Conv-Linformer: Boosting Linformer's Performance with Convolution in Small-Scale Settings

Lenny Malard
Angers, France
lennymalard@gmail.com

Transformers (Vaswani et al. [2017]) capture token dependencies effectively, but suffer from $O(n^2)$ complexity. Linformer (Wang et al. [2020]) reduces this to $O(kn)$ via a low-rank projection while aiming to preserve Transformer performance, but its behavior under resource constraints remains underexplored. We reproduce Linformer on a 50M-token masked-language-modeling (MLM) task, evaluating its ability to maintain linear complexity without degrading performance. Our results show that high learning rates benefit short sequences but destabilize longer ones, while lower learning rates improve stability at some performance cost. To address this, we propose Conv-Linformer, which integrates linear projection with 1D convolution for better local feature extraction, achieving Linformer's efficiency with performance comparable to Transformers.

1 Methodology

1.1 Linformer Overview

Linformer is based on the observation that the rank of self-attention tends to decrease with depth, suggesting that attention information can be compressed. By projecting the keys and values from an $n \times d_m$ space to a $k \times d_m$ space (with $k \ll n$), the resulting attention matrix is reduced from size $n \times n$ to size $n \times k$. This approach achieves linear scaling with respect to the sequence length, providing significant efficiency gains in both time and memory for long sequences.

$$\text{Attention}(Q, K, V) = \underbrace{\text{softmax}\left(\frac{Q(EK)^T}{\sqrt{d_k}}\right)}_{n \times k} \cdot \underbrace{FV}_{k \times d_v}, \tag{1}$$

where E and F are matrices that project K and V to dimension k.

1.2 Convolutional Approach

We propose a hybrid architecture in which the early layers use the original linear projections to capture long-range dependencies, while the later layers use 1D convolution to extract local patterns. By using a kernel size and stride of n/k, we effectively downsample the sequence to match the reduced dimension k. This design takes advantage of both global and local feature extraction, making it particularly effective for MLM.

1.3 Pretraining Setup

Datasets We use the WikiText-103 (Merity et al. [2016]) dataset, which offers high-quality Wikipedia articles, to construct four subsets of 50 million tokens each. These subsets are prepared with varying sequence lengths (128, 256, 512 and 1024) and processed for MLM.

Models All models use an encoder-only architecture with eight layers and token embeddings of dimension 512 across eight attention heads.

Training Configurations We train all models using the Hugging Face Trainer framework with PyTorch in the backend. Optimization is performed using AdamW, with a weight decay of 0.001 and a warmup ratio of 10%.

2 Results

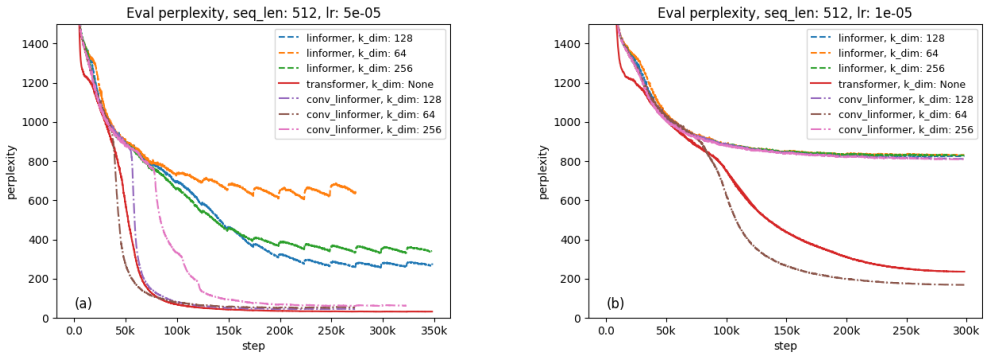

Figure 1: Pretraining validation perplexity with respect to the number of steps.

We found that a high learning rate destabilizes training for longer sequences, whereas reducing it improves stability at the cost of performance. In Figure 1a, Linformer exhibits instability, as its perplexity struggles to converge, while Conv-Linformer remains stable and performs near Transformer-level results. In Figure 1b, with a lower learning rate, both architectures converge more consistently, but Conv-Linformer escapes the common local minima in one configuration, even surpassing Transformer. For smaller sequences (not shown), Conv-Linformer consistently outperforms Linformer across all configurations, achieving performance on par with Transformer, despite Linformer showing better performance at shorter sequence lengths.

3 Conclusion

Our evaluation of Linformer revealed significant limitations, as it did not achieve the expected Transformer-level performance, particularly with longer sequences. This gap from the original work is likely due to the limited amount of training data. With its low-rank approximations, Linformer may require more data to compensate for information loss. In contrast, Conv-Linformer exhibited greater stability and near-Transformer performance while preserving efficiency, surely due to its ability to better extract local information. Future work should explore these architectures on larger datasets and evaluate their effectiveness in downstream tasks.

References

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017. URL `https://papers.nips.cc/paper/7181-attention-is-all-you-need`.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *ArXiv*, abs/2006.04768, 2020. URL `https://arxiv.org/abs/2006.04768`.