Faster Transformers for text summarization

Alexandre Matton, Amaury Sabran

Stanford University

Objectives

In this project, we explored a few models based on the **Transformer** [1], a recent seq2seq architecture relying exclusively on attention. Our goal is to speed it up while sacrificing as little accuracy as possible. We apply the transformer architecture to Text Summarization since this task involves long input texts.

Introduction

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

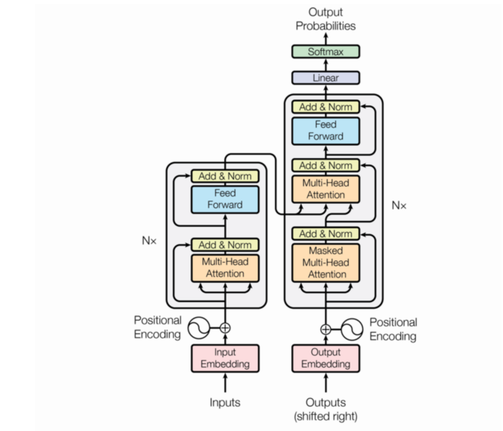

Figure: Transformer Architecture

Attention layers are the core blocks of the transformer. They change the embeddings of each token by taking into account the rest of the input tokens, according to the formula:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V$$

The attention layer in the encoder is **quadratic** in the size of the text, whereas all the other blocks of the model are at most linear in the size of the text. For inputs with size 400, this attention layer takes **24%** of the total time of execution, while it takes up to **64%** for inputs with size 2000.

Models

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

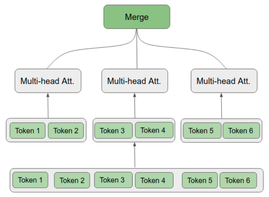

Figure: Local Attention

Complexity: $\mathcal{O}(n \times k \times d)$

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: $\mathcal{O}(n \times k \times d)$

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.

$$O_{i,c} = \sum_{j=1}^{k} W'_{c,j} \cdot X_{(i+j-\lceil \frac{k+1}{2} \rceil),c}$$

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: $\mathcal{O}(n \times k \times d)$.

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 convolution summarizes small contiguous groups of words (typically 4) and the transformer processes the summarized inputs.
Complexity: $\mathcal{O}(n \times d^2 + (\frac{n}{k})^2 \times d)$

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:

$$MC\_Att(Q, K, V) = softmax(\frac{Q * c_1(K)^T}{\sqrt{d}})c_2(V)$$

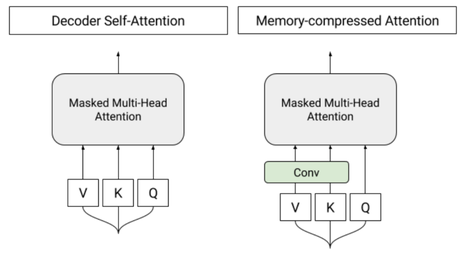

Figure: Self-Attention vs. Memory compressed attention

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

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.

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

Table: ROUGE scores and speedups for our models

The CNN/DailyMail dataset

It consists of over 280K news articles paired with multi-sentence summaries. The articles are rather long with 39 sentences on average. During training and testing we truncate the articles to 400 tokens.

Speed curves

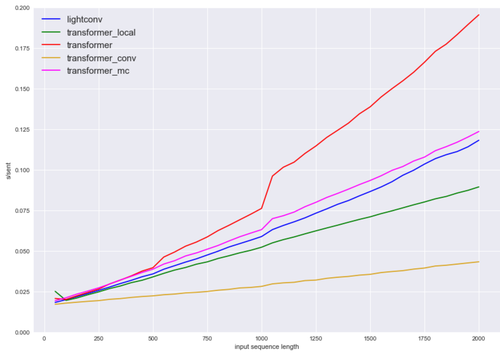

Figure: Time per sentence for each model (s)

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

[1] Vaswani, Ashish, et al. "Attention is all you need." Advances in Neural Information Processing Systems. 2017.

[2] Wu, Felix, et al. "Pay Less Attention with Lightweight and Dynamic Convolutions." arXiv preprint arXiv:1901.10430 (2019).

[3] Liu, Peter J., et al. "Generating wikipedia by summarizing long sequences." arXiv preprint arXiv:1801.10198 (2018).