COSFORMER : RETHINKING SOFTMAX IN ATTENTION
Reading Report

Tianyu Shi

March 12, 2022

1 Introduction

Transformer has achieved great success in natural language processing and computer vision. As one of its core components, softmax helps to capture long-range dependencies, but prohibits its extension due to the quadratic space and time complexity of sequence length. Many researchers have tried to replace softmax with other methods to reduce the complexity. However, an efficient and accurate approximation of softmax is difficult to achieve. Therefore, can we replace softmax with a linear function while maintaining its performance?

2 Method

In this paper, researchers propose a linear Transformer, called COSFORMER, that satisfies both properties affecting softmax. (i) the nonnegativity of the attention matrix. (ii) A nonlinear reweighting scheme that aggregates the attention matrix distribution. COSFORMER, which completely discards softmax normalization, has both nonnegativity and a reweighting mechanism. COSFORMER contains two main components, the linear projection kernel ϕ_{linear} and the cos-Based reweighting mechanism.

First, the linear projection kernel is really important. The researcher defines linear similarity as shown below.

$$S(Q, K) = s(\phi_{linear}(Q), \phi_{linear}(K)) = S(Q', K')$$

Specifically, to ensure a full positive attention matrix A and to avoid aggregating negatively correlated information, the researcher uses ReLU as a transformation function, which effectively eliminates negative values, as shown below.

$$\phi_{linear}(x) = ReLU(x)$$

The researchers also reorganized the order of dot-product and obtained the formula for the attention proposed in the linear complexity.

$$O_i = \frac{ReLU(Q_i) \sum_{j=1}^{N} ReLU(K_j)^T V_j}{ReLU(Q_i) \sum_{j=1}^{N} ReLU(K_j)^T}$$

The nonlinear reweighting mechanism can aggregate the distribution of attention weights and therefore stabilize the training process. The researchers propose a cos-based reweighting mechanism that perfectly satisfies the objective and enables COSFORMER to perform more localization than COSFORMER without the reweighting mechanism. The model with a cosine reweighting mechanism can be defined as shown below.

$$S(Q'_i, K'_i) = Q'_i K'^T_i cos(\frac{\pi}{2} \times \frac{i - j}{M})$$

3 Experiment

First, the study verified the ability of COSFORMER in language modeling using WikiText-103, with autoregressive and bidirectional language models. They replaced the self-attention module with the proposed linear attention module. From the results, it is concluded that although the baseline model is a powerful performance standard transformer, it requires a quadratic computational complexity, but COSFORMER significantly outperforms the baseline model in terms of linear computational complexity.

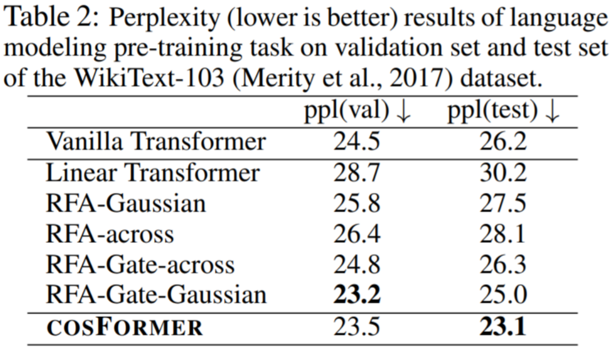

Table 2: Perplexity (lower is better) results of language modeling pre-training task on validation set and test set of the WikiText-103 (Merity et al., 2017) dataset.

	ppl(val) ↓	ppl(test) ↓
Vanilla Transformer	24.5	26.2
Linear Transformer	28.7	30.2
RFA-Gaussian	25.8	27.5
RFA-across	26.4	28.1
RFA-Gate-across	24.8	26.3
RFA-Gate-Gaussian	**23.2**	25.0
COSFORMER	23.5	**23.1**

For bidirectional language modeling, the study used RoBERTa as the baseline model. The same replaces the self-attention module in RoBERTa with a linear attention module and keeps the other structures unchanged. As shown in Figure 4, COSFORMER converges faster than vanilla Transformer on both the training and validation sets.

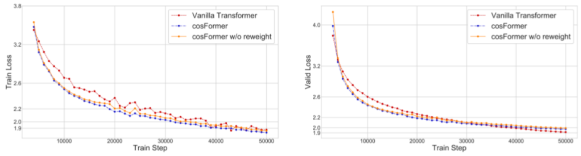

Figure 4: Training loss (left) and validation loss (right) of the bidirectional language modeling pre-train. In both training and validation, the proposed COSFORMER has a faster converge speed than vanilla transformer.

In addition, this study investigated the generalization ability of COSFORMER on downstream tasks by comparing it with other existing Transformer variants. From Table 3, we can see that COSFORMER outperforms the baseline on several datasets. Moreover, COSFORMER achieves the best or second-best position on all five downstream datasets compared to other efficient transformers.

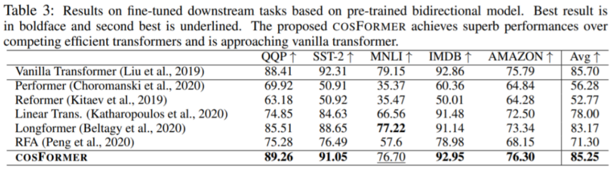

Table 3: Results on fine-tuned downstream tasks based on pre-trained bidirectional model. Best result is in boldface and second best is underlined. The proposed COSFORMER achieves superb performances over competing efficient transformers and is approaching vanilla transformer.

	QQP ↑	SST-2 ↑	MNLI ↑	IMDB ↑	AMAZON ↑	Avg ↑
Vanilla Transformer (Liu et al., 2019)	88.41	92.31	79.15	92.86	75.79	85.70
Performer (Choromanski et al., 2020)	69.92	50.91	35.37	60.36	64.84	56.28
Reformer (Kitaev et al., 2019)	63.18	50.92	35.47	50.01	64.28	52.77
Linear Trans. (Katharopoulos et al., 2020)	74.85	84.63	66.56	91.48	72.50	78.00
Longformer (Beltagy et al., 2020)	85.51	88.65	**77.22**	91.14	73.34	83.17
RFA (Peng et al., 2020)	75.28	76.49	57.6	78.98	68.15	71.30
COSFORMER	**89.26**	**91.05**	76.70	**92.95**	**76.30**	**85.25**

4 Conclution

Researchers have proposed a new replacement for softmax that not only achieves comparable or even better performance than softmax in a range of tasks, but also has linear space and time complexity. However, in some tasks, COSFORMER still does not perform as well as Transformer. Therefore, maybe how to use the cosine function better is the key to continue to optimize the design of linear Transformer.

References

[1] Zhen Qin, Weixuan Sun, Hui Deng, Dongxu Li, Yunshen Wei, Baohong Lv, Junjie Yan, Lingpeng Kong, Yiran Zhong. *COSFORMER : RETHINKING SOFTMAX IN ATTENTION*, ICLR, 2022.