Refiner: Refining Self-attention for Vision Transformers

1、Motivation

许多近期针对ViT的工作集中于设计更复杂的结构或者设计训练策略来提升ViT的性能，而很少关注基础的self-attention结构。

传统的SA中，每个token都要和其他所有token计算相似度并根据相似度将values叠加，这样每个token可以充分交换信息。但带来的问题是可能会导致tokens变得越来越像，导致ViT效果不够好。因此，作者希望针对这一问题，提出相应的解决方案。

作者想了两个解决方案，1. 增加SA的头数，增强attention map的离散程度同时引入attention expansion保证送入每个head的embedding足够；2. 引入局部信息，让token可以对附近位置的token施加更多关注

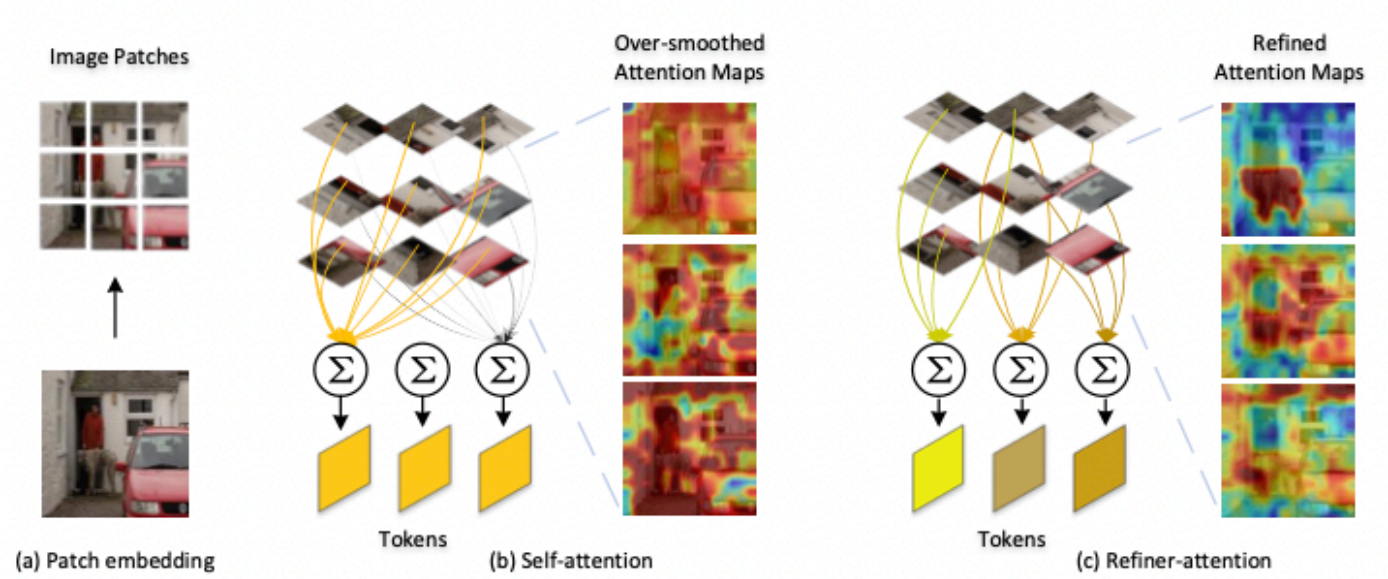

Figure 1: Illustration on our motivation. (a) The input image is regularly partitioned into patches for patch embedding. (b) The token-wise attention maps from vanilla self-attention of ViTs tend to be uniform, and thus they aggregate all the patch embeddings densely and generate overly-similar tokens. (c) Differently, our proposed refiner augments the attention maps into diverse ones with enhanced local patterns, such that they aggregate the token features more selectively and the resulting tokens are distinguishable from each other.

虽然思路简单，但是效果看起来还不错。

2、Refiner

2.1 Limitations of ViTs

作者认为缺少归纳偏置的原始ViTs的feature演进速度比CNN慢，并用实验证明了这一点。

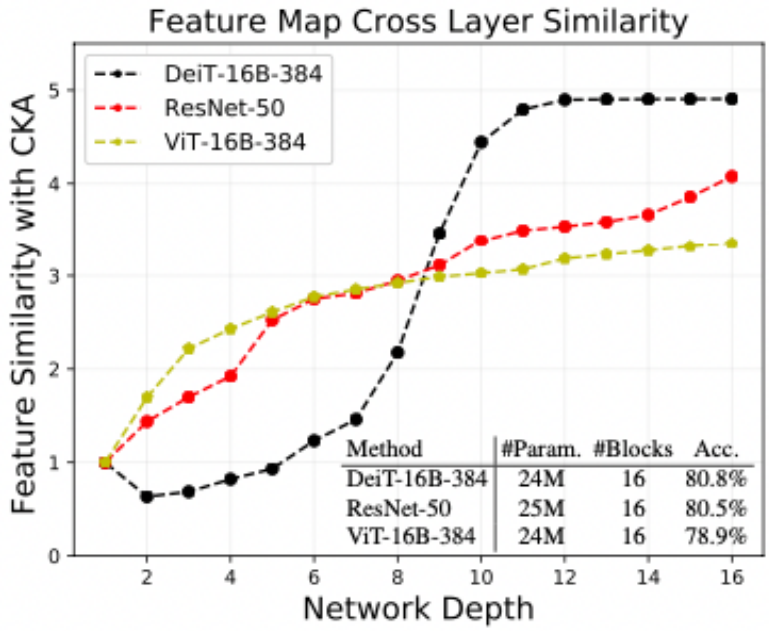

Figure 2: The features of ViT evolves slower than ResNet [25] and DeiT [48] across the model blocks.

作者认为可以通过让attention maps更不同以及让其感知局部信息来解决这一问题。

这个观察特征演进的方法可以学一下

2.2 Attention Expansion

对于MHSA的头大数量，首先有这样的直觉：

Increasing the number of attention heads within MHSA, which is shown to be effective at improving the model performance, can potentially increase diversity among the attention maps.

If naively adding more SA heads, for a model with fixed hidden dimension, it is difficult to trade-off the benefit of having more SA heads and the harm of reducing the embedding dimension per head.

为了平衡这两点，作者采用linear transformation 来实现self-attention map的拓展。

原始的ViT计算方法：

Then ViT applies multiple multi-head self-attention (MHSA) and feedforward layers to process the patch embeddings to model their long-range dependencies and evolve the token embedding features. Suppose the input tensor is $X \in \mathbb{R}^{d_{in} \times n}$, the MHSA applies linear transformation with parameters W_K, W_Q, W_V to embed them into the key $K = W_K X \in \mathbb{R}^{d \times n}$, query $Q = W_Q X \in \mathbb{R}^{d \times n}$ and value $V = W_V X \in \mathbb{R}^{d \times n}$ respectively. Suppose there are H self-attention heads. These embeddings are uniformly split into H segments $Q_h, K_h, V_h \in \mathbb{R}^{d/H \times n}$. Then the MHSA module computes the head-specific attention matrix (map)[1] A and aggregate the token value features as follows:

$$\text{Attention}(X, h) = A^h V_h^\top \text{ with } A^h = \text{Softmax}\left(\frac{Q_h^\top K_h}{\sqrt{d/H}}\right), h = 1, \ldots, H. \quad (1)$$

Extension方法：

用linear projection $W_A \in \mathbb{R}^{H' \times H}$将$A[A^1, \ldots, A^H]$映射到更多头的版本$\tilde{A} = [\tilde{A}^1, \ldots, \tilde{A}^H]$

$$\tilde{A}^h = \sum_{i=1}^{H} W_A(h, i) \cdot A^i, h = 1, \ldots, H'$$

2.3 Distributed Local Attention

这个部分跟过往对特征施加卷积的方法不同的地方在于，其直接对self attention map加卷积。

对计算得到的某个头h的attention map A^h，在其上采用一个可学的k x k卷积：

$$A^{h*}_{i,j} = \sum_{a,b=1}^{k} \mathbf{w}_{a,b} \cdot A^h_{i-\lfloor \frac{k}{2} \rfloor + a, j - \lfloor \frac{k}{2} \rfloor + b}.$$

然后用这个新的attention map跟v计算得到SA的输出。

此外，作者证明了这样的操作与对feature做局部操作等价。

Though being conceptually simple, the above operation establishes an interesting synergy between the global context aggregation (with self-attention) and local context modeling (with convolution). To see this, consider applying 1D convolution $\mathbf{w}$ of length k to obtain the convolution-augmented SA matrix, with which the feature aggregation becomes

$$\begin{aligned}\tilde{\mathbf{v}}_i &= \sum_{j=1}^{n} A^{h*}_{i,j} \cdot \mathbf{v}_j = \sum_{j=1}^{n} \left(\sum_{a=1}^{k} \mathbf{w}_a \cdot A^h_{i,j-\lfloor \frac{k}{2} \rfloor + a} \right) \cdot \mathbf{v}_j \\ &= \sum_{j=1}^{n} \left(\sum_{a=1}^{k} \mathbf{w}_a \cdot A^h_{i,j-\lfloor \frac{k}{2} \rfloor + a} \cdot \mathbf{v}_j \right) = \sum_{j=1}^{n} \left(\sum_{a=1}^{k} A^h_{i,j} \cdot \mathbf{w}_a \cdot \mathbf{v}_{j-a+\lfloor \frac{k}{2} \rfloor} \right).\end{aligned}$$

The above clearly shows that the feature aggregation based on the convolution-processed attention matrix is equivalent to: (1) applying the convolution $\mathbf{w}$, with a location-specific reweighing scalar $A^h_{i,j}$, to aggregate features locally at first and (2) summing over the locally aggregated features. Therefore, we name such an operation as *distributed local attention* (DLA).

这个真的可以学！

2.4 Linear reduction

为了减少计算开销并保持embedding 维度的一致，在与V相乘前，还需要进行跟expension类似的linear reduction操作将头数降下来。

综上，Refiner的结构可以被描述为下图：

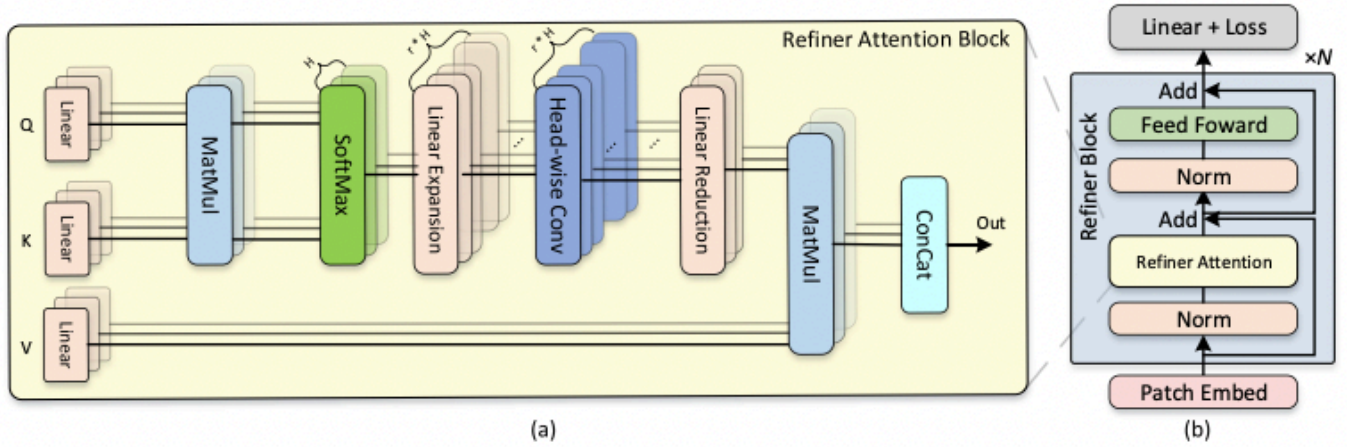

Figure 3: (a) Architecture design of refiner. Different from the vanilla self-attention block, the refiner applies linear attention expansion to attention maps output from the softmax operation to increase their number. Then head-wise spatial convolution is applied to augment these expanded attention maps. Finally another linear projection is deployed to reduce the number of attention maps to the original one. Note that $r = H'/H$ is the expansion ratio. (b) Modified transformer block with refiner as a drop-in component.

3、Experiment

3.1 Ablation study

Effect of attention expansion

Expan. Ratio	Params	Converge (#Epoch)	Top-1 (%)
1	25M	300	82.3
2	25M	300	82.8
3	25M	273	82.8
4	25M	270	82.9
6	25M	261	83.0

随着expansion ratio的增加，模型的param没发生大的改变，但是可以更快收敛到更好的结果。

Effect of attention reduction

Table 3: Effect of attention reduction on Refined-ViT-16B with 384 hidden dimension.

Model	Attn. map	Top-1 (%)
Refined-ViT-16B	w/o reduction	82.99
Refined-ViT-16B	w/ reduction	82.95

加入reduction以后效果有所下降，但这是可以理解的，不过我觉得这里可以做一个关于计算开销和参数量的对比验证reduction 很好地做到了accuracy-computation trand-off。

Effect of distributed local attention

Table 2: Impacts of convolution on attention maps. We directly apply the 3×3 convolution on the attention maps from the multi-head self-attention of ViTs with respect to various architectures. We can observe clear improvement for all ViT variants when adding the proposed DLA.

Model	#Blocks	Hidden dim	#Heads	Params	Top-1 (%)
ViT	12	768	12	86M	79.5
+ DLA	12	768	12	86M	81.2
ViT	16	384	12	24M	78.9
+ DLA	16	384	12	24M	80.3
ViT	24	384	12	36M	79.3
+ DLA	24	384	12	36M	80.9
ViT	32	384	12	48M	79.2
+ DLA	32	384	12	48M	81.1

Effect of the local attention kernels

Table 4: Evaluation on how the spatial span within DLA affects the model performance. We compare the model performance with three different constraints on the local kernels.

Model	Constraints	Top-1 (%)
Refined-ViT-16B	None	83.0
Refined-ViT-16B	Spatial	82.7
Refined-ViT-16B	Row+Col	81.7

Refiner augments attention maps and accelerates feature evolving

这个实验要学着怎么去做

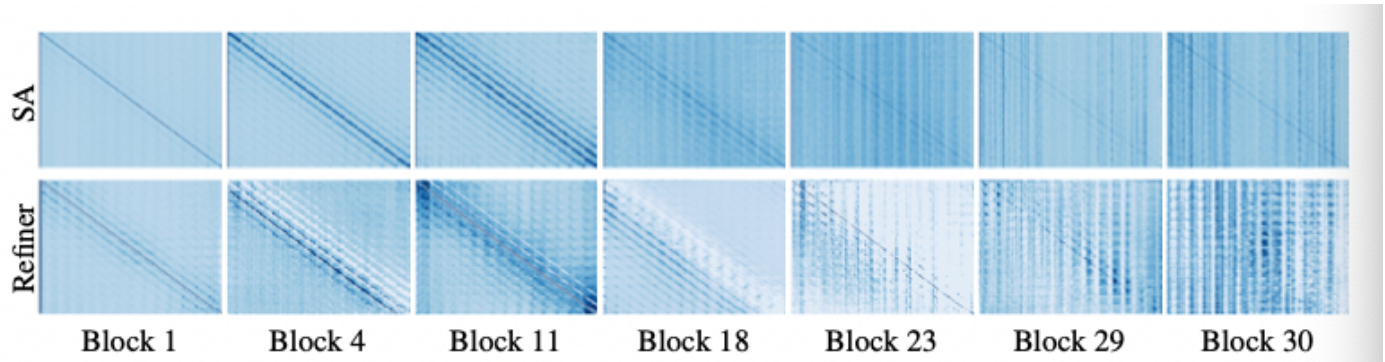

Figure 5: Compared with the attention matrices A from the vanilla SA (top), for deeper blocks, refiner (bottom) strengthens the local patterns of their attention maps, making them less uniform and better model local context.

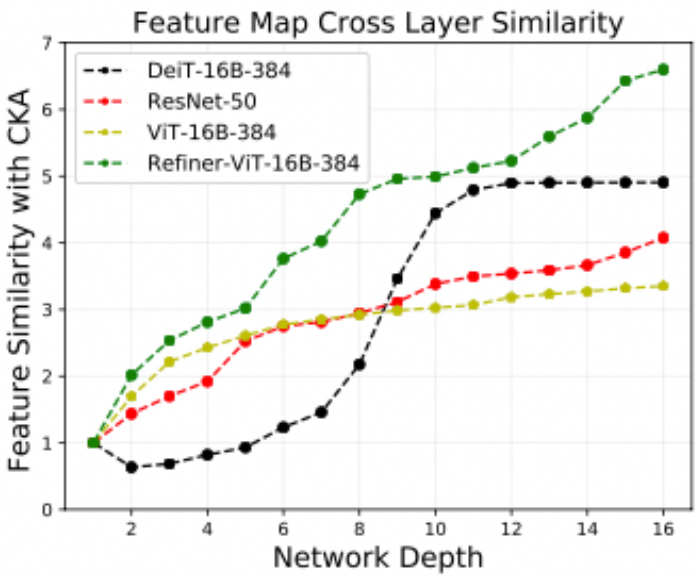

Figure 4: Refiner accelerates feature evolving compared with CNNs, the vanilla ViT and the Deit trained with a more complex scheme.