Long Document Summarization Using a Fourier Transform Based Attention Mechanism in a Transformer Model (FNET)

Anonymous ACL submission

Abstract

This paper attempts to use the newly developed Fourier Transforms attention based transformer model to summarize long documents (>512 tokens) which are computationally challenging with the original self-attention mechanism in the original transformer model. As baseline, we also carry out long document summarization using established methods such as breaking up the long document into chunks followed by sequential summarization of each reduced segment as well as the longformer transformer model that is capable of processing over 8000 tokens and is currently the gold standard for these type of problems. This provides us with an improvement over the segmentation based approach and allows us to compare the proposed new approach that replaces the computationally expensive self-attention mechanism in the transformer with a Fourier transform module. The original paper implements this in an encoder only approach while abstractive summarization requires both an encoder and a decoder. We implement the whole Fourier based approach in an encoder/decoder architecture which we train starting with Glove embeddings for the latent dimension.

1 Introduction

Abstractive summarization has made significant strides since the introduction of the Transformer based model in NLP. However, the quadratic computational and memory complexities of large Transformers have limited their scalability for long document summarization as the token length for a standard Transformer is limited to 512 tokens. One can try extractive summarization to reduce the length of the document into chunks followed by an abstractive approach on the reduced document. In the extractive step, the top k sentences are chosen per chunk to reduce the size of the total count to the token limits of the Transformer model.

Another way is to use successive abstractive summarization to summarize the document into chunks in iterative fashion. This is computationally very expensive. We will try the chunk extractive method described above followed by an abstractive summarization of the individual chunks as a our baseline model.

More recently, ˜(**?**) a group at Google has introduced a new implementation that replaces the self-attention module in Transformers with a Fourier transform that does not suffer from this quadratic computation penalty. We propose to extend this architecture to the long document summarization problem and compare the results to two current baseline practices: Breaking the document into chunks and applying summarization on each segment or using a longformer implementation. We investigated multiple hyperparameter optimization of both these approaches on a Pubmed dataset and score the summaries relative to their corresponding abstracts. This becomes the method for comparing the performance of each approach.

2 Baselines

2.1 Text Chunking Summarization

Modern long document summarization must account for the quadratic computation complexities of transformer architectures. The standard limit for these architectures is 512 tokens. PEGASUS (https://arxiv.org/pdf/1912.08777v2.pdf) is a pretrained model transformer that was created for abstractive text summarization. To utilize PEGASUS on long documents we break up our data into chunks of less than 512 tokens. To ensure that each chunk contains whole ideas we ensure each chunk contains full paragraphs. If paragraphs are larger than 512 tokens we ensure that each chunk contains full sentences. Once the data is chunked we can run each chunk through

PEGASUS and concatenate the results to compare with our human generated summary. We are also attempting to preprocess our chunks using BERT (https://arxiv.org/pdf/1810.04805.pdf) extractive text summarization before summarizing with PEGASUS. This will increase the performance of PEGASUS by only applying abstractive summarization on the key text elements selected by the BERT model. Chunking allows us to break up documents into more manageable pieces for processing. The BERT extractive summary highlights only the important features of the documents then finally we can run our abstractive summarization to and score the results against human generated summaries.

2.2 Longformer Baseline

Longformer was first introduced by Allen AI in the paper Longformer: The Long-Document Transformer. The idea behind the approach is to remove the quadratic dependency on sequence length in the self-attention layer. The approach instead uses an attention operation that scales linearly with the sequence length. This is achieved by using alternatives to the full attention architecture. These alternative approaches are: Sliding window attention, Dilated window attention, and Global plus Sliding window. These methodologies allow the expensive quadratic term QK^T from (Vaswani et al., 2017) to be replaced with a term that computes only a fixed number of diagonals of QK^T, using the dilated sliding window attention. Allen AI's paper does not introduce this methodology for summarization tasks. However, longformer has since been used in a number of summarization tasks, including examples done on the PubMed dataset. The approach basically consists of fine tuning the smaller LED checkpoint "allenai/led-base-16384" (https://huggingface.co/allenai/led-base-16384). The dataset is first tokenized and then, in addition to the attention mask, we make use of the global attention mask, as is the case in longformer. Following the suggestions from (https://arxiv.org/pdf/2004.05150.pdf: Longformer: The Long-Document Transformer) we only use global attention for the very first token. For this milestone, we trained the model for three epochs only.

2.3 Baseline results

We decided to use Rouge2 as a metric for evaluation, as it is one of the most commonly used metric.

Rouge-2 Score	PEGASUS	Longformer
Precision	0.06	0.175
Recall	0.05	0.109
F-score	0.05	0.127

Table 1: Baseline model results for Long Document Summarization

3 Datasets

The primary dataset used for this paper is the PubMed dataset. This dataset has a median token length of 2,715 with the 90^{th} percentile token length being 6,101. Cite: https://arxiv.org/pdf/2007.14062.pdf (Big Bird: Transformers for Longer Sequences)

4 Fourier Transform Based Attention (FNET)

Transformer architectures have come to dominate the natural language processing (NLP) field since their 2017 introduction. One of the only limitations to transformer application is the huge computational overhead of its key component — a self-attention mechanism that scales with quadratic complexity with regard to sequence length.

New research from a Google team proposes replacing the self-attention sub-layers with simple linear transformations that "mix" input tokens to significantly speed up the transformer encoder with limited accuracy cost. Even more surprisingly, the team discovers that replacing the self-attention sub-layer with a standard, unparameterized Fourier Transform achieves 92 percent of the accuracy of BERT on the GLUE benchmark, with training times that are seven times faster on GPUs and twice as fast on TPUs.

The Fourier transform is a mathematical operation that transforms a complex temporal or spatial signal, into simpler sub-components defined by a frequency spectrum. This operation can be applied to a sentence inside the Transformer architecture as a replacement of the self-attention mechanism. Intuitively, applying a Fourier transform is just encoding the input as a linear combination of the text embeddings. These linear combinations are mixed with simple non-linearities in the feed-forward network. The result is a faster Transformer that is able to competitively understand the semantic relationship in the text in several NLP tasks. Google researchers proposed replacing the self-attention sub-layers with simple linear transformations that

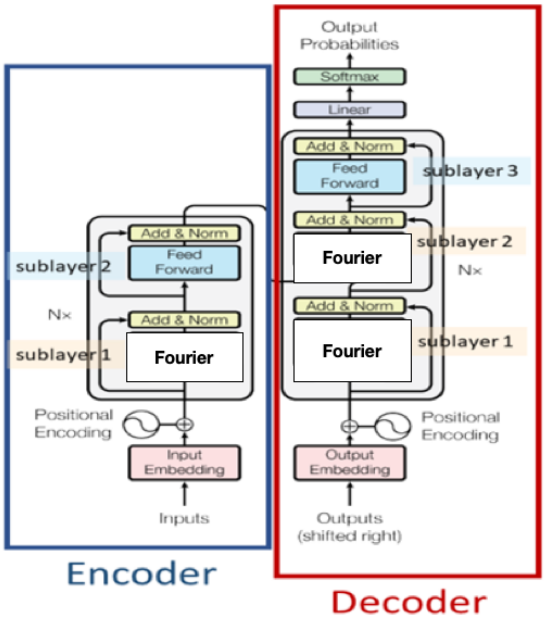

Figure 1: Original Google FNET paper is implemented with a Fourier module replacing the self attention mechanism. In this work, we have completed the decoder architecture with a similar replacement of the self-attention mechanism as shown in the figure. To our knowledge, this is the first full implementation of the transformer model using only Fourier transformations for the attention mechanism.

"mix" input tokens. This approach speeds up the Transformer encoder architectures while keeping the accuracy costs limited. Additionally, the complexity and memory footprint of the Transformer architecture is reduced. However, all this was done only on the encoder side. In this work, we need the entire transformer for long document summarization and hence we extended the idea by replacing the self attention blocks in the decoder and re-implementing the entire decoder from scratch as shown in the red block in the figure. As the Google researchers pointed out in their paper, "designing the equivalent of encoder-decoder cross-attention remains an avenue for future work", which is exactly the main contribution of this work to the long document summarization problem.

5 Next Steps

1. Run additional epochs using the PubMed dataset for the PEGASUS/BERT baseline

2. Explore alternative ways of segmenting long documents for summarization with PEGASUS such as Divide-and-Conquer methods.

3. Increase the number of articles to train and validate the Longformer baseline

4. Explore additional hyperparameter tuning for the Longformer baseline

5. Complete coding of the decoder component of the transformer model from scratch by replacing the self-attention with a Fourier model and combine with the encoder as proposed by Google team.

6. Train FNET transformer with PubMed articles as sources with the abstract as target in a downstream summarization task.

7. Generate Rouge scores with the FNET transformer concept on validation dataset and compare to the longformer and baseline segmentation approach used in the long document summarization. task.

6 References

Allen Institute for Artificial Intelligence. 2020. *Longformer: The Long-Document Transformer.*

Google Research. 2021. *Big Bird: Transformers for Longer Sequences.*

Jingqing Zhang. 2020. *PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization.*

Alexios Gidiotis. 2020. *A Divide-and-Conquer Approach to the Summarization of Long Documents.*

Google Research. 2021. *Big Bird: Transformers for Longer Sequences.*