Faster Transformers for text summarization
Alexandre Matton, Amaury Sabran
Stanford University

Introduction
The transformer corresponds to an encoder/decoder architecture. In our case it takes as input a text and outputs its summary.

Models

Local Transformer:
The local transformer divides the input sequence into chunks of fixed-size which are processed independently by the encoder.

Complexity: \( O(n \times k \times d) \)

Local Transformer with shifts:
One major problem of the local transformer is that it prevents information flow from one chunk to another. We implemented a fix to this issue by shifting all chunks by half of their size in odd layers of the encoder.

Complexity: \( O(n \times k \times d) \)

Lightweight Convolutions [2]:
This model replaces self-attention layers by some kind of local convolutions where each filter only takes into account one dimension, via a matrix \( W \in \mathbb{R}^{d \times k} \) where \( k \) is the size of the convolution window.

\[
O_{i,i'} = \sum_{j=1}^{k} W_{i,j} \cdot X_{i+j-\lceil \frac{k}{2} \rceil,i'}
\]

where \( X \in \mathbb{R}^{n \times d} \) is the input and \( O \in \mathbb{R}^{n \times d} \) is the output. \( W \) is the matrix \( W \) with a softmax layer applied across each channel.

Complexity: \( O(n \times k \times d) \)

Convolution before Transformer:
We reduce the size of the inputs by applying strided convolutions on them before feeding them to the Transformer. From a high-level perspective, the transformer processes the summarized inputs.

Complexity: \( O(n \times d^2 + (\frac{d}{2})^2 \times d) \)

Memory-compressed attention [3]:
This architecture also uses strided convolutions to decrease the size of the inputs. However, the convolutions are located in the self-attention layers. The memory compressed module is described as follows:

\[
MC_{\text{Att}}(Q, K, V) = \text{softmax}(\frac{Q \times c_d (K)^T}{\sqrt{d}}) c_d (V)
\]

Complexity: \( O(n \times d^2 + \frac{d}{2} \times d) \)

ROUGE Scores
The ROUGE metrics is commonly used in text summarization. It compares the produced summaries with humanly-written summaries, taking into account precision and recall.

Our models are based on small architectures. The results with the full ones for Transformer and Lightweight convolutions are also given.

Table: ROUGE scores and speedups for our models

<table>
<thead>
<tr>
<th>Model</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transformer</td>
<td>32.39</td>
<td>8.78</td>
<td>26.8</td>
<td>1</td>
</tr>
<tr>
<td>+ Input conv.</td>
<td>30.30</td>
<td>8.63</td>
<td>26.05</td>
<td>1.62</td>
</tr>
<tr>
<td>LightConv</td>
<td>36.27</td>
<td>14.31</td>
<td>30.91</td>
<td>1.08</td>
</tr>
<tr>
<td>Local Transf.</td>
<td>35.53</td>
<td>14.01</td>
<td>30.62</td>
<td>1.13</td>
</tr>
<tr>
<td>+ Shift</td>
<td>35.8</td>
<td>14.54</td>
<td>30.92</td>
<td>1.13</td>
</tr>
<tr>
<td>MC Att</td>
<td>31.43</td>
<td>7.70</td>
<td>26.12</td>
<td>1.01</td>
</tr>
<tr>
<td>Full LightConv</td>
<td>38.37</td>
<td>16.20</td>
<td>32.7</td>
<td></td>
</tr>
<tr>
<td>Full Transformer</td>
<td>25.55</td>
<td>5.08</td>
<td>22.5</td>
<td></td>
</tr>
</tbody>
</table>

Speed curves

Conclusion
• Encoder self-attention is the main cost when length of input > 1500.
• Models that focus on extracting information at a local level outperform the Transformer.
• Hence, Lightweight convolutions model and our local transformer model are most suited to Text Summarization.

References