Exploiting General Purpose Sequence Representations for Low Resource Neural Machine Translation

H.Jeung Han*, Sathish Indurthi*, Sangha Kim
Samsung Research

*Equal contribution
Introduction

- **Neural Machine Translation**
  - Neural Machine Translation system has achieved near human level performance with abundant parallel corpus. (but... with small dataset...?)

- **Previous work on Low Resource NMT**
  - NMT with monolingual data, multilingual NMT, etc.
  - Transfer learning and meta-learning with auxiliary high resource parallel corpus and fine-tuned on low resource corpus.
    1. They assume the availability of several high-resource languages pairs for pre-training
    2. Most of them also assume English as the target language in all the language pairs -> difficult to apply non-English language translation.

- **General purpose sequence representations**
  - Led strong improvements in numerous NLP tasks.
  - Several attempts to utilize such representations into NMT system.
Introduction

- **NMTwGSR** (Neural Machine Translation with General Purpose Sequence Representations system)
  - In the proposed model, not explicitly training the encoder for source language using general purpose sequence representations.
  - The trainable parameters are only from the decoder and the output layer.
  - **Advantages**
    1. It doesn’t assume the availability of several high resource language pairs for pre-training as required by the previous approaches.
    2. The previously proposed techniques such as back-translation, multilingual-NMT, transfer/meta-learning training strategies are straight forward to integrated into the proposed system when high resource is available for the corresponding target pair.
    3. It can be easily adopted to a new language pair coming from low resource languages, whenever the sequence representations are available for these in language models.
  - Extensive evaluation on four low-resource translation task.
NMTwGSR

- **Model**
  - $\tilde{X} = BERT(x) \in \mathbb{R}^{m \times d}$.
  - $\tilde{Y} = BERT_{Masked}(y, mask) \in \mathbb{R}^{n \times d}$.
  - $\hat{Y} = \text{Transformer}_{decode}(\tilde{X}, \tilde{Y})$.
  - $p(w_t) = \text{softmax}(W_o \hat{y}_t + b_o)$

$W_o \in \mathbb{R}^{V \times d}$ and $b_o$ are learnable parameters.

$V$: output vocabulary size

$\hat{y}_t \in \hat{Y}$: output from decoder at time $t$. 

Experiments Settings

- **Dataset and Implementation details**
  - 4 language pair (Romanian, Finnish, Turkish, Latvian -> English) from WMT
  - Low resource environment simulation by 5 times random sampling of 160k, 320k, 640k, 1280k tokens
  - BERT-Base, Multilingual Cased model, weight fixed
  - tensorflow, OpenNMT

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Tokens (En)</th>
<th># Sentences</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lv-En</td>
<td>Fi-En</td>
<td>Ro-En</td>
<td>Tr-En</td>
</tr>
<tr>
<td>Train</td>
<td>Full</td>
<td>4.46M</td>
<td>2.63M</td>
<td>0.61M</td>
<td>0.21M</td>
</tr>
<tr>
<td></td>
<td>1280K</td>
<td>99.1K</td>
<td>58.8K</td>
<td>55.4K</td>
<td>60K</td>
</tr>
<tr>
<td></td>
<td>640K</td>
<td>49.8K</td>
<td>29.4K</td>
<td>27.7K</td>
<td>30K</td>
</tr>
<tr>
<td></td>
<td>320K</td>
<td>24.7K</td>
<td>14.7K</td>
<td>13.9K</td>
<td>15K</td>
</tr>
<tr>
<td></td>
<td>160K</td>
<td>12.4K</td>
<td>7.3K</td>
<td>7K</td>
<td>7.5K</td>
</tr>
<tr>
<td>Dev</td>
<td>-</td>
<td>2K</td>
<td>3K</td>
<td>2K</td>
<td>3K</td>
</tr>
<tr>
<td>Test</td>
<td>-</td>
<td>2K</td>
<td>3K</td>
<td>2K</td>
<td>3K</td>
</tr>
</tbody>
</table>
Experimental Results

- **Training in Low Resource Environment**
  - BLEU score of model trained on sampled subset on 4 languages in Table- Forward and Reverse direction

<table>
<thead>
<tr>
<th>#Tok</th>
<th>Ro-En Base</th>
<th>Ours</th>
<th>En-Ro Base</th>
<th>Ours</th>
<th>Fi-En Base</th>
<th>Ours</th>
<th>En-Fi Base</th>
<th>Ours</th>
<th>Tr-En Base</th>
<th>Ours</th>
<th>En-Tr Base</th>
<th>Ours</th>
<th>Lv-En Base</th>
<th>Ours</th>
<th>En-Lv Base</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>31.76</td>
<td>32.68</td>
<td>-</td>
<td>-</td>
<td>20.20</td>
<td>22.39</td>
<td>-</td>
<td>-</td>
<td>13.74</td>
<td>15.19</td>
<td>-</td>
<td>-</td>
<td>15.15</td>
<td>17.39</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>640K</td>
<td>14.14</td>
<td>17.68</td>
<td>12.70</td>
<td>16.38</td>
<td>5.53</td>
<td>7.62</td>
<td>3.67</td>
<td>4.58</td>
<td>5.08</td>
<td>7.76</td>
<td>4.58</td>
<td>6.43</td>
<td>3.80</td>
<td>5.22</td>
<td>3.01</td>
<td>4.30</td>
</tr>
<tr>
<td>320K</td>
<td>8.25</td>
<td>14.23</td>
<td>6.63</td>
<td>11.01</td>
<td>3.24</td>
<td>5.47</td>
<td>1.98</td>
<td>2.92</td>
<td>2.59</td>
<td>5.44</td>
<td>1.74</td>
<td>3.91</td>
<td>2.29</td>
<td>3.52</td>
<td>1.71</td>
<td>2.76</td>
</tr>
<tr>
<td>160K</td>
<td>4.80</td>
<td>10.48</td>
<td>4.20</td>
<td>7.97</td>
<td>1.04</td>
<td>3.12</td>
<td>0.53</td>
<td>1.91</td>
<td>1.44</td>
<td>3.49</td>
<td>1.02</td>
<td>2.15</td>
<td>0.85</td>
<td>2.15</td>
<td>0.96</td>
<td>1.72</td>
</tr>
</tbody>
</table>

Table 1: Test BLEU results of models trained on 160k, 320k, 640k, and 1280k sampled subsets of four languages parsed both for forward and backward. The models trained on Full dataset is only presented with forward direction.
Experimental Results

- Training in Low Resource Environment
  - Forward direction
  - Reverse direction
Experimental Results

- **Learning curves**
  - The learning curves of BLEU scores for Lv-En and En-Lv pairs
  - In each plot, the upper learning curves are obtained using 1280k subset and lower learning curves are obtained using 160k subset.
Thank you