020]. Furthermore, parameter sharing across various languages encourages knowledge transfer, especially from the high-resource language (HRL) to low-resource language (LRL) and even enables zero-shot translation [Aharoni et al., 2019; Zhang et al., 2020].

* Contribution during internship at Microsoft Research.
† Corresponding author.
the language-specific modules to avoid the negative impact caused by LRLs. To address the negative interference problem among HRLs, we introduce a language-specific pool containing a sequence of independent modules for HRLs. Considering the increasing number of the languages, we apply the selection mechanism to the language-specific pool with the constrained size, denoted as selective language-specific pool (SLP), which enables different groups of certain languages to share the same module from SLP. After pretraining with the high-resource languages, we extract the shared representations of the decoder to transfer the fine-trained knowledge to the low-resource languages.

Our method aims to enhance translation quality of high-resource directions with the selective language-specific pool compared to the bilingual counterpart and then transfer the knowledge to the low-resource directions based on the bottom shared features. We conduct experiments on the WMT-10 benchmark of 11 languages and OPUS-100 benchmark of 95 languages. Experimental results demonstrate that our method significantly outperforms previous bilingual and multilingual baselines. Besides, extensive probing experiments are performed for the multilingual baseline and HLT-MT, helping further analyze how our method can benefit the multilingual machine translation. Empirical studies show that HLT-MT maintains a balance between language-agnostic and language-distinct features and thus helps to alleviate the negative language interference among various languages.

2 Negative Language Interference

Multilingual translation model aims at transferring knowledge across languages to boost performance on low-resource languages, where the multilingual model is trained in multiple translation directions simultaneously to enable cross-lingual transfer through parameter sharing. However, different groups of languages have heterogeneous characteristics, such as different dictionaries and grammars. The previous works have shown [Wang et al., 2020b; Yu et al., 2020] that knowledge transfer is not beneficial for all languages by sharing all parameters. To analyze the effect of mutual influence among different languages, we calculate the cosine similarities between gradients of two translation directions. In Figure 3, we observe that the certain high-resource language sustains negative interference from other HRLs and LRLs. For example, En→Cs acutely conflicts with En→De and En→Hi in the second row of Figure 3. This shows that HRLs are conducive to LRLs but may be hindered by other HRLs and LRLs in turn. To prevent the HRLs from negative interference introduced by LRLs, we focus on sufficiently training the high-resource directions and then continue tuning on all directions. To further address the conflicts among HRLs, we propose the selective language-specific pool for different high-resource languages. Our method effectively mitigates the negative interference in the analytic experiments compared to baselines.

3 Our Method

In this section, we introduce HLT-MT for multilingual translation. We propose the two-stage training framework, where the selective languages-specific pool (SLP) and the universal layer are applied for HRLs and LRLs respectively.

3.1 Overview of HLT-MT

Given different translation directions of $K$ languages $L_{all} = \{L_k\}_{k=1}^K$, we employ SLP for high-resource directions and the universal layer for low-resource directions. Our method is illustrated in Figure 2, where the selective language-specific pool (SLP) for the high-resource languages is inserted into the top of the decoder denoted by $\theta_u$. The constrained size of the selective language-specific pool and
be simplified into $g(L_k)$. Therefore, we project the language-agnostic representations $h_s^{L_k}$ to the language-specific features $h_b^{L_k}$ using function $F_{\theta_g(L_k)}$ as below:

$$h_b^{L_k} = F_{\theta_g(L_k)}(h_s^{L_k})$$  \hspace{1cm} (3)

where $h_b^{L_k}$ is the representations generated by the shared parameters. $F_{\theta_g(L_k)}$ is a function defined as below:

$$F_{\theta_g(L_k)}(h_s^{L_k}) = f(W_u(g(L_k)) \sigma(W_g(g(L_k)) h_s^{L_k}) + h_b^{L_k})$$  \hspace{1cm} (4)

where $f(\cdot)$ is the layer normalization and $\sigma(\cdot)$ is the ReLU activation function. $W_u(g(L_k)) \in R^{d_u \times h_b}$ is the up-projection matrix and $W_d(g(L_k)) \in R^{h_b \times d_e}$ is the down-projection matrix as shown in the right part of Figure 2, where $d_e$ and $d_h$ are the embedding size and hidden size of SLP ($d_e < d_h$).

Another issue is how to design a proper map function $g(\cdot)$ with an appropriate selection mechanism for the translation direction $L_i \rightarrow L_k$. In our work, each source sequence is prefixed with a special target language symbol to indicate the translation direction, which enables the decoder to correctly generate the target sentence with the shared decoder parameters. Therefore, the embedding of the target language symbol is used to select the language-specific module from SLP. The selection function $g(\cdot)$ is defined as:

$$g(L_k) = \arg \max \limits_{1 \leq T \leq T} \exp(e^{L_k}_i) \sum_{t=1}^{T} \exp(e^{L_k}_t)$$  \hspace{1cm} (5)

where $e^{L_k} = W_g E[L_k]$ of $T$ dimensions. $E[L_k]$ denotes the embedding of the target language $L_k$ symbol. $W_g \in R^{d_e \times T}$ maps the target embedding to the vector $e^{L_k}$, where $e^{L_k}_i$ is the $i$-th element of $e^{L_k}$ and SLP is comprised of $T$ sub-networks. The sub-network with the highest probability will be selected to produce the language-specific features.

Equation 3 and 5 are only related to $\theta_g(L_k)$ and thus can not propagate gradients to all language-specific parameters. The selective language-specific pool $\theta_p = \{\theta_i\}_{i=1}^{T}$ contains a set of modules described in Equation 4. To tackle the undifferentiable problem of SLP, we use the weighted average to ensure gradients to be propagated to all language-specific modules:

$$h_b^{L_k} = \sum_{i=1}^{T} \alpha_i^{L_k} F_{\theta_i}(h_s^{L_k})$$  \hspace{1cm} (6)

where $\alpha_i^{L_k}$ is calculated by the target embedding and softmax function. We project the target embedding $E[L_k]$ the probability vector $e^{L_k} = W_g E[L_k]$ with the learned matrix $W_g$.

$$\alpha_i^{L_k} = \frac{\exp(e^{L_k}_i)}{\sum_{t=1}^{T} \exp(e^{L_k}_t)}$$  \hspace{1cm} (7)

where $e^{L_k} = W_g E[L_k]$ of $T$ dimensions. $W_g \in R^{d_e \times T}$ and $E[L_k]$ denote the language embedding of $L_k$. $d_e$ is the embedding size. $\alpha_i^{L_k}$ is the $i$-th element of the vector $\alpha^{L_k}$.

In the practice training, we alternately leverage the Equation 3 and 6 with equal probabilities to learn the map function and generate the language-distinct representations $h_b^{L_k}$. Finally, the representations are used to generate the target sentence $x^{L_k}$:

$$x^{L_k} = \text{softmax}(W^o h_b^{L_k})$$  \hspace{1cm} (8)

where $x^{L_k}$ is the target sentence and $W^o \in d_e \times V$ is the output matrix, where $V$ is the vocabulary size.

$T \leq K$ since the number of languages $K$ can be numerous. Thus, SLP contains $T$ individual modules with the same architecture, and each $\theta_i$ is a sub-network. If the translation direction $L_i \rightarrow L_k$ is the high-resource direction (from the source language $L_i$ to target language $L_k$), we only activate the relevant language-specific module from SLP for $L_k$. Otherwise, the universal layer is triggered for $L_k$.

### 3.2 Multilingual Machine Translation

Given the high-resource bilingual corpora $D^h = \{D^h_m\}_{m=1}^{M}$ and low-resource corpora $D^l = \{D^l_n\}_{n=1}^{N}$, where $M$ and $N$ separately represent the number of the high-resource and low-resource training corpora of $K$ languages $L_{all} = \{L_k\}_{k=1}^{K}$. The multimodal model is jointly trained on the union of the high- and low-resource training corpora $D^h \cup D^l$:

$$L_{MT} = - \sum_{m=1}^{M} \sum_{x,y \sim D^h_m} \log P(y|x; \Theta) \tag{1}$$

$$- \sum_{n=1}^{N} \sum_{x,y \sim D^l_n} \log P(y|x; \Theta)$$

where the first and second term denote objective of the high- and low-resource training corpora respectively. $\Theta$ are shared parameters for all languages. $x$ and $y$ are the sentence pair.

### 3.3 High-resource Language-specific Training

To prevent the high-resource languages from the negative interference caused by low-resource languages, we only train the model with SLP on high-resource directions, which effectively ameliorates translation quality of high-resource translation directions with slight extra parameters.

To take advantages of the cross-lingual pretrained encoder to boost model performance, our multimodal model is initialized by XLM-R [Conneau et al., 2020]. Besides, we verify the effectiveness of our method on the Transformer model [Vaswani et al., 2017] without any pretrained model. Given the source sentence $x^{L_i} = \{x_1^{L_i}, \ldots, x_n^{L_i}\}$ with $n$ words and target sentence $x^{L_k} = \{x_1^{L_k}, \ldots, x_v^{L_k}\}$ with $v$ words, the shared features $h_s^{L_k}$ of the target language $L_k$ at the top of the decoder are obtained by the Transformer model:

$$h_s^{L_k} = \text{Transformer}(x^{L_i}, x^{L_k}; \Theta) \tag{2}$$

where $h_s^{L_k} = \{h_s^{L_k}_1, \ldots, h_s^{L_k}_i, \ldots, h_s^{L_k}_v\}$ and $h_s^{L_k}_i$ is the $i$-th representation of the target token $h_s^{L_k}_i$ generated by the single shared Transformer encoder and decoder.

After obtaining a sequence of decoder representations $h_s^{L_k}$ of the high-resource language $L_k$, we project the language-agnostic representations to the language-distinct ones via the language-specific pool with the selective mechanism. Given the translation direction $L_i \rightarrow L_k$ ($1 \leq i, k \leq K$ and $i \neq k$) and the selective language-specific pool $\theta_p = \{\theta_i\}_{i=1}^{T}$, the corresponding module $\theta_{g(L_i), L_k}$ is used to generate the language-distinct representations. $g(\cdot)$ is a map function that maps the language index to the corresponding module index: $L_k \in \{1, \ldots, K\} \rightarrow t \in \{1, \ldots, T\}$, $g(L_i, L_k)$ is the map function only depending on the target language and thus can
3.4 Low-resource Transfer

After training the multilingual model on the high-resource language pairs, the bottom features generated by the shared parameters are utilized for the low-resource target sentence generation. Then, our multilingual model is jointly trained on the high-resource bilingual corpora $D_h^i$ and low-resource bilingual corpora $D_l^j$. Given the low-resource translation direction $L_i \rightarrow L_j$, the shared representations $h_y^i$ generated by the shared parameters $\Theta$ are fed into a universal layer $\theta_u$:

$$h_y^i = F_{\theta_u}(h_y^i)$$  \hspace{1cm} (9)

where $h_y^i$ are features generated by the shared parameters in Equation 2. $\theta_u$ is a sub-network same as the $\theta_1$ ($1 \leq t \leq T$) of SLP $\theta_p = \{\theta_t\}_{t=1}^T$. All low-resource languages share the same universal layer to project the shared features $h_y^i$ to the last representations $h_y^j$. Then the target sentence is produced by $h_y^j$ and output matrix $W^o$ similar to Equation 8.

3.5 Training Strategy

Our model first accumulates the cross-entropy loss on the high-resource pairs and the disparity loss. Then, the multilingual model trained on multilingual corpora to maintain the performance of high-resource languages and meanwhile transfer the knowledge to the low-resource languages.

Disparity Loss To encourage different languages to select different language-specific modules of SLP, we minimize the disparity loss $L_d$, which measures the similarity of language-specific layer selection among languages.

$$L_d = \sum_{i=1}^{N} \sum_{j=i+1}^{N} (\alpha_{L_i} \cdot \alpha_{L_j})$$  \hspace{1cm} (10)

where $\alpha_{L_i}, \alpha_{L_j} \in R^T$ denote the selection probabilities generated by Equation 7, where SLP contains $T$ modules.

High-resource Language-specific Training The objective is to minimize the cross-entropy loss of high-resource training corpora and the auxiliary disparity loss jointly as below:

$$L_{high} = \sum_{m=1}^{M} E_{x,y \sim D_m^h} \log P(y|x; \Theta, \theta_p) + L_d$$  \hspace{1cm} (11)

where $\Theta$ denotes all shared parameters and $\theta_p$ are parameters of SLP. $x$ and $y$ are sentence pair.

Multilingual Training After trained on the high-resource directions, the multilingual model is continued to be tuned on the union of the high- and low-resource corpora $D_h^i \cup D_l^j$ with the extra SLP for high-resource languages and the universal layer for low-resource languages:

$$L_{all} = \sum_{m=1}^{M} E_{x,y \sim D_m^h} \log P(y|x; \Theta, \theta_p)$$

$$- \sum_{n=1}^{N} E_{x,y \sim D_n^l} \log P(y|x; \Theta, \theta_n)$$

where $\Theta$ are shared parameters for all languages. SLP contains a list of language-specific layers for HRLs and $\theta_u$ is a universal layer for LRLs.

4 Experiments

4.1 Datasets

To evaluate our method, we conduct experiments on the WMT-10 and the OPUS-100 dataset.

WMT-10 We use a collection of parallel data in different languages from the WMT datasets to evaluate the models [Wang et al., 2020]. The parallel data is between English and other 10 languages, including French (Fr), Czech (Cs), German (De), Finnish (Fi), Latvian (Lv), Estonian (Et), Romanian (Ro), Hindi (Hi), Turkish (Tr) and Gujarati (Gu).

OPUS-100 We use the OPUS-100 corpus [Zhang et al., 2020] for massively multilingual machine translation. OPUS-100 is an English-centric multilingual corpus covering 100 languages, which is randomly sampled from the OPUS collection. We obtain 94 English-centric language pairs after dropping out 5 languages, which lack corresponding test sets.

4.2 Baselines

Our method is compared to the bilingual and multilingual methods. For a fair comparison, XLM-R and LS-MNMT are initialized by XLM-R [Conneau et al., 2020]. BiNMT [Vaswani et al., 2017] is the bilingual Transformer model. MNMT [Johnson et al., 2017] is jointly trained on all directions, where the target language symbol is prefixed to the input sentence. mBART [Liu et al., 2020] is an encoder-decoder pretrained model and then is finetuned on all corpora. XLM-R [Conneau et al., 2020] is initialized by the pretrained model XLM-R [Ma et al., 2020]. LS-MNMT [Fan et al., 2020] integrates the language-specific layers of all languages into the end of the decoder.

4.3 Training and Evaluation

We adopt Transformer as the backbone model for all experiments. We train multilingual models with Adam ($\beta_1 = 0.9$, $\beta_2 = 0.98$). The learning rate is set as 5e-4 with a warm-up step of 4,000. The models are trained with the label smoothing cross-entropy with a smoothing ratio of 0.1. The batch size is 4096 tokens on 64 Tesla V100 GPUs. For WMT-10, we first train the multilingual model with 6 languages and then finetune on all languages. For OPUS-100, the model is trained in the languages where the number of pairs exceeds 10K. The evaluation metric is the case-sensitive detokenized sacreBLEU [Post, 2018].

4.4 Main Results

WMT-10 As shown in Table 1, our method clearly improves multilingual baselines by a large margin in 10 translation directions. Previously, multilingual machine translation underperforms the bilingual translation model in rich-resource scenarios. It is worth noting that our multilingual machine translation baseline XLM-R is already very competitive initialized by the cross-lingual pretrained model. Interestingly, our method outperforms the bilingual baseline in the high-resource translation direction, such as En→De translation direction (+1.5 BLEU points). Our method consistently outperforms the multilingual baseline on all language pairs, confirming that using HLT-MT to alleviate negative interference can help boost performance.
Table 1: En→X evaluation results for bilingual (1→1), one-to-many (1→N), and many-to-many (N→N) models on WMT-10. The languages are ordered from high-resource languages (left) to low-resource languages (right).

<table>
<thead>
<tr>
<th>Models (N→N)</th>
<th>#Params</th>
<th>Fr</th>
<th>Cs</th>
<th>De</th>
<th>Fi</th>
<th>Lv</th>
<th>Et</th>
<th>Ro</th>
<th>Hi</th>
<th>Tr</th>
<th>Gu</th>
<th>AvgEnX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Previous Best System [Zhang et al., 2020]</td>
<td>254M</td>
<td>30.3</td>
<td>32.6</td>
<td>31.9</td>
<td>31.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>23.7</td>
</tr>
<tr>
<td>MNMT [Johnson et al., 2017]</td>
<td>242M</td>
<td>32.3</td>
<td>35.1</td>
<td>35.8</td>
<td>33.9</td>
<td>ref</td>
<td>26.3</td>
<td>31.4</td>
<td>31.2</td>
<td>28.9</td>
<td>ref</td>
<td>-</td>
</tr>
<tr>
<td>mBART [Liu et al., 2020]</td>
<td>611M</td>
<td>32.4</td>
<td>19.0</td>
<td>37.0</td>
<td>13.2</td>
<td>17.0</td>
<td>19.5</td>
<td>25.1</td>
<td>15.7</td>
<td>16.7</td>
<td>14.2</td>
<td>21.0</td>
</tr>
<tr>
<td>XLM-R [Conneau et al., 2020]</td>
<td>362M</td>
<td>34.2</td>
<td>21.4</td>
<td>39.7</td>
<td>15.3</td>
<td>18.9</td>
<td>20.6</td>
<td>26.5</td>
<td>15.6</td>
<td>17.5</td>
<td>14.5</td>
<td>22.4</td>
</tr>
<tr>
<td>LS-MNMT [Fan et al., 2020]</td>
<td>409M</td>
<td>34.8</td>
<td>21.1</td>
<td>39.3</td>
<td>15.2</td>
<td>18.7</td>
<td>20.5</td>
<td>26.3</td>
<td>14.9</td>
<td>17.3</td>
<td>12.3</td>
<td>22.0</td>
</tr>
<tr>
<td>HLT-MT (Our method)</td>
<td>381M</td>
<td>35.8</td>
<td>22.4</td>
<td>41.5</td>
<td>16.3</td>
<td>19.6</td>
<td>21.0</td>
<td>26.6</td>
<td>15.7</td>
<td>17.6</td>
<td>14.7</td>
<td>23.1</td>
</tr>
</tbody>
</table>

Table 2: X→En and En→X test BLEU for high/medium/low resource language pairs in many-to-many setting on OPUS-100 test sets. The BLEU scores are average across all language pairs in the respective groups. “WR”: win ratio (%) compared to ref (MNMT).

<table>
<thead>
<tr>
<th>Models (N→N)</th>
<th>#Params</th>
<th>X→En</th>
<th>En→X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Previous Best System [Zhang et al., 2020]</td>
<td>254M</td>
<td>High45</td>
<td>Med21</td>
</tr>
<tr>
<td>MNMT [Johnson et al., 2017]</td>
<td>242M</td>
<td>32.3</td>
<td>35.1</td>
</tr>
<tr>
<td>mBART [Liu et al., 2020]</td>
<td>611M</td>
<td>32.4</td>
<td>19.0</td>
</tr>
<tr>
<td>XLM-R [Conneau et al., 2020]</td>
<td>362M</td>
<td>34.2</td>
<td>21.4</td>
</tr>
<tr>
<td>LS-MNMT [Fan et al., 2020]</td>
<td>409M</td>
<td>34.8</td>
<td>21.1</td>
</tr>
<tr>
<td>HLT-MT (Our method)</td>
<td>381M</td>
<td>35.8</td>
<td>22.4</td>
</tr>
</tbody>
</table>

Figure 4: Average results of En→X high-resource directions (Fr, Cs, De, Fi, Lv, and Et) on the WMT-10 benchmark.

OPUS-100  In Table 2, we observe that HLT-MT achieves reasonable results on 94 translation directions. The improvement can be attributed to the high-resource training with the selective language-specific pool, which avoids the competition between high-resource and low-resource training directions. Another benefit of our approach is light and convenient to be applied to different backbone models since the parameters of the selective language-specific pool on the top of the decoder are tiny compared to all parameters.

5 Analysis

Size of Language-specific Parameters  The size of the selective language-specific pool depends on the two key factors, namely the hidden size $d_h$ and the number of modules $T$ described in Equation 4 and 3. We tune the different values of $d_h$ and $T$ in Figure 4(a) and 4(b) on the WMT-10 dataset. Naturally, the selective language-specific pool with a larger capacity leads to better performance. Increasing the number of the selective pool brings more improvement than the improvement of hidden size. Our method can efficiently reduce the language-specific parameters ($T = 3$) using the selection mechanism and get comparable results compared to the baseline, where each high-resource language has the independent language-specific layer ($T = 6$).

Ablation Study  In Table 3, we empirically validate our approach on the different backbone models including Transformer [Vaswani et al., 2017] without any pretrained model and XLM-T [Ma et al., 2020] initialized by the cross-lingual pretrained model XLM-R [Conneau et al., 2020]. High-resource training significantly helps improve the model performance but has trouble in effectively handling the low-resource translation directions merely by sharing all parameters, which is caused by the competition in the shared parameters between high-resource and low-resource languages. By introducing the selective language-specific pool (SLP) and extracting the bottom representations for low-resource languages, our approach ameliorates all translation directions.

Conflicting Gradient  To delve into the function of the language-specific module for multilingual training [Yu et al., 2020], we define $\Phi(L_a, L_b) = \frac{gL_a \cdot gL_b}{\|gL_a\|_2 \|gL_b\|_2}$ as the cosine similarity between two task gradients $gL_a$ and $gL_b$, where $gL_a$ and $gL_b$ separately denote the gradients of the En→L_a and En→L_b translation direction. $\Phi(L_a, L_b)$ determines whether $gL_a$ conflicts with $gL_b$ by computing the cosine similarity be-
Language-specific Selection Mechanism

The selective language-specific pool (SLP) of high-resource languages significantly contributes to translation quality. Equation 7 describes the selection of the module with the highest probabilities generated by the embedding of the target language symbol for the given translation direction. In Figure 6(a), \( \theta_1 \) is used for En, Fr, Cs, and Fi, \( \theta_2 \) is for De and Lv, and \( \theta_3 \) is for Et. The universal layer is used for all low-resource languages.

In Figure 6(b), we calculate the overlaps of dictionaries among multiple languages to measure the relationship of different languages. The language similarity between \( L_a \) and \( L_b \) is calculated by \( \text{Sim}(L_a, L_b) = \frac{||D_{L_a} \cap D_{L_b}||}{||D_{L_a} \cup D_{L_b}||} \), where \( D_{L_a} \) and \( D_{L_b} \) are the dictionary of language \( L_a \) and \( L_b \). Figure 6(b) shows that the language Et has the minimum overlap between other high-resource languages, where \( \theta_3 \) is only used for Et. Therefore, we conclude that similar languages tend to select the same language-specific module from SLP.

Figure 5: Cosine similarities between two gradients of training directions in (a) the baseline and (b) our method. Lower similarity (darker color) means higher negative interference.

Figure 6: (a) is the selection probabilities of different language-specific modules generated by the embedding of the target language symbol. The universal layer is only used for the low-resource languages. (b) is the overlapping ratio of the dictionaries between different languages. Lighter green means the higher selection probability of (a) and overlapping ratio of (b).

Between vectors \( g_{L_a} \) and \( g_{L_b} \), where the small value indicates conflicting gradients. Figure 5(b) and 5(a) show the similarities of the baseline and our method between different training directions. Different training tasks of our method have similar optimization, where \( \Phi(L_a, L_b) \) has larger value compared to the baseline scores. It corroborates that our language-specific training can effectively mitigate the conflicting gradients.

Decoder Representation Visualization

We randomly select 500 English sentences and visualize their representations [Maaten and Hinton, 2008] of the bottom decoder layers and the language-specific layer in Figure 7. The first hidden state of the decoder is regarded as the sentence representation. Compared to Figure 7(a), 7(b), and 7(c), different languages become more distinct and less likely to overlap with each other in Figure 7(d), proving that the selective language-specific pool (SLP) effectively projects the language-shared representations into language-distinct ones for better target generation of different target languages.

Figure 7: t-SNE visualization of the sentence representations from the bottom decoder layer (a), (b), (c), to the language-specific layer from SLP (d). Each color denotes one language.

6 Related Work

Multilingual neural machine translation (MNMT) [Johnson et al., 2017; Aharoni et al., 2019; Fan et al., 2020; Kong et al., 2021; Tang et al., 2021] enables numerous translation directions by shared encoder and decoder for all languages. The MNMT system can be categorized into one-to-many [Wang et al., 2018], many-to-one [Tan et al., 2019], and many-to-many [Pan et al., 2021] translation. Previous studies utilize assisting high-resource languages to improve low-resource or even zero-shot translation.

While MNMT is promising, it often underperforms bilingual baselines due to the interference in shared parameters, especially on high-resource pairs [Wang et al., 2020b]. To address this issue, language-specific modules are proposed to both enhance the low-resource translation and maintain the high-resource performance. Recent works mainly focus on designing language-specific components to boost the rich-resource translation quality [Vázquez et al., 2019; Philip et al., 2020; Gong et al., 2021]. Further works discuss when and where language-specific capacity matters most in MNMT [Escolano et al., 2021]. Our method finds a better balance between language-specific and language-agnostic models to mitigate negative interference.

7 Conclusion

In this work, we propose a novel multilingual translation model with the high-resource language-specific training called HLT-MT. The multilingual model is trained on multiple high-resource corpora with the selective language-specific pool, followed by continuing training on both high- and low-resource languages. Experimental results evaluated on WMT-10 and OPUS-100 benchmarks demonstrate that HLT-MT significantly outperforms all previous baselines.
References


