Longformer: The Long-Document Transformer

Beltagy et al., 2020

Presented by Leslie Zhou

Background

- **Transformers:** have achieved state-of-the-art results in a wide range of natural language tasks including generative language modeling and discriminative language understanding.
- **BUT** infeasible (or very expensive) to process long sequences on current hardware

Longformers: The Long-Document Transformer

- **Longformers:** a modified Transformer architecture with a self-attention operation that scales linearly with the sequence length, making it versatile for processing long documents.

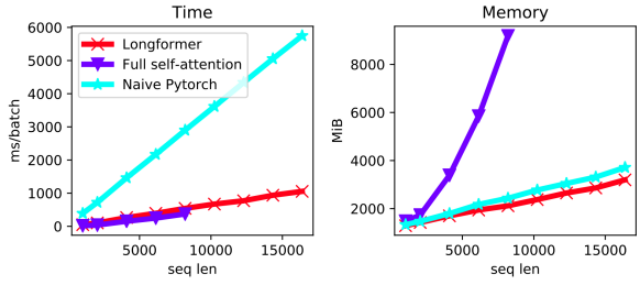
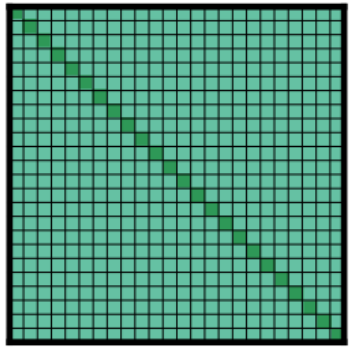
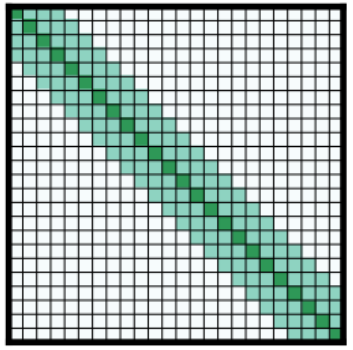


Figure 1: Longformer's memory usage scales linearly with the sequence length, unlike the full self-attention mechanism that runs out of memory for long sequences on current GPUs. Longformer's GPU-kernel is nearly as fast as the highly optimized full self-attention operation, and nearly 6X faster than naive Pytorch.

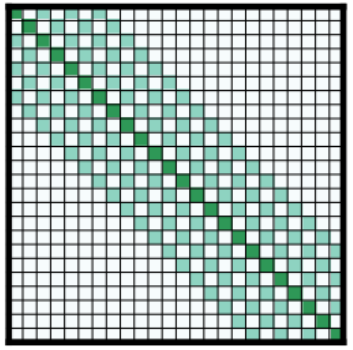
Models (Attention Patterns)



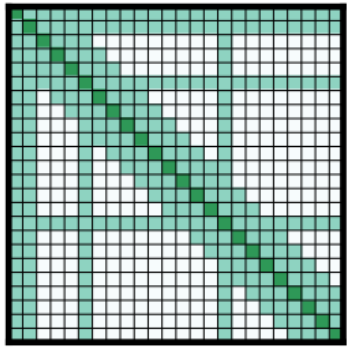
(a) Full n^2 attention



(b) Sliding window attention



(c) Dilated sliding window

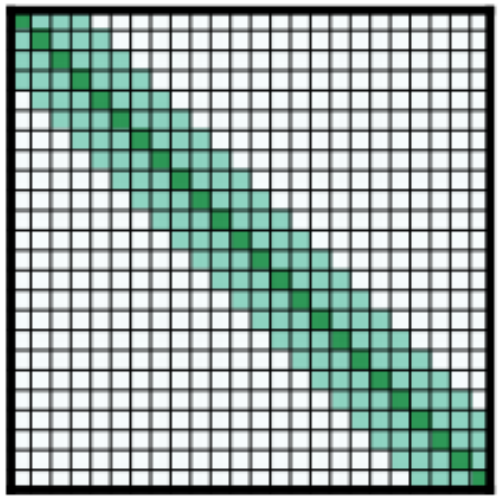


(d) Global+sliding window

Abstract

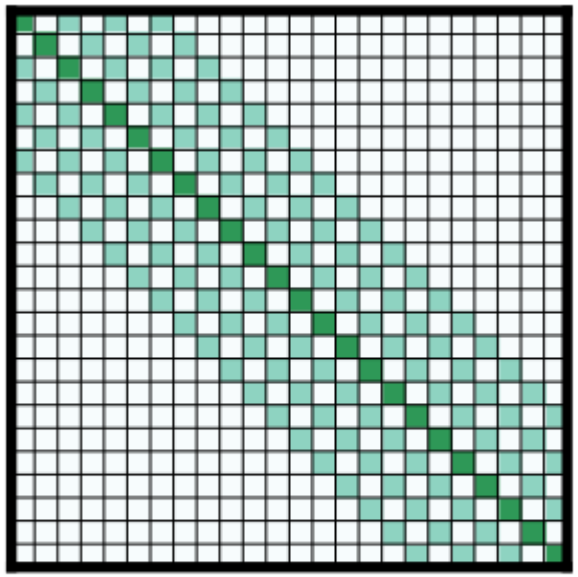
Transformer-based models are unable to process long sequences due to their self-attention operation, which scales quadratically with the sequence length. To address this limitation, we introduce the Longformer with an attention mechanism that scales linearly with sequence length, making it easy to process documents of thousands of tokens or longer. Longformer's attention mechanism is a drop-in replacement for the standard self-attention and combines a local windowed attention with a task motivated global attention. Following prior work on long-sequence transformers, we evaluate Longformer on character-level language modeling and achieve state-of-the-art results on `text8` and `enwik8`. In contrast to most prior work, we also pretrain Longformer and finetune it on a variety of downstream tasks. Our pretrained Longformer consistently outperforms RoBERTa on long document tasks and sets new state-of-the-art results on WikiHop and TriviaQA.[1]

Sliding Window



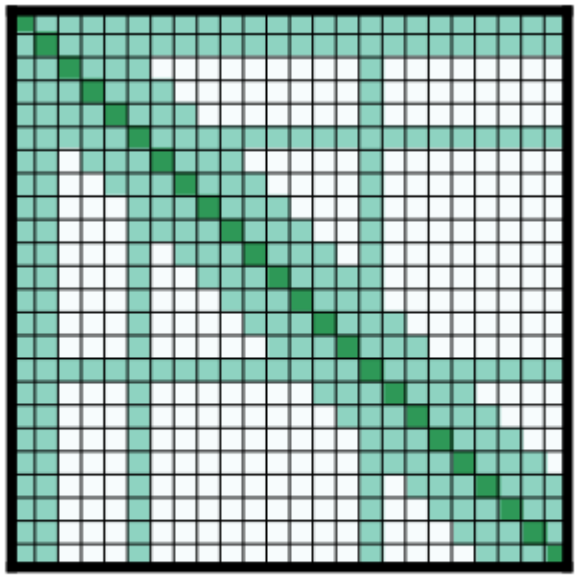
(b) Sliding window attention

Dilated Sliding Window



(c) Dilated sliding window

Global Attention



(d) Global+sliding window

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

CUDA Kernels

- The dilated sliding window attention pattern is not straightforward to implement using modern deep learning libraries
- Custom CUDA Kernel
- Tensor Virtual Machine (TVM) (Chen et al., 2018)

Autoregressive Language Modeling

- Left-to-right language modeling
- Estimates the probability distribution of an existing token/character given its previous tokens/characters in an input sequence.

Pretraining

Source	Tokens	Avg doc len
Books (Zhu et al., 2015)	0.5B	95.9K
English Wikipedia	2.1B	506
Realnews (Zellers et al., 2019)	1.8B	1.7K
Stories (Trinh and Le, 2018)	2.1B	7.8K

Table 5: Pretraining data

- Can process sequences up to 4096 tokens long
- Pretrain with masked language modeling (MLM)
- Uses sliding window attention with window size of 512 on all layers (matches RoBERTa's sequence length)
- Mix of long and short documents to both allow the model to learn longer dependencies while not to forget information from the original RoBERTa pretraining.

Tasks

- Apply Longformer to multiple long document tasks
  - Question Answering (WikiHop, Wikipedia setting of TriviaQA , and HotpotQA)
  - Coreference Resolution (OntoNotes and the model from Joshi et al. (2019))
  - Classification (IMDB and Hyperpartisan news detection datasets.1)

Result

	QA			Coref.	Classification	
Model	WikiHop	TriviaQA	HotpotQA	OntoNotes	IMDB	Hyperpartisan
RoBERTa-base	72.4	74.3	63.5	78.4	95.3	87.4
Longformer-base	**75.0**	**75.2**	**64.4**	**78.6**	**95.7**	**94.8**

Table 8: Summary of finetuning results on QA, coreference resolution, and document classification. Results are on the development sets comparing our Longformer-base with RoBERTa-base. TriviaQA, Hyperpartisan metrics are F1, WikiHop and IMDB use accuracy, HotpotQa is joint F1, OntoNotes is average F1.

Conclusion

Longformer: a transformer-based model that is scalable for processing long documents

-Easy to perform a wide range of document-level NLP tasks without chunking/shortening the long input

-No complex architecture to combine information across these chunks

-Combines local and global information while also scaling linearly with the sequence length

-Outperforms RoBERTa on long document tasks

Thanks!

Questions?