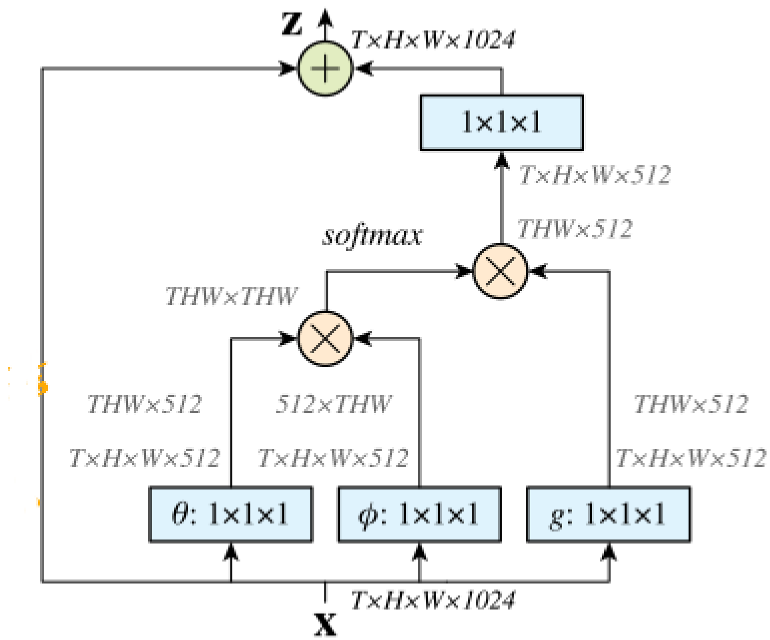
2021.12.5 论文整理

Non local Network中存在的退化现象

自注意力机制



- 利用token与其他tokens的关系构建长距离依赖，转换token的特征。
- 通过对Value的加权求和，其中权重由Query和Key的内积计算得出。

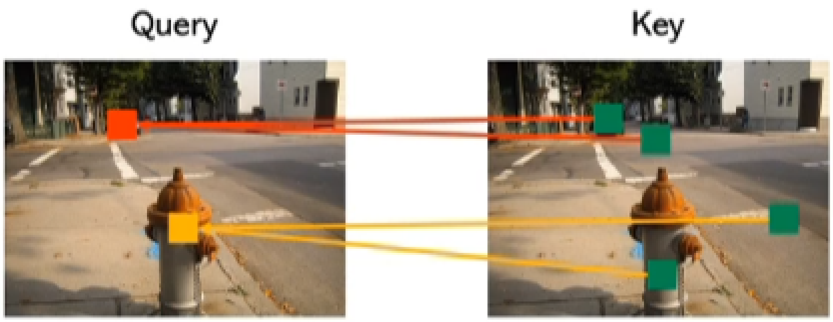
代表性的工作: Non local Neural Networks(CVPR 2018), Relation Networks(CVPR 2018)....

被应用在很多的视觉任务中，包括图像分类、目标检测、语义分割、行为识别等。

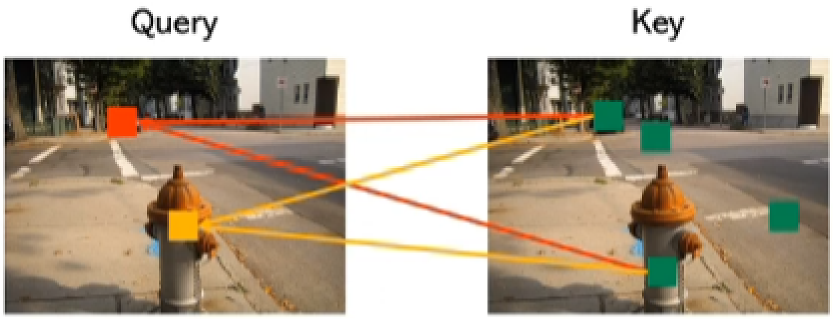
退化现象

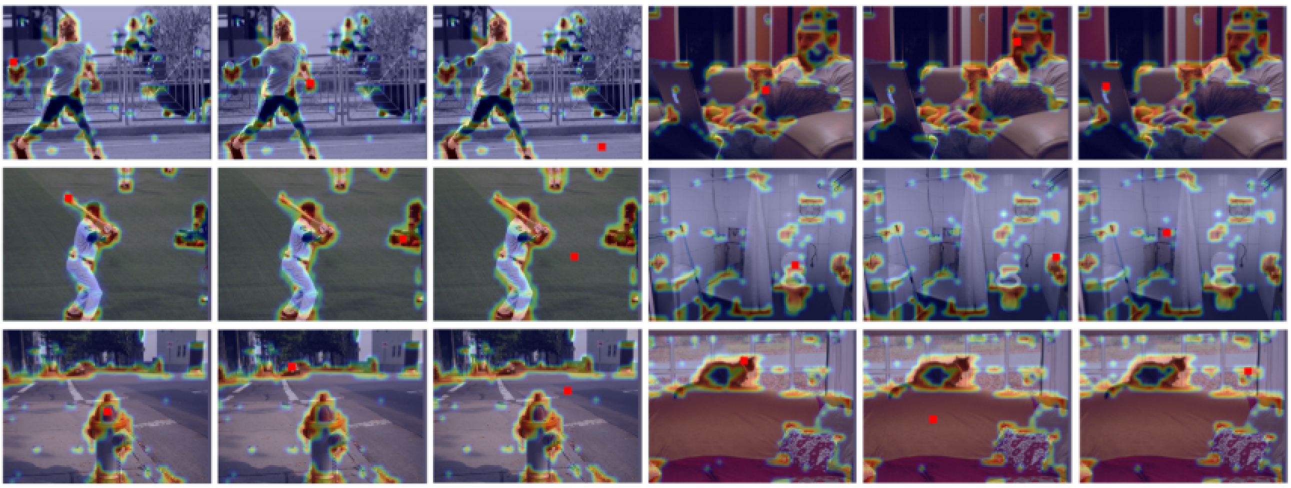
问题：在视觉任务中，自注意力是否真的能够学到不同**token**之间的相关关系？

预期的效果是对于不同的query，会受到不同的key的影响。

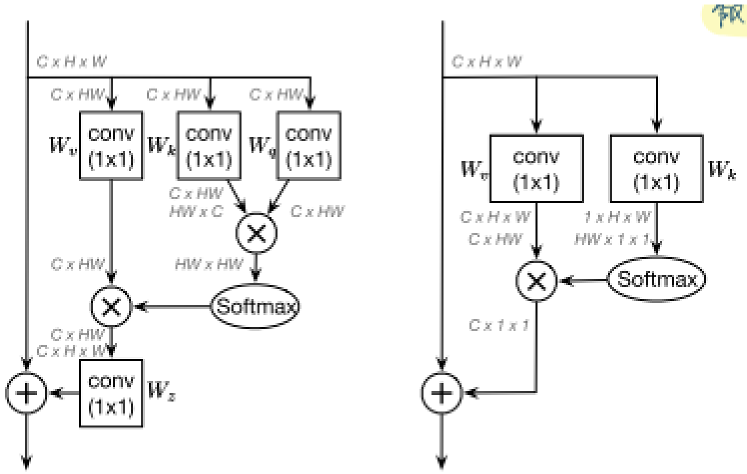


但是通过可视化，发现实际得到的注意力图退化为了一个全局的一阶模型。





基于该发现，(ICCV 2019) GCNet: Non-local Networks Meet Squeeze-Excitation Networks and Beyond 将Non Local Block进行简化，将冗余的pairwise计算优化为计算一个与query无关的全局注意力图。在减少计算量的同时，保留了NL Block的性能。



(a) NL block

(b) Simplified NL block (Eqn 2)

(ECCV 2020) Disentangled non-local neural networks 则在此基础上进行更深入的讨论。

Why?

$$w(\mathbf{q}_i,\mathbf{k}_j) \sim exp(\mathbf{q}_i^T\mathbf{k}_j) = exp(\underbrace{(\mathbf{q}_i-\boldsymbol{\mu}_q)^T(\mathbf{k}_j-\boldsymbol{\mu}_k)}_{\text{(whitened) pairwise}} + \underbrace{\boldsymbol{\mu}_q^T\mathbf{k}_j}_{\text{(hidden) unary}})$$

* $\boldsymbol{\mu}_q$ and $\boldsymbol{\mu}_k$ are global average of $\mathbf{q}$ and $\mathbf{k}$

- 对注意力计算的数学表达式进行拆解，可以得到两项，其中前一项与query有关，第二项与query无关。

> μ 的选择可以通过求解损失函数，使得不同的*query-key*对权重差异最大。
>
> $$\arg\max_{\alpha,\beta} \frac{\sum_{i,m,n\in\Omega}\left((\mathbf{q}_i-\alpha)^T(\mathbf{k}_m-\beta)-(\mathbf{q}_i-\alpha)^T(\mathbf{k}_n-\beta)\right)^2}{\sum_{i\in\Omega}\left((\mathbf{q}_i-\alpha)^T(\mathbf{q}_i-\alpha)\right)\cdot\sum_{m,n\in\Omega}\left((\mathbf{k}_m-\mathbf{k}_n)^T(\mathbf{k}_m-\mathbf{k}_n)\right)} + \frac{\sum_{m,i,j\in\Omega}\left((\mathbf{k}_m-\beta)^T(\mathbf{q}_i-\alpha)-(\mathbf{k}_m-\beta)^T(\mathbf{q}_j-\alpha)\right)^2}{\sum_{m\in\Omega}\left((\mathbf{k}_m-\beta)^T(\mathbf{k}_m-\beta)\right)\cdot\sum_{i,j\in\Omega}\left((\mathbf{q}_i-\mathbf{q}_j)^T(\mathbf{q}_i-\mathbf{q}_j)\right)}$$
>
> - 取*min*并将函数取负，此时该函数的海森矩阵是一个半正定矩阵，即该函数是一个凸函数。可以通过求导数为零，得到全局最优解，即*q*、*k*均值。

其中前一项的操作被称作白化，个人理解即将两项特征进行解耦。

method	formulation	mIoU
Baseline	none	75.8%
Joint (Self-Attention)	$\sim exp(\mathbf{q}_i^T\mathbf{k}_j)$	$\boxed{78.5\%}$
Pairwise Alone	$\sim exp((\mathbf{q}_i-\boldsymbol{\mu}_q)^T(\mathbf{k}_j-\boldsymbol{\mu}_k))$	77.5%
Unary Alone	$\sim exp(\boldsymbol{\mu}_q^T\mathbf{k}_j)$	$\boxed{79.3\%}$ +0.8%

Quantitative results on semantic segmentation (Cityscapes)

可以看出两项一起学习时，性能还比不上单独使用**unary**一项的性能。**Why**?

$$w(\mathbf{q}_i,\mathbf{k}_j) \sim exp((\mathbf{q}_i-\boldsymbol{\mu}_q)^T(\mathbf{k}_j-\boldsymbol{\mu}_k) + \boldsymbol{\mu}_q^T\mathbf{k}_j)$$

$$= \underbrace{exp((\mathbf{q}_i-\boldsymbol{\mu}_q)^T(\mathbf{k}_j-\boldsymbol{\mu}_k))}_{\text{Pairwise } \boldsymbol{w_p}} \times \underbrace{exp(\boldsymbol{\mu}_q^T\mathbf{k}_j)}_{\text{Unary } \boldsymbol{w_u}}$$

将pairwise项与unary项拆解开，可以看到两者呈相乘关系，这就带来一个问题。

$$\frac{\partial L}{\partial w_p} = \frac{\partial L}{\partial w}\frac{\partial w}{\partial w_p} \sim \frac{\partial L}{\partial w} w_u \qquad \frac{\partial L}{\partial w_u} = \frac{\partial L}{\partial w}\frac{\partial w}{\partial w_u} \sim \frac{\partial L}{\partial w} w_p$$

两项的梯度相互耦合，若一项的值接近于零，另一项的学习也将会停止。将乘法操作转变为加法，两者就解耦了，此时两者的梯度也解耦了。

通过可视化发现，两项对应的是完全不同的信息，pairwise项对应与query同类别的区域，unary则注重不同类别的边界。