论自注意力机制与卷积层之间的关系

让-巴蒂斯特·科尔东尼耶、安德烈亚斯·卢卡斯与马丁·贾吉
瑞士洛桑联邦理工学院（EPFL）édérale de Lausanne (EPFL
) {first.last}@epfl.ch

摘要

近年来，视觉领域引入注意力机制的趋势促使研究者重新审视卷积层作为核心构建模块的主导地位。Ramachandran等人（2019）的研究表明，注意力机制不仅能帮助CNN处理长程依赖关系，甚至可以完全取代卷积操作，在视觉任务中实现最先进的性能。这引发了一个问题：习得的注意力层是否以类似于卷积层的方式运作？本研究通过实证表明，注意力层能够执行卷积运算，且在实践中确实经常学习到这种模式。具体而言，我们证明了具有足够多头数的多头自注意力层，其表达能力至少不逊于任何卷积层。随后的数值实验显示，自注意力层会像CNN层一样关注像素网格模式，这进一步验证了我们的理论分析。代码已公开[1]。

1 引言

自然语言处理（NLP）领域的最新进展，很大程度上归功于*trans-former* (Vaswani等人2017年)提出的变革性架构。通过在大量文本语料库上进行无监督任务预训练，诸如GPT-2（Radford等人，2018）、BERT（Devlin等人，2018）和Transformer-XL（Dai等人，2019）等基于Transformer的模型，似乎具备了学习文本底层结构的能力，从而能够习得跨任务泛化的表征。Transformer与此前方法（如循环神经网络（Hochreiter & Schmidhuber，1997）和卷积神经网络（CNN））的关键区别在于，前者能同时关注输入序列中的每个单词。这一特性得益于*attentionmechanism*——该机制最初在神经机器翻译中引入，旨在更好地处理长距离依赖关系（Bahdanau等人，2015）。特别是自注意力机制，它通过衡量两个单词表征之间的距离来计算注意力分数，从而捕捉序列中单词间的相似性。随后，每个单词的表征会根据注意力分数最高的那些单词进行更新。

受其学习词语间有意义相互依赖关系的能力启发，研究人员近期开始探索将自注意力机制应用于视觉任务。最初，自注意力通过两种方式被引入CNN：一种是基于通道的注意力机制（Hu等人，2018），另一种是利用图像中的非局部关系（Wang等人，2018）。最近，Bello等人（2019）通过用自注意力层替换部分卷积层来增强CNN，从而在图像分类和物体检测任务上取得了提升。值得注意的是，Ramachandran等人（2019）发现，尽管结合注意力与卷积特征能取得最先进的结果，但在相同的计算量和模型大小限制下，自注意力-*only*架构也能达到具有竞争力的图像分类精度。

These findings raise the question, do self-attention layers process images in a similar manner to convolutional layers? 从理论角度来看，可以认为transformer具备模拟任何函数的能力——包括CNN。事实上，Pérez等人（2019年）的研究表明，在无限精度算术等强理论假设下，采用加法位置编码的多层注意力架构具有图灵完备性。遗憾的是，普适性结论并不能揭示机器如何解决具体任务，仅能证明其具备解决能力。因此，自注意力层究竟如何处理图像这一问题仍悬而未决。

1代码：github.com /epfml/attention-cnn. Website: epfml.github.io/attention-cnn. 1

贡献。在这项工作中，我们提出了理论和实证证据，表明自注意力层能够（并且确实）学会表现得类似于卷积层：

一、从理论角度出发，我们提供了一个构造性证明，表明自注意力层能够表达任何卷积层。

具体而言，我们证明了使用相对位置编码的单层多头自注意力机制可通过重新参数化来表达任何卷积层。

二、我们的实验表明，仅注意力架构（Ramachandran等人，2019）的前几层确实学会了关注每个查询像素周围的网格状模式，这与我们的理论构建类似。

引人注目的是，这种行为不仅在我们的二次编码中得到验证，也在学习到的相对编码中得以确认。我们的结果似乎表明，局部卷积是图像分类网络前几层的正确归纳偏置。我们提供了一个交互式网站[2]，用于探索自注意力机制如何在较低层利用基于位置的局部注意力，而在更深层则转向基于内容的注意力。为了确保可复现性，我们的代码已公开提供。

2 视觉注意力机制的背景

我们在此回顾自注意力层的数学表述，并着重强调位置编码{v*}的作用。

2.1 多头自注意力层

设$\boldsymbol{X} \in \mathbb{R}^{T \times D_{in}}$为一个输入矩阵，由$T$个维度各为$D_{in}$的令牌组成。在自然语言处理（NLP）中，每个令牌对应句子中的一个单词，但同样的形式体系可应用于任何T离散对象的序列，例如像素。自注意力层将任意查询令牌$t \in [T]$从D_{in}维映射到D_{out}维，具体如下：

$$\text{Self-Attention}(\boldsymbol{X})_{t,:} := \text{softmax}\left(\boldsymbol{A}_{t,:}\right)\boldsymbol{X}\boldsymbol{W}_{val}, \tag{1}$$

这里我们指的是$T \times T$矩阵中的元素

$$\boldsymbol{A} := \boldsymbol{X}\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}\boldsymbol{X}^{\top} \tag{2}$$

作为*attention scores*和softmax输出[3]作为*attention probabilities*。该层由查询矩阵$\boldsymbol{W}_{qry} \in \mathbb{R}^{D_{in} \times D_k}$、键矩阵$\boldsymbol{W}_{key} \in \mathbb{R}^{D_{in} \times D_k}$和值矩阵$\boldsymbol{W}_{val} \in \mathbb{R}^{D_{in} \times D_{out}}$参数化。为简化起见，我们排除了任何残差连接、批量归一化及常数因子。

上述自注意力模型的一个关键特性是它对重新排序具有等变性，也就是说，无论T输入标记如何打乱，它都会给出相同的输出。这对于我们期望事物顺序至关重要的情况来说是有问题的。为了缓解这一限制，序列中的每个标记（或图像中的像素）都会学习一个*positional encoding*，并在应用自注意力之前将其添加到标记本身的表示中。

$$\boldsymbol{A} := (\boldsymbol{X} + \boldsymbol{P})\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}(\boldsymbol{X} + \boldsymbol{P})^{\top}, \tag{3}$$

其中$\boldsymbol{P} \in \mathbb{R}^{T \times D_{in}}$包含每个位置的嵌入向量。更一般地，$\boldsymbol{P}$可替换为任何能返回位置向量表示的函数。

实践中发现，将这种自注意力机制复制为*multiple heads*是有益的，每个机制通过使用不同的查询、键和值矩阵，能够关注输入的不同部分。在多头自注意力中，输出维度为D_h的N_h个头部的输出被拼接起来，并按如下方式投影到D_{out}维度：

$$\text{MHSA}(\boldsymbol{X}) := \underset{h \in [N_h]}{\text{concat}}\left[\text{Self-Attention}_h(\boldsymbol{X})\right]\boldsymbol{W}_{out} + \boldsymbol{b}_{out} \tag{4}$$

并引入了两个新参数：投影矩阵$\boldsymbol{W}_{out} \in \mathbb{R}^{N_h D_h \times D_{out}}$和偏置项$\boldsymbol{b}_{out} \in \mathbb{R}^{D_{out}}$。

[2] epfml.github.io/attention-cnn

[3] $\text{softmax}\left(\boldsymbol{A}_{t,:}\right)_k = \exp(\boldsymbol{A}_{t,k}) / \sum_p \exp(\boldsymbol{A}_{t,p})$

2.2 图像的注意力机制

卷积层是构建用于处理图像的神经网络的*de facto*选择。我们回顾一下，给定一个宽度为W、高度为H、具有D_{in}个通道的图像张量$\mathbf{X}\in\mathbb{R}^{W\times H\times D_{in}}$，卷积层对于像素$(i,j)$的输出由以下公式给出

$$\mathrm{Conv}(\boldsymbol{X})_{i,j,:}:=\sum_{(\delta_1,\delta_2)\in\mathbb{\Delta}_K}\mathbf{X}_{i+\delta_1,j+\delta_2,:}\mathbf{W}_{\delta_1,\delta_2,:,:}+\boldsymbol{b}, \tag{5}$$

其中$\mathbf{W}$是$K\times K\times D_{in}\times D_{out}$权重张量[4]，$\boldsymbol{b}\in\mathbb{R}^{D_{out}}$为偏置向量，集合

$$\mathbb{\Delta}_K:=\left[-\left\lfloor\frac{K}{2}\right\rfloor,\cdots,\left\lfloor\frac{K}{2}\right\rfloor\right]\times\left[-\left\lfloor\frac{K}{2}\right\rfloor,\cdots,\left\lfloor\frac{K}{2}\right\rfloor\right]$$

包含与$K\times K$核卷积图像时出现的所有可能位移。

以下，我们回顾如何将自注意力机制从一维序列适配到图像。

对于图像而非标记，我们拥有查询和关键像素$\boldsymbol{q},\boldsymbol{k}\in[W]\times[H]$。相应地，输入是一个维度为$W\times H\times D_{in}$的张量$\mathbf{X}$，每个注意力分数关联一个查询像素和一个关键像素。

为了使公式与一维情况保持一致，我们采用了符号上的简化，通过二维索引向量对张量进行切片：若$\boldsymbol{p}=(i,j)$，则记$\mathbf{X}_{\boldsymbol{p},:}$和$\mathbf{A}_{\boldsymbol{p},:}$分别表示$\mathbf{X}_{i,j,:}$和$\mathbf{A}_{i,j,:,:}$。在此符号约定下，像素$\boldsymbol{q}$处的多头自注意力层输出可表示为：

$$\text{Self-Attention}(\boldsymbol{X})_{\boldsymbol{q},:}=\sum_{\boldsymbol{k}}\mathrm{softmax}\left(\mathbf{A}_{\boldsymbol{q},:}\right)_{\boldsymbol{k}}\mathbf{X}_{\boldsymbol{k},:}\,\boldsymbol{W}_{val} \tag{6}$$

相应地，对于多头情况也是如此。

2.3 图像的位置编码

在基于Transformer的架构中，使用了两种类型的位置编码：*absolute*和*relative*编码（另见附录中的表3）。

在绝对编码中，每个像素$\boldsymbol{p}$都被分配一个（固定或学习得到的）向量$\mathbf{P}_{\boldsymbol{p},:}$。那么，我们在等式(2)中看到的注意力分数计算可以分解如下：

$$\begin{aligned}\mathbf{A}^{\mathrm{abs}}_{\boldsymbol{q},\boldsymbol{k}}&=(\mathbf{X}_{\boldsymbol{q},:}+\mathbf{P}_{\boldsymbol{q},:})\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}(\mathbf{X}_{\boldsymbol{k},:}+\mathbf{P}_{\boldsymbol{k},:})^{\top}\\&=\mathbf{X}_{\boldsymbol{q},:}\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}\mathbf{X}_{\boldsymbol{k},:}^{\top}+\mathbf{X}_{\boldsymbol{q},:}\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}\mathbf{P}_{\boldsymbol{k},:}^{\top}+\mathbf{P}_{\boldsymbol{q},:}\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}\mathbf{X}_{\boldsymbol{k},:}+\mathbf{P}_{\boldsymbol{q},:}\boldsymbol{W}_{qry}\boldsymbol{W}_{key}^{\top}\mathbf{P}_{\boldsymbol{k},:}\end{aligned} \tag{7}$$

其中$\boldsymbol{q}$和$\boldsymbol{k}$分别对应查询像素和键像素。

相对位置编码由Dai等人（2019年）提出。其核心思想是仅考虑查询像素（我们计算表征的像素）与键像素（我们关注的像素）之间的位置差异，而非键像素的绝对位置：

$$\mathbf{A}^{\mathrm{rel}}_{\boldsymbol{q},\boldsymbol{k}}:=\mathbf{X}_{\boldsymbol{q},:}^{\top}\boldsymbol{W}_{qry}^{\top}\boldsymbol{W}_{key}\,\mathbf{X}_{\boldsymbol{k},:}+\mathbf{X}_{\boldsymbol{q},:}^{\top}\boldsymbol{W}_{qry}^{\top}\widehat{\boldsymbol{W}}_{key}\,\boldsymbol{r}_{\boldsymbol{\delta}}+\boldsymbol{u}^{\top}\boldsymbol{W}_{key}\,\mathbf{X}_{\boldsymbol{k},:}+\boldsymbol{v}^{\top}\widehat{\boldsymbol{W}}_{key}\,\boldsymbol{r}_{\boldsymbol{\delta}} \tag{8}$$

通过这种方式，注意力分数仅取决于偏移量$\boldsymbol{\delta}:=\boldsymbol{k}-\boldsymbol{q}$。其中，可学习向量$\boldsymbol{u}$和$\boldsymbol{v}$对每个注意力头是唯一的，而对于每个偏移量$\boldsymbol{\delta}$，相对位置编码$\boldsymbol{r}_{\boldsymbol{\delta}}\in\mathbb{R}^{D_p}$则由所有层和头共享。此外，现在键权重被分为两种类型：$\boldsymbol{W}_{key}$与输入相关，而$\widehat{\boldsymbol{W}}_{key}$则与像素的相对位置相关。

3 自注意力作为卷积层

本节推导了使多头自注意力层能够模拟卷积层的充分条件。我们的主要结果如下：

定理1. *A multi-head self-attention layer with N_h heads of dimension D_h, output dimension D_{out} and a relative positional encoding of dimension $D_p\geq 3$ can express any convolutional layer of kernel size $\sqrt{N_h}\times\sqrt{N_h}$ and* $\min(D_h, D_{out})$ *output channels.*

[4] 为了简化表示，我们将张量的前两个维度从$-\lfloor K/2\rfloor$索引到$\lfloor K/2\rfloor$。

该定理通过构造性地选择多头自注意力层的参数得以证明，使得该层能够像卷积层一样运作。在所提出的构造中，每个自注意力头的注意力分数应关注于$\mathbb{\Delta}_K = \{-\lfloor K/2 \rfloor, \dots, \lfloor K/2 \rfloor\}^2$这一集合内不同的相对位移，该集合代表了$K \times K$核中所有像素位移的可能。具体条件可参见引理1的陈述。

随后，引理2表明，前述条件在我们称为*quadratic encoding*的相对位置编码中得到了满足：

$$\boldsymbol{v}^{(h)} := -\alpha^{(h)}\,(1, -2\boldsymbol{\Delta}_1^{(h)}, -2\boldsymbol{\Delta}_2^{(h)}) \quad \boldsymbol{r}_{\boldsymbol{\delta}} := (\|\boldsymbol{\delta}\|^2, \boldsymbol{\delta}_1, \boldsymbol{\delta}_2) \quad \boldsymbol{W}_{qry} = \boldsymbol{W}_{key} := \boldsymbol{0} \quad \widehat{\boldsymbol{W}_{key}} := \boldsymbol{I} \tag{9}$$

学习到的参数$\boldsymbol{\Delta}^{(h)} = (\boldsymbol{\Delta}_1^{(h)}, \boldsymbol{\Delta}_2^{(h)})$和$\alpha^{(h)}$分别决定了每个注意力头的中心和宽度。另一方面，$\boldsymbol{\delta} = (\boldsymbol{\delta}_1, \boldsymbol{\delta}_2)$是固定的，表达了查询像素与键像素之间的相对偏移。

需要强调的是，上述编码方式并非唯一能满足引理1条件的方案。实际上，在我们的实验中，神经网络学习到的相对编码同样符合该引理的条件（尽管与二次编码有所不同）。然而，上文定义的编码在规模效率上表现卓越——仅需$D_p =$三个维度即可编码像素的相对位置，同时还能达到与学习所得编码相当或更优的实证性能。

该定理涵盖了如式(17)所定义的一般卷积算子。然而，使用微分编程框架(Paszke et al., 2017; Abadi et al., 2015)的机器学习从业者可能会质疑：该定理是否对所有二维卷积层的超参数{v*}都成立：

- *Padding*多头自注意力层默认使用“SAME”填充方式，而卷积层会使图像尺寸每边减少$K-1$个像素。缓解这种边界效应的正确方法是在输入图像的每侧填充$\lfloor K/2 \rfloor$个零值。这样处理后，MHSA（多头自注意力）层与卷积层裁剪后的输出尺寸将保持一致。
- *Stride*步进卷积可视为卷积后接一个固定的池化操作——并进行了计算优化。定理1针对的是步长为1的情况，但可以通过在自注意力层后附加一个固定池化层来模拟任意步长。
- *Dilation*多头自注意力层能够表达任何扩张卷积，因为每个头可以关注任意像素偏移处的值，并形成（扩张的）网格模式。

一维情况下的说明。作用于序列的卷积层在文本（Kim, 2014）、音频（van den Oord等, 2016）和时间序列（Franceschi等, 2019）的研究中已被广泛采用。定理1可直接推广以证明：具有N_h个头部的多头自注意力机制，配合维度为$D_p \geq$的位置编码，同样能够模拟核大小为$K = N_h$、输出通道数为$\min(D_h, D_{out})$的一维卷积层。由于我们尚未通过实证检验上述构造是否与实际一维自注意力的行为相符，故无法断言其确实学会了卷积输入序列——仅能确认其具备这种能力。

主要定理的证明

证明直接源于下述引理1和引理2：

引理1. *Consider a multi-head self-attention layer consisting of $N_h = K^2$ heads, $D_h \geq D_{out}$ and let $\boldsymbol{f} : [N_h] \to \mathbb{\Delta}_K$ be a bijective mapping of heads onto shifts. Further, suppose that for every head the following holds:*

$$\operatorname{softmax}(\boldsymbol{A}^{(h)}_{\boldsymbol{q},:})_{\boldsymbol{k}} = \begin{cases} 1 & \text{if } \boldsymbol{f}(h) = \boldsymbol{q} - \boldsymbol{k} \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

Then, for any convolutional layer with a $K \times K$ kernel and D_{out} output channels, there exists $\{\boldsymbol{W}^{(h)}_{val}\}_{h \in [N_h]}$ such that 多头自注意力$(\boldsymbol{X}) =$ 卷积$(\boldsymbol{X})$ *for every $\boldsymbol{X} \in \mathbb{R}^{W \times H \times D_{in}}$.*

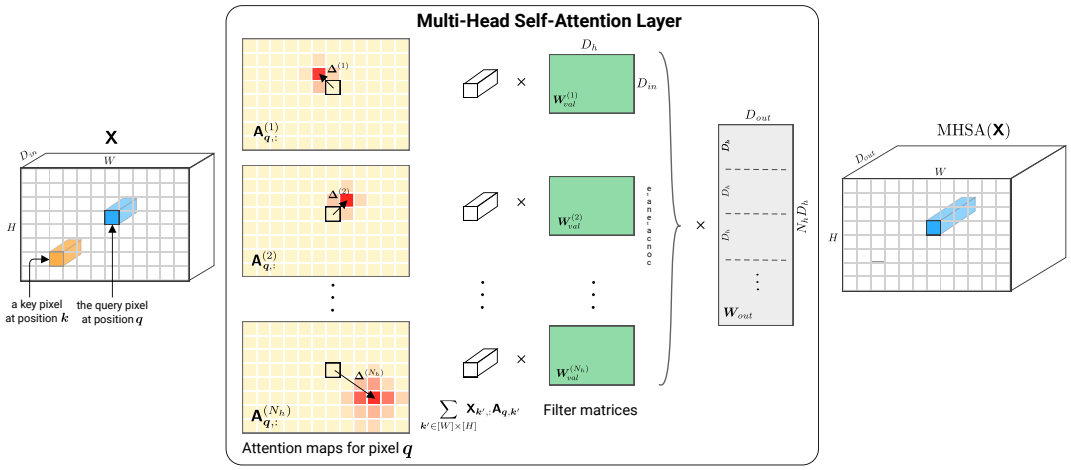

图1：应用于张量图像X的多头自注意力层示意图。每个头h关注位移$\boldsymbol{\Delta}^{(h)}$周围的像素值，并学习一个滤波器矩阵$\boldsymbol{W}_{val}^{(h)}$。我们展示了针对位置$\boldsymbol{q}$处查询像素计算得到的注意力图。

Proof. 我们的第一步将是重新表述多头自注意力算子，从方程(1)和方程(4)出发，使得多头机制的效果更加透明：

$$\text{MHSA}(\boldsymbol{X}) = \boldsymbol{b}_{out} + \sum_{h\in[N_h]} \text{softmax}(\boldsymbol{A}^{(h)})\boldsymbol{X}\underbrace{\boldsymbol{W}_{val}^{(h)}\boldsymbol{W}_{out}[(h-1)D_h+1:hD_h+1]}_{\boldsymbol{W}^{(h)}} \tag{11}$$

需要注意的是，每个注意力头的值矩阵$\boldsymbol{W}_{val}^{(h)} \in \mathbb{R}^{D_{in}\times D_h}$以及维度为$D_h \times D_{out}$的投影矩阵$\boldsymbol{W}_{out}$的每个块都是通过学习得到的。假设$D_h \geq D_{out}$，我们可以为每个头用学习到的矩阵$\boldsymbol{W}^{(h)}$替换每一对矩阵。我们考虑多头自注意力机制的一个输出像素：

$$\text{MHSA}(\boldsymbol{X})_{\boldsymbol{q},:} = \sum_{h\in[N_h]} \left(\sum_{\boldsymbol{k}} \text{softmax}(\mathbf{A}_{\boldsymbol{q},:}^{(h)})_{\boldsymbol{k}}\mathbf{X}_{\boldsymbol{k},:}\right)\boldsymbol{W}^{(h)} + \boldsymbol{b}_{out} \tag{12}$$

根据引理的条件，对于第h个注意力头，当$\boldsymbol{k} = \boldsymbol{q} - \boldsymbol{f}(h)$时注意力概率为1，否则为零。因此，该层在像素$\boldsymbol{q}$处的输出等于

$$\text{MHSA}(\mathbf{X})_{\boldsymbol{q}} = \sum_{h\in[N_h]} \mathbf{X}_{\boldsymbol{q}-\boldsymbol{f}(h),:}\boldsymbol{W}^{(h)} + \boldsymbol{b}_{out} \tag{13}$$

对于$K = \sqrt{N_h}$，上述内容可视为等同于方程17中表达的卷积层：存在一个一对一映射（由映射$\boldsymbol{f}$隐含），在$h = [N_h]$的矩阵$\boldsymbol{W}^{(h)}$与所有$(k_1, k_2) \in [K]^2$.的矩阵$\mathrm{W}_{k_1,k_2,:,:}$之间。 □

关于D_h和$D_{\boldsymbol{out}}$的说明。在基于Transformer的架构中，通常设置$D_h = D_{out}/N_h$，因此$D_h < D_{out}$。在这种情况下，$\boldsymbol{W}^{(h)}$可视为秩为$D_{out} - D_h$，这不足以表达每个具有D_{out}通道的卷积层。然而，可以看出MHSA($\boldsymbol{X}$)的D_{out}个输出中的任意D_h个，都能表达任何具有D_h个输出通道的卷积层的输出。为了涵盖这两种情况，在主定理的陈述中，我们断言卷积层的输出通道应为$\min(D_h, D_{out})$。实践中，我们建议将维度为$D_h = D_{out}$的头部进行拼接，而非在头部间分割D_{out}维度，以实现精确的重参数化，避免“未使用”的通道。

引理2. *There exists a relative encoding scheme* $\{\boldsymbol{r}_{\boldsymbol{\delta}} \in \mathbb{R}^{D_p}\}_{\boldsymbol{\delta}\in\mathbb{Z}^2}$ *with* $D_p \geq 3$ *and parameters* $\boldsymbol{W}_{qry}, \boldsymbol{W}_{key}, \widehat{\boldsymbol{W}}_{key}, \boldsymbol{u}$ *with* $D_p \leq D_k$ *such that, for every* $\boldsymbol{\Delta} \in \mathbb{\Delta}_K$ *there exists some vector* $\boldsymbol{v}$ *(conditioned on* $\boldsymbol{\Delta}$*) yielding* $\text{softmax}(\mathrm{A}_{\boldsymbol{q},:})_{\boldsymbol{k}} = 1$ *if* $\boldsymbol{k} - \boldsymbol{q} = \boldsymbol{\Delta}$ *and zero, otherwise.*

Proof. 我们通过构造证明存在一个D_p =三维相对编码方案，能够产生所需的注意力概率。

由于注意力概率与输入张量X无关，我们设定$\boldsymbol{W}_{key} = \boldsymbol{W}_{qry} = \mathbf{0}$，这使得仅保留方程(8)的最后一项。将$\widehat{\boldsymbol{W}}_{key} \in \mathbb{R}^{D_k \times D_p}$设为恒等矩阵（并进行适当的行填充），可得到$\mathrm{A}_{\boldsymbol{q},\boldsymbol{k}} = \boldsymbol{v}^\top \boldsymbol{r}_{\boldsymbol{\delta}}$，其中$\boldsymbol{\delta} := \boldsymbol{k} - \boldsymbol{q}$。上文我们假设$D_p \leq D_k$，以确保$\boldsymbol{r}_{\boldsymbol{\delta}}$的信息不会丢失。

现在，假设我们可以写成：

$$\mathbf{A}_{\boldsymbol{q},\boldsymbol{k}} = -\alpha(\|\boldsymbol{\delta} - \boldsymbol{\Delta}\|^2 + c) \tag{14}$$

对于某个常数c。在上述表达式中，$\mathrm{A}_{\boldsymbol{q},:}$上的最大注意力分数为$-\alpha c$，且当$\mathrm{A}_{\boldsymbol{q},\boldsymbol{k}}$满足$\boldsymbol{\delta} = \boldsymbol{\Delta}$时达到该最大值。另一方面，$\alpha$系数可用于任意缩放$\mathrm{A}_{\boldsymbol{q},\boldsymbol{\Delta}}$与其他注意力分数之间的差异。

这样，对于$\boldsymbol{\delta} = \boldsymbol{\Delta}$，我们有

$$\begin{aligned}\lim_{\alpha\to\infty} \operatorname{softmax}(\mathbf{A}_{\boldsymbol{q},:})_{\boldsymbol{k}} &= \lim_{\alpha\to\infty} \frac{e^{-\alpha(\|\boldsymbol{\delta}-\boldsymbol{\Delta}\|^2+c)}}{\sum_{\boldsymbol{k}'} e^{-\alpha(\|(\boldsymbol{k}-\boldsymbol{q}')-\boldsymbol{\Delta}\|^2+c)}} \\ &= \lim_{\alpha\to\infty} \frac{e^{-\alpha\|\boldsymbol{\delta}-\boldsymbol{\Delta}\|^2}}{\sum_{\boldsymbol{k}'} e^{-\alpha\|(\boldsymbol{k}-\boldsymbol{q}')-\boldsymbol{\Delta}\|^2}} = \frac{1}{1 + \lim_{\alpha\to\infty}\sum_{\boldsymbol{k}'\neq\boldsymbol{k}} e^{-\alpha\|(\boldsymbol{k}-\boldsymbol{q}')-\boldsymbol{\Delta}\|^2}} = 1\end{aligned}$$

对于$\boldsymbol{\delta} \neq \boldsymbol{\Delta}$，方程变为$\lim_{\alpha\to\infty} \operatorname{softmax}(\mathrm{A}_{\boldsymbol{q},:})_{\boldsymbol{k}} = 0$,，恰好满足引理陈述所需。

剩下的就是证明存在$\boldsymbol{v}$和$\{\boldsymbol{r}_{\boldsymbol{\delta}}\}_{\boldsymbol{\delta}\in\mathbb{Z}^2}$使得等式(14)成立。展开该方程的右侧，我们得到$-\alpha(\|\boldsymbol{\delta} - \boldsymbol{\Delta}\|^2 + c) = -\alpha(\|\boldsymbol{\delta}\|^2 + \|\boldsymbol{\Delta}\|^2 - 2\langle\boldsymbol{\delta}, \boldsymbol{\Delta}\rangle + c)$.。现在如果我们设$\boldsymbol{v} = -\alpha\,(1, -2\boldsymbol{\Delta}_1, -2\boldsymbol{\Delta}_2)$和$\boldsymbol{r}_{\boldsymbol{\delta}} = (\|\boldsymbol{\delta}\|^2, \boldsymbol{\delta}_1, \boldsymbol{\delta}_2)$,，那么

$$\mathbf{A}_{\boldsymbol{q},\boldsymbol{k}} = \boldsymbol{v}^\top \boldsymbol{r}_{\boldsymbol{\delta}} = -\alpha(\|\boldsymbol{\delta}\|^2 - 2\boldsymbol{\Delta}_1\boldsymbol{\delta}_1 - 2\boldsymbol{\Delta}_2\boldsymbol{\delta}_2) = -\alpha(\|\boldsymbol{\delta}\|^2 - 2\langle\boldsymbol{\delta}, \boldsymbol{\Delta}\rangle) = -\alpha(\|\boldsymbol{\delta} - \boldsymbol{\Delta}\|^2 - \|\boldsymbol{\Delta}\|^2),$$

这与带有$c = -\|\boldsymbol{\Delta}\|^2$的方程(14)相匹配，证明至此完成。 □

关于α量级的说明。尽管随着α的增长，所有其他像素的注意力概率以指数速度收敛至0，但精确表示一个像素需要α (，或者矩阵$\boldsymbol{W}_{qry}$和$\boldsymbol{W}_{key}$)任意大。然而，实际实现总是依赖于有限精度算术，此时一个常数α足以满足我们的构造。例如，由于最小的正float32标量约为10^{-45}，设置α =为46即可获得硬注意力机制。

4 实验

本节旨在验证我们理论结果的适用性——即自注意力*can*执行卷积操作——并探究在实践中，自注意力层在标准图像分类任务训练下是否确实学会了像卷积层那样运作。特别是，我们研究了自注意力与卷积之间的关系，其中涉及*quadratic*和*learned*相对位置编码。我们发现，在这两种情况下，学习到的注意力概率往往遵循引理1的条件，这支持了我们的假设。

4.1 实现细节

我们研究了一个由六个多头自注意力层组成的全注意力模型。正如Bello等人（2019年）已证明的，将注意力特征与卷积特征相结合能提升在Cifar-100和ImageNet上的性能，因此我们并不追求达到最先进的性能表现。尽管如此，为验证我们的模型能学习到有意义的分类器，我们在CIFAR-10数据集（Krizhevsky等人）上将其与标准ResNet18（He等人，2015年）进行了对比。所有实验中，我们在输入端采用2×2可逆下采样（Jacobsen等人，2018年）以缩小图像尺寸。由于注意力系数张量（前向传播时存储）的大小随输入图像尺寸呈二次方增长，*full*注意力无法应用于较大图像。输入图像的固定尺寸表征通过最后一层表征的平均池化计算得出，并馈送至线性分类器。

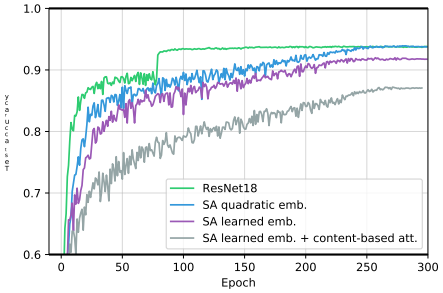

图2：CIFAR-10上的测试准确率。

Models	accuracy	# of params	# of FLOPS
ResNet18	0.938	11.2M	1.1B
SA quadratic emb.	0.938	12.1M	6.2B
SA learned emb.	0.918	12.3M	6.2B
SA learned emb. + content	0.871	29.5M	15B

表1：CIFAR-10上的测试准确率及模型大小。SA表示自注意力机制（Self-Attention）。

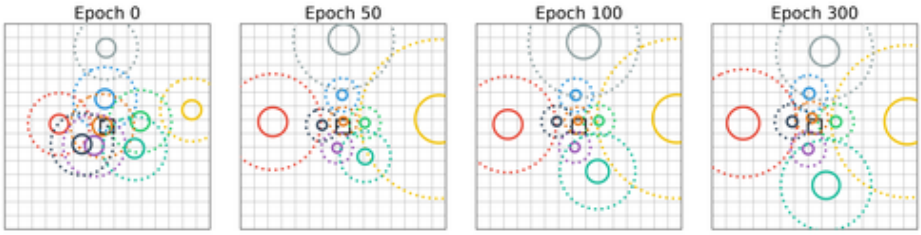

图3：在采用二次相对位置编码的训练过程中，第4层各注意力头（不同颜色）的关注中心。中央黑色方块为查询像素，实线和虚线圆圈则分别代表每个高斯分布的50%和90%百分位。

我们采用了PyTorch库（Paszke等人，2017年）并基于PyTorch Transformers[5]实现了我们的模型。代码已开源至Github[6]，超参数详见表2（附录）。

关于准确性的说明。为了验证我们的自注意力模型表现合理，我们在图6中展示了自注意力模型与小型ResNet（表1）在CIFAR-10数据集上300个训练周期内测试准确率的变化情况。ResNet收敛速度更快，但我们无法确定这是架构的固有特性还是所采用优化过程的产物。我们的实现可以进一步优化，以利用高斯注意力概率的局部性，显著减少浮点运算次数。我们观察到，基于内容的注意力学习嵌入训练难度更大，可能是由于参数数量增加所致。我们相信这一性能差距可以通过调整来匹配ResNet的表现，但这并非本工作的重点。

4.2 二次编码

作为第一步，我们的目标是验证，在引入方程(9)中的相对位置编码后，注意力层能够学会表现得像卷积层一样。我们在每一层训练九个注意力头，以匹配ResNet架构主要采用的3×3卷积核。每个注意力头h的中心被初始化为$\boldsymbol{\Delta}^{(h)} \sim \mathcal{N}(\mathbf{0}, 2\boldsymbol{I}_2)$。

图3展示了第4层中头部初始位置（不同颜色）在训练过程中的变化。可以看到，经过优化后，这些头部聚焦于图像上围绕查询像素形成的网格中的特定像素。我们关于自注意力机制应用于图像时能学习到以查询像素为中心的卷积滤波器的直觉得到了证实。

图4展示了训练结束时模型每一层所有注意力头的情况。可以看出，前几层的注意力头倾向于关注局部模式（第1、2层），而更深层（第3-6层）则通过将注意力中心定位在远离查询像素位置处，同时关注更大范围的模式。附录中还提供了更多注意力头（$N_h = 16$）的注意力位置分布图。图14显示了类似CNN的局部模式与长程依赖关系。值得注意的是，各注意力头互不重叠，呈现出一种最大化输入空间覆盖的排列方式。

[5] github.com/huggingface/pytorch-transformers
[6] github.com/epfml/attention-cnn

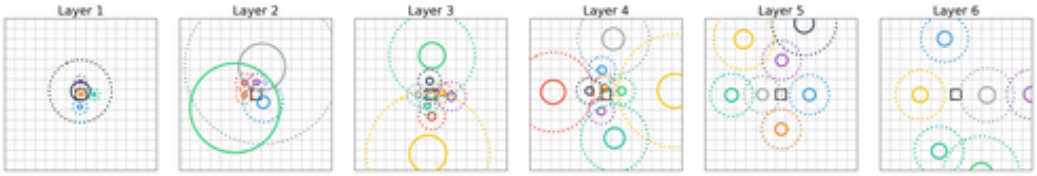

图4：使用二次位置编码的6个自注意力层中，各注意力头（不同颜色）的关注中心。中央黑色方块为查询像素，实线和虚线圆圈分别代表每个高斯分布的50%和90%百分位。

4.3 学习型相对位置编码

我们接着研究全注意力模型在图像上实际使用的位置编码。

我们采用了(Ramachandran等人，2019；Bello等人，2019)所使用的二维相对位置编码方案：为每个行和列像素位移学习一个$\lfloor D_p/2 \rfloor$维的位置编码向量。因此，键像素位于$\boldsymbol{k}$位置、查询像素位于$\boldsymbol{q}$位置的相对位置编码，是行位移嵌入$\boldsymbol{\delta}_1$与列位移嵌入$\boldsymbol{\delta}_2$ (的拼接，其中$\boldsymbol{\delta} = \boldsymbol{k} - \boldsymbol{q}$)。实验中我们设定$D_p = D_{out}$ =400。我们在以下几点上与他们（未发表的）实现有所不同：（i）我们未采用卷积主干和ResNet瓶颈结构进行下采样，而仅在输入端使用一个2×2可逆下采样层（Jacobsen等人，2018）；（ii）基于我们关于学习滤波器有效数量为min(D_h, D_{out})的理论，我们使用$D_h = D_{out}$而非$D_h = D_{out}/N_h$。

最初，我们舍弃输入数据，仅计算等式(8)最后一项作为注意力分数。图5展示了每一层每个头的注意力概率分布。该图证实了我们对于前两层的假设，并在一定程度上支持了第三层的情况：即便让模型从随机初始化的向量中自行学习位置编码方案，某些自注意力头（如左侧所示）仍学会了聚焦于单个像素点，这与引理1的条件高度吻合，进而符合定理1的结论。与此同时，其他注意力头则关注水平对称但非局部化的模式，以及长距离像素间的相互依赖关系。

我们转向一个更为现实的设定，其中注意力分数是通过位置和基于内容的注意力共同计算得出的（即(Ramachandran et al., 2019)中的$q^\top k + q^\top r$），这对应着一个成熟独立的自我注意力模型。

图6展示了每一层中每个注意力头的注意力概率分布。我们通过对100张测试图像批次的注意力概率进行平均，以勾勒出每个头的关注焦点，并消除对输入图像的依赖。我们的假设在第二层和第三层的部分注意力头上得到了验证：即便让模型从数据中自行学习编码，某些自注意力头仍仅利用基于位置的注意力机制，以固定的偏移量关注与查询像素不同的位置，从而复现了卷积核的感受野。其他注意力头则更多地运用基于内容的注意力（未平均概率见附录图8至10），发挥了自注意力相较于CNN的优势，这与我们的理论并不矛盾。实践中，Bello等人（2019）已证明，结合CNN与自注意力特征的表现优于单独使用任一方法。我们的实验表明，当优化一个无约束的全注意力模型时，这种组合方式会被自动习得。

卷积与多头自注意力之间的相似性在查询像素滑过图像时尤为显著：图6中可见的局部注意力模式会跟随查询像素移动。这一特性行为在图6与附录中图7（展示不同查询像素处的注意力概率）的对比中得以体现。第2和第3层的注意力模式不仅具有局部性，还与查询像素保持恒定偏移，类似于卷积核的感受野在图像上的滑动。这一现象在我们的交互式网站[7]上得到了清晰展示。该工具旨在探索不同图像在有或无基于内容的注意力情况下，各注意力组件的表现。我们相信，它是进一步理解MHSA如何学习处理图像的有力工具。

[7] epfml.github.io/attention-cnn

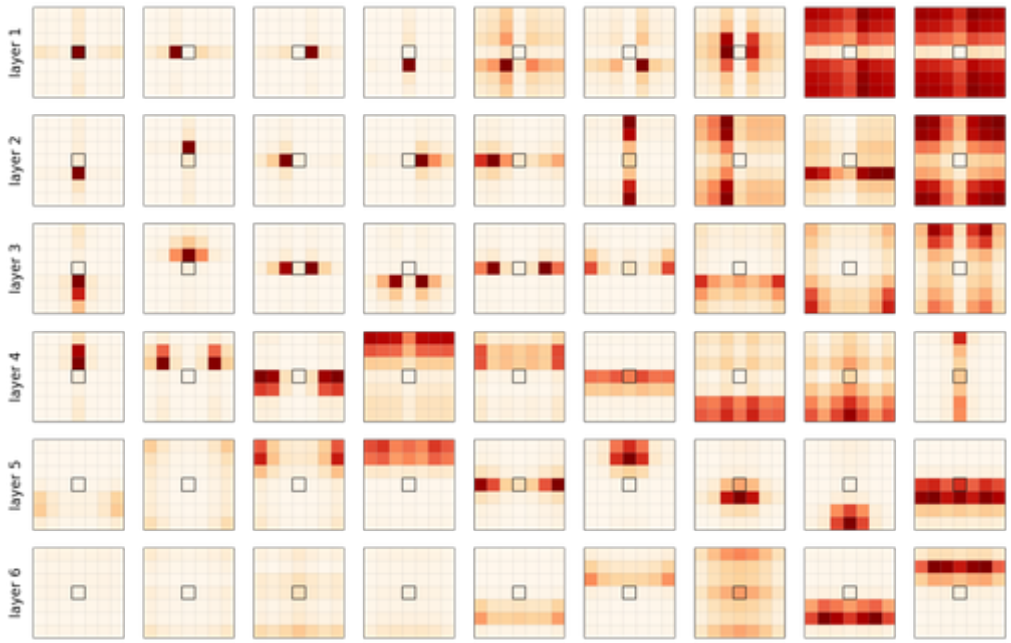

图5：使用学习到的相对位置编码（不含基于内容的注意力机制）时，各层(*row*)中每个注意力头(*column*)的注意力概率分布。中央黑色方块为查询像素点。为便于可视化，我们对注意力头进行了重新排序，并放大了查询像素周围7x7像素区域。

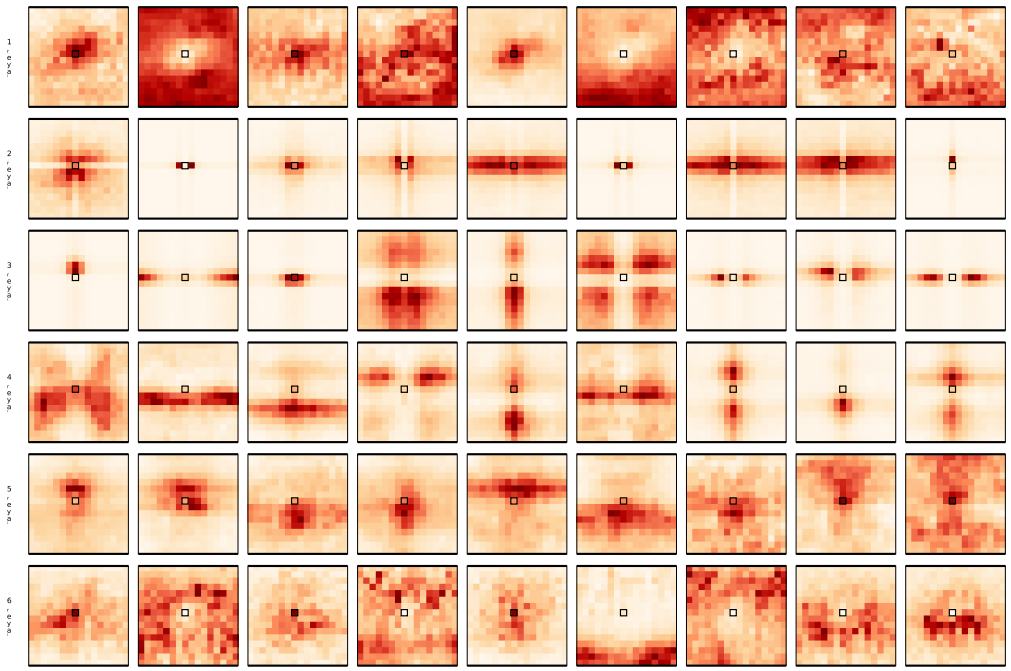

图6：采用学习型相对位置编码和基于内容-内容注意力机制的6层（*rows*）模型与9个头（*columns*）的注意力概率分布。注意力图通过对100张测试图像取平均值来展示头部行为，并消除对输入内容的依赖。黑色方块代表查询像素。更多示例见附录A。

5 相关工作

在本节中，我们回顾了CNN与transformer之间已知的差异与相似之处。

在文本处理中应用CNN网络——无论是基于词级别（Gehring等人，2017）还是字符级别（Kim，2014）——相比Transformer（或RNN）更为少见。Transformer与卷积模型在自然语言处理和神经机器翻译任务上已进行了广泛的实证比较。研究发现，Transformer在处理文本时较卷积模型更具竞争优势（Vaswani等人，2017）。直到近期，Bello等人（2019）和Ramachandran等人（2019）才将Transformer应用于图像领域，并证明其能达到与ResNet相当的精度。然而，他们的比较仅涵盖了性能、参数量及FLOPS，而未涉及表达能力。

除了比较Transformer与CNN的性能和计算成本外，关于这些架构表达能力的研究主要集中在它们捕捉长期依赖关系的能力上（Dai等人，2019）。另一项有趣的研究表明，Transformer具有图灵完备性（Dehghani等人，2018；Pérez等人，2019），这一理论成果虽重要，但对实践者而言信息量有限。据我们所知，我们首次证明了单层自注意力所表达的函数类包含所有卷积滤波器。

在弥合注意力机制与卷积之间差距的研究中，最接近的工作来自Andreoli（2019）。他们通过张量外积将注意力与卷积纳入统一框架。该框架中，卷积的感受野由“基”张量$A \in \mathbb{R}^{K\times K\times H\times W\times H\times W}$表示。例如，经典$K \times K$卷积核的感受野将通过$A_{\boldsymbol{\Delta},\boldsymbol{q},\boldsymbol{k}} = 1\{\boldsymbol{k}-\boldsymbol{q}=\boldsymbol{\Delta}\}$对$\boldsymbol{\Delta} \in \mathbb{\Delta}_K$进行编码。作者区分了这种*index-based*卷积与*content-based*卷积——后者中的A是根据输入值动态计算的（例如采用键/查询点积注意力机制）。我们的研究更进一步，提出了在输入内容中注入相对位置编码（实践中常用做法）的充分条件，使得*content-based*卷积能够表达任意*index-based*卷积。实验还表明，这种特性能够在实际训练中被习得。

6 结论

我们证明了应用于图像的自注意力层能够表达任何卷积层（在提供足够多头的情况下），并且全注意力模型能够学会根据输入内容结合局部行为（类似于卷积）与全局注意力。更广泛地说，全注意力模型似乎学习到了CNN的一种泛化形式，其中核模式与滤波器同时被学习——类似于可变形卷积（Dai等人，2017；Zampieri，2019）。未来工作的有趣方向包括将CNN丰富文献中的现有洞见转化回适用于各种数据模态（如图像、文本和时间序列）的Transformer模型。

致谢

Jean-Baptiste Cordonnier 感谢瑞士数据科学中心（SDSC）对本研究的资助。Andreas Loukas获得了瑞士国家科学基金会的支持（项目《图结构数据的深度学习》，资助编号 PZ00P2 179981）。

参考文献 Mart´n Abadi、Ashish Agarwal、Paul Barham、Eugene Brevdo、Zhifeng Chen、Craig Citro、Greg S. Corrado、Andy Davis、Jeffrey Dean、Matthieu Devin、Sanjay Ghemawat、Ian Goodfellow、Andrew Harp、Geoffrey Irving、Michael Isard、Yangqing Jia、Rafal Jozefowicz、Lukasz Kaiser、Manjunath Kudlur、Josh Levenberg、Dan Mané、Rajat Monga、Sherry Moore、Derek Murray、Chris Olah、Mike Schuster、Jonathon Shlens、Benoit Steiner、Ilya Sutskever、Kunal Talwar、Paul Tucker、Vincent Vanhoucke、Vijay Vasudevan、Fernanda Viégas、Oriol Vinyals、Pete Warden、Martin Wattenberg、Martin Wicke、Yuan Yu 和 Xiaoqiang Zheng。TensorFlow：异构系统上的大规模机器学习，2015年。软件可从 tensorflow.org 获取。Jean-Marc Andreoli。卷积、注意力与结构嵌入。*NeurIPS 2019 workshop on Graph Representation Learning, Dec 13, 2019, Vancouver, BC, Canada*，2019年。 Dzmitry Bahdanau、Kyunghyun Cho 和 Yoshua Bengio。通过联合学习对齐与翻译的神经机器翻译。见*3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*，2015年。 Irwan Bello、Barret Zoph、Ashish Vaswani、Jonathon Shlens 和 Quoc V. Le。注意力增强的卷积网络。*arXiv:1904.09925 [cs]*，2019年4月。 Jifeng Dai、Haozhi Qi、Yuwen Xiong、Yi Li、Guodong Zhang、Han Hu 和 Yichen Wei。可变形卷积网络。*CoRR*，abs/1703.06211，2017年。 Zihang Dai、Zhilin Yang、Yiming Yang、Jaime G. Carbonell、Quoc V. Le 和 Ruslan Salakhutdinov。Transformer-XL：超越固定长度上下文的注意力语言模型。*CoRR*，abs/1901.02860，2019年。 Mostafa Dehghani、Stephan Gouws、Oriol Vinyals、Jakob Uszkoreit 和 Lukasz Kaiser。通用Transformer。*CoRR*，abs/1807.03819，2018年。 Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。BERT：用于语言理解的深度双向Transformer预训练。*CoRR*，abs/1810.04805，2018年。 Jean-Yves Franceschi、Aymeric Dieuleveut 和 Martin Jaggi。多元时间序列的无监督可扩展表示学习。见*NeurIPS 2019*，2019年。 Jonas Gehring、Michael Auli、David Grangier、Denis Yarats 和 Yann N. Dauphin。卷积序列到序列学习。*CoRR*，abs/1705.03122，2017年。 Kaiming He、Xiangyu Zhang、Shaoqing Ren 和 Jian Sun。深度残差学习用于图像识别。*CoRR*，abs/1512.03385，2015年。 Sepp Hochreiter 和 Jürgen Schmidhuber。长短期记忆。*Neural Computation*，9(8): 1735–1780，1997年。 Jie Hu、Li Shen 和 Gang Sun。挤压与激励网络。见*2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*，第7132–7141页，2018年。

Jrn-Henrik Jacobsen、Arnold W.M. Smeulders 和 Edouard Oyallon。i-revnet：深度可逆网络。载于*International Conference on Learning Representations*，2018年。

Yoon Kim. 用于句子分类的卷积神经网络. 见 *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 第1746–1751页, 卡塔尔多哈, 2014年10月. 计算语言学协会. doi: 10.3115/v1/D14-1181.

亚历克斯·克里泽夫斯基（Alex Krizhevsky）、维诺德·奈尔（Vinod Nair）与杰弗里·辛顿（Geoffrey Hinton）。CIFAR-10（加拿大高等研究院）。

亚当·帕兹克、萨姆·格罗斯、苏米特·钦塔拉、格雷戈里·查南、爱德华·杨、扎卡里·德维托、林泽明、阿尔班·德斯梅森、卢卡·安蒂加和亚当·勒雷尔。PyTorch中的自动微分。载于*NIPS Autodiff Workshop*，2017年。

豪尔赫·P雷斯、哈维尔·马林科维奇和巴勃罗·巴塞洛。论现代神经网络架构的图灵完备性。*CoRR*, abs/1901.03429, 2019。

亚历克·拉德福德、杰弗里·吴、雷文·柴尔德、大卫·栾、达里奥·阿莫迪和伊利亚·苏茨克弗。语言模型是无监督多任务学习者。2018。

Prajit Ramachandran、Niki Parmar、Ashish Vaswani、Irwan Bello、Anselm Levskaya与Jonathon Shlens。视觉模型中的独立自注意力机制。*CoRR*，abs/1906.05909，2019年。

Aron van den Oord、Sander Dieleman、Heiga Zen、Karen Simonyan、Oriol Vinyals、Alexander Graves、Nal Kalchbrenner、Andrew Senior 和 Koray Kavukcuoglu。WaveNet：一种原始音频生成模型。*arXiv preprint arXiv:1609.03499*，2016。

阿希什·瓦斯瓦尼、诺姆·沙泽尔、尼基·帕尔马、雅各布·乌兹科雷特、利昂·琼斯、艾丹·N·戈麦斯、卢卡什·凯泽和伊利亚·波洛苏欣。《注意力就是你所需要的一切》。*CoRR*，abs/1706.03762，2017年。

王晓龙、Ross B. Girshick、Abhinav Gupta与何恺明。非局部神经网络。载于*2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*，第7794–7803页，2018年。

杨志林、戴子航、杨一鸣、Jaime G. Carbonell、Ruslan Salakhutdinov和Quoc V. Le。XLNet：面向语言理解的广义自回归预训练。*CoRR*，abs/1906.08237，2019年。

卢卡·赞皮耶里。面向体积计算流体力学的几何深度学习。第67页，2019年。

附录

更多基于内容注意力的示例

我们展示了更多由自注意力模型计算得出的注意力概率示例。图7展示了与图6不同查询像素的平均注意力分布。图8至图10则展示了针对单张图像的注意力情况。

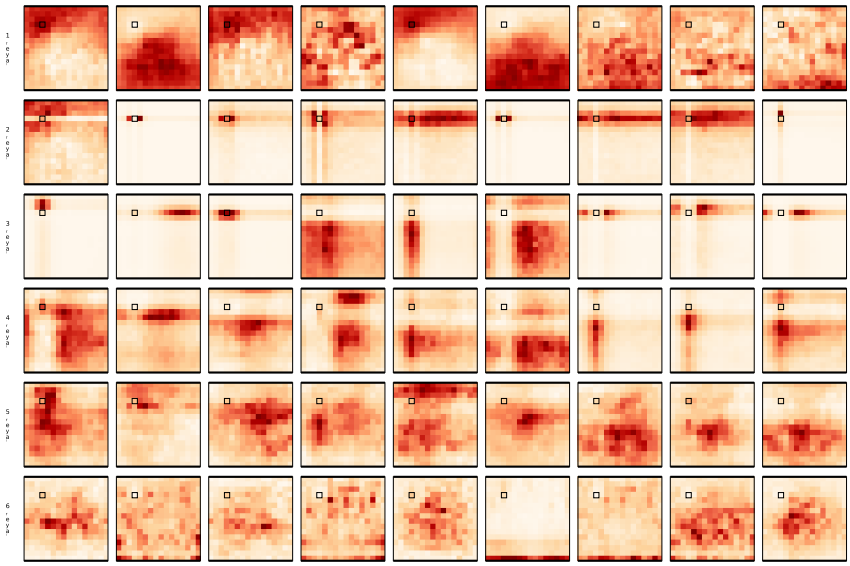

图7：使用学习到的相对位置编码和内容-内容注意力机制的6层（*rows*）模型及9个头（*columns*）的注意力概率分布。展示的是100张测试图像的平均结果。黑色方块代表查询像素。

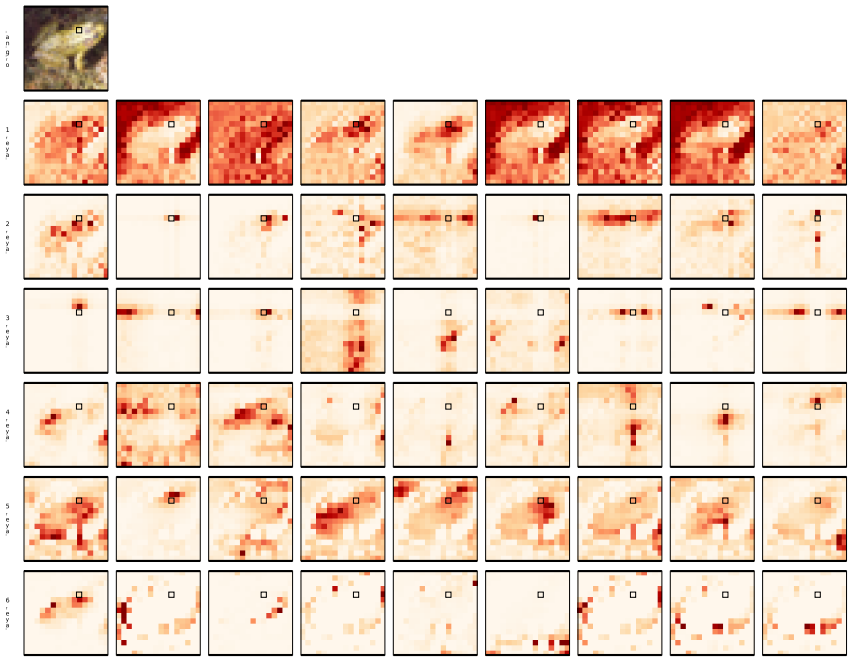

图8：使用学习到的相对位置编码和基于内容-内容的注意力机制，具有6层（*rows*）和9个头（*columns*）的模型的注意力概率分布。查询像素（黑色方块）位于青蛙头部。

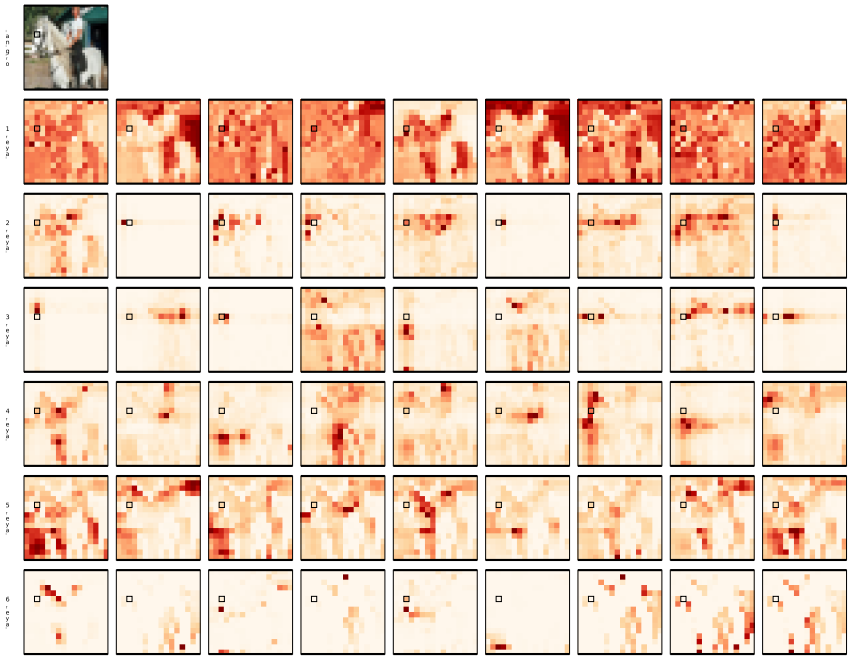

图9：使用学习到的相对位置编码和基于内容-内容注意力机制的模型（6层*rows*，9个头*columns*）的注意力概率分布。查询像素（黑色方块）位于马头位置。

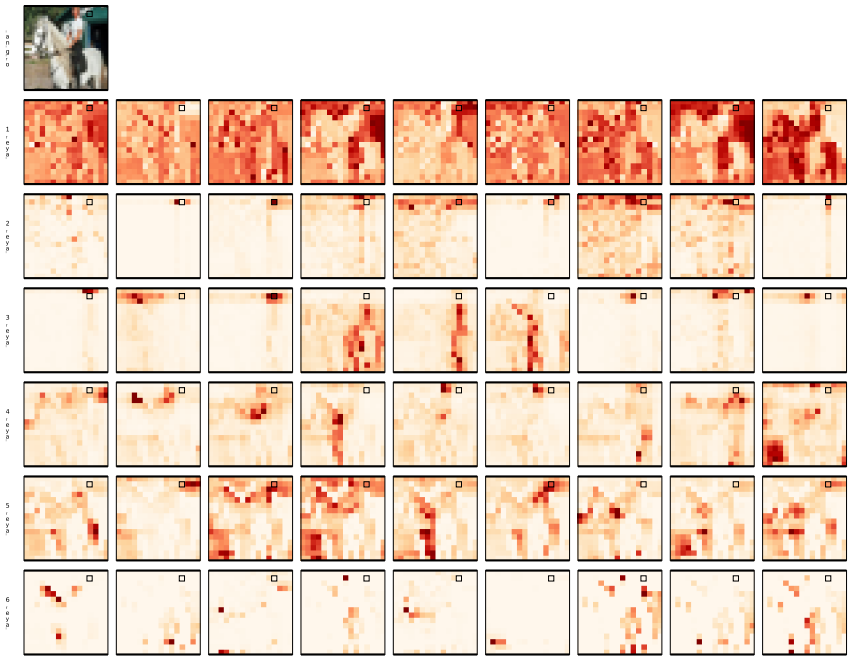

图10：使用学习到的相对位置编码和基于内容-内容注意力机制的6层（*rows*）9头（*columns*）模型的注意力概率分布。查询像素（黑色方块）位于背景中的建筑物上。

B 我们实验中使用的超参数

Hyper-parameters	
number of layers	6
number of heads	9
hidden dimension	400
intermediate dimension	512
invertible pooling width	2
dropout probability	0.1
layer normalization epsilon	10^{-12}
number of epochs	300
batch size	100
learning rate	0.1
weight decay	0.0001
momentum	0.9
cosine decay	✓
linear warm up ratio	0.05

表2：自注意力网络参数

C 位置编码参考文献

Model	type of positional encoding			relative
	sinusoids	learned	quadratic	
Vaswani et al. (2017)	✓			
Radford et al. (2018)		✓		
Devlin et al. (2018)		✓		
Dai et al. (2019)	✓			✓
Yang et al. (2019)	✓			✓
Bello et al. (2019)		✓		✓
Ramachandran et al. (2019)		✓		✓
Our work		✓	✓	✓

表3：应用于文本（*top*）和图像（*bottom*）的Transformer模型所使用的位置编码类型。当尝试过多种编码类型时，我们报告作者推荐的那一种。

D 广义引理 1

我们提出了引理1的一个推广，它用一个更温和的假设取代了对单一像素的硬注意力的必要性：注意力概率应覆盖网格感受野。引理2仍满足该引理的条件，因此定理1随之成立。

引理3. *Consider a multi-head self-attention layer consisting of* $N_h \geq K^2$ *heads,* $D_h \geq D_{out}$ *and let* $\omega : [H] \times [W] \to [HW]$ *be a pixel indexing. Then, for any convolutional layer with a* $K \times K$ *kernel and* D_{out} *output channels, there exists* $\{\boldsymbol{W}_{val}^{(h)}\}_{h \in [N_h]}$ *and* $\boldsymbol{W}_{out}$ *such that* MHSA(X) = 卷积(X) *for every* $\mathrm{X} \in \mathbb{R}^{W \times H \times D_{in}}$ *if and only if, for all* $\boldsymbol{q} \in [H] \times [W]$, [8]

$$\mathrm{span}(\{\boldsymbol{e}_{\omega(\boldsymbol{q}+\boldsymbol{\Delta})} \in \mathbb{R}^{HW} : \boldsymbol{\Delta} \in \mathbb{\Delta}_K\}) \subseteq \mathrm{span}(\{\mathrm{vect}(\mathrm{softmax}(\mathbf{A}_{\boldsymbol{q},:}^{(h)})) : h \in [N_h]\}) \,.$$

[8] the vectorization operator $\mathrm{vect}(\cdot)$ flattens a matrix into a vector

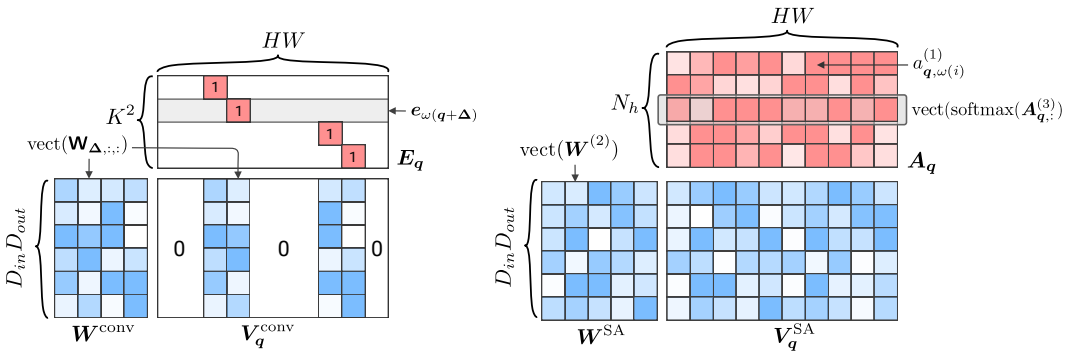

图11：向量化权重矩阵$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$和$\boldsymbol{V}_{\boldsymbol{q}}^{\text{SA}}$的分解，用于计算维度为$H \times W$的输入图像在位置$\boldsymbol{q}$处的输出。在*left*上：核大小为2×2的卷积，在*right*上：具有N_h =5个头的自注意力机制。两种情况下D_{in} =2，D_{out} =3。

Proof. 我们的第一步将是重新表述多头自注意力算子，从方程(1)和方程(4)出发，使得多头机制的效果更加透明：

$$\text{MHSA}(\mathbf{X}) = \boldsymbol{b}_{out} + \sum_{h\in[N_h]} \text{softmax}(\mathbf{A}^{(h)})\mathbf{X}\underbrace{\boldsymbol{W}_{val}^{(h)}\boldsymbol{W}_{out}[(h-1)D_h+1:hD_h+1]}_{\boldsymbol{W}^{(h)}} \tag{15}$$

需要注意的是，每个注意力头的值矩阵$\boldsymbol{W}_{val}^{(h)} \in \mathbb{R}^{D_{in}\times D_h}$以及维度为$D_h \times D_{out}$的投影矩阵$\boldsymbol{W}_{out}$的每个块都是通过学习得到的。假设$D_h \geq D_{out}$，我们可以将每对矩阵替换为每个头的一个学习矩阵$\boldsymbol{W}^{(h)}$。为了简化起见，我们考虑多头自注意力的一个输出像素，并忽略偏置项：

$$\text{MHSA}(\mathbf{X})_{\boldsymbol{q},:} = \sum_{h\in[N_h]}\Big(\sum_{\boldsymbol{k}} a_{\boldsymbol{q},\boldsymbol{k}}^{(h)}\mathbf{X}_{\boldsymbol{k},:}\Big)\boldsymbol{W}^{(h)} = \sum_{\boldsymbol{k}}\mathbf{X}_{\boldsymbol{k},:}\underbrace{\Big(\sum_{h\in[N_h]} a_{\boldsymbol{q},\boldsymbol{k}}^{(h)}\boldsymbol{W}^{(h)}\Big)}_{\boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{SA}}\in\mathbb{R}^{D_{in}\times D_{out}}}, \tag{16}$$

使用$a_{\boldsymbol{q},\boldsymbol{k}}^{(h)} = \text{softmax}(\text{A}_{\boldsymbol{q},:}^{(h)})_{\boldsymbol{k}}$。我们以同样的方式重写像素$\boldsymbol{q}$处的卷积输出：

$$\text{Conv}(\mathbf{X})_{\boldsymbol{q},:} = \sum_{\boldsymbol{\Delta}\in\mathbb{\Delta}_K}\mathbf{X}_{\boldsymbol{q}+\boldsymbol{\Delta},:}\mathbf{W}_{\boldsymbol{\Delta},:,:} = \sum_{\boldsymbol{k}\in[H]\times[W]}\mathbf{X}_{\boldsymbol{k},:}\underbrace{\mathbb{1}_{\{\boldsymbol{k}-\boldsymbol{q}\in\mathbb{\Delta}_K\}}\mathbf{W}_{\boldsymbol{k}-\boldsymbol{q},:,:}}_{\boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{conv}}\in\mathbb{R}^{D_{in}\times D_{out}}}. \tag{17}$$

当且仅当每一对键/查询像素的线性变换相等时，即$\boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{conv}} = \boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{SA}}\ \forall \boldsymbol{q},\boldsymbol{k}$，方程(16)与(17)对于任意输入X的等式成立。我们将权重矩阵向量化为维度$D_{in}D_{out} \times HW$的矩阵，即$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$:=[向量化$(\boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{conv}})]_{\boldsymbol{k}\in[H]\times[W]}$和$\boldsymbol{V}_{\boldsymbol{q}}^{\text{SA}}$:=[向量化$(\boldsymbol{W}_{\boldsymbol{q},\boldsymbol{k}}^{\text{SA}})]_{\boldsymbol{k}\in[H]\times[W]}$。因此，要证明对于所有X都有Conv(X)=MHSA(X)，就必须证明对于所有$\boldsymbol{q}$都有$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}} = \boldsymbol{V}_{\boldsymbol{q}}^{\text{SA}}$。

矩阵$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$具有受限的支持：只有与像素$\boldsymbol{q}$感受野中的像素偏移$\boldsymbol{\Delta} \in \mathbb{\Delta}_K$相关的列可以非零。这导致了图11中展示的因子分解$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}} = \boldsymbol{W}^{\text{conv}}\boldsymbol{E}_{\boldsymbol{q}}$，其中$\boldsymbol{W}^{\text{conv}} \in \mathbb{R}^{D_{in}D_{out}\times K^2}$和$\boldsymbol{E}_{\boldsymbol{q}} \in \mathbb{R}^{K^2\times HW}$。给定由$j$索引的偏移$\boldsymbol{\Delta} \in \mathbb{\Delta}_K$的排序，设$(\boldsymbol{W}^{\text{conv}})_{:,j}$ =为向量化$\text{W}_{\boldsymbol{\Delta},:,:}$，以及$(\boldsymbol{E}_{\boldsymbol{q}})_{j,:} = \boldsymbol{e}_{\omega(\boldsymbol{q}+\boldsymbol{\Delta})}$。另一方面，我们分解$\boldsymbol{V}_{\boldsymbol{q}}^{\text{SA}} = \boldsymbol{W}^{\text{SA}}\boldsymbol{A}_{\boldsymbol{q}}$，令$(\boldsymbol{W}^{\text{SA}})_{:,h}$ =为向量化$\boldsymbol{W}^{(h)}$，且$(\boldsymbol{A}_{\boldsymbol{q}})_{h,i} = a_{\boldsymbol{q},\omega(i)}^{(h)}$。

证明的结论在于表明，row$(\boldsymbol{E}_{\boldsymbol{q}}) \subseteq$ row$(\boldsymbol{A}_{\boldsymbol{q}})$ 是存在一个$\boldsymbol{W}^{\text{SA}}$使得任意$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}} = \boldsymbol{W}^{\text{conv}}\boldsymbol{E}_{\boldsymbol{q}}$可表示为$\boldsymbol{W}^{\text{SA}}\boldsymbol{A}_{\boldsymbol{q}}$的充分必要条件。

充分。给定行$(\boldsymbol{E}_{\boldsymbol{q}}) \subseteq$ 行$(\boldsymbol{A}_{\boldsymbol{q}})$，存在$\Phi \in \mathbb{R}^{K^2\times N_h}$使得$\boldsymbol{E}_{\boldsymbol{q}} = \Phi\boldsymbol{A}_{\boldsymbol{q}}$，且有效分解为$\boldsymbol{W}^{\text{SA}} = \boldsymbol{W}^{\text{conv}}\Phi$，从而得到$\boldsymbol{W}^{\text{SA}}\boldsymbol{A}_{\boldsymbol{q}} = \boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$。

必要。假设存在$\boldsymbol{x} \in \mathbb{R}^{HW}$使得$\boldsymbol{x} \in$行$(\boldsymbol{E}_{\boldsymbol{q}})$且$\boldsymbol{x} \notin$行$(\boldsymbol{A}_{\boldsymbol{q}})$，并设$\boldsymbol{x}^\top$为$\boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$的一行。那么，对于任何$\boldsymbol{W}^{\text{SA}}$都有$\boldsymbol{W}^{\text{SA}}\boldsymbol{A}_{\boldsymbol{q}} \neq \boldsymbol{V}_{\boldsymbol{q}}^{\text{conv}}$，且不存在可能的分解。

□

E 广义二次位置编码

我们注意到二次位置编码（第3节）中的注意力概率与有界支撑的各向同性双变量高斯分布的相似性：

$$\text{softmax}(\mathbf{A}_{\boldsymbol{q},:})_{\boldsymbol{k}} = \frac{e^{-\alpha\|(\boldsymbol{k}-\boldsymbol{q})-\boldsymbol{\Delta}\|^2}}{\sum_{\boldsymbol{k}'\in[W]\times[H]} e^{-\alpha\|(\boldsymbol{k}'-\boldsymbol{q})-\boldsymbol{\Delta}\|^2}} \,. \tag{18}$$

基于这一观察，我们进一步将注意力机制扩展到像素位置上的非各向同性高斯分布。每个注意力头由一个关注中心$\boldsymbol{\Delta}$和一个协方差矩阵$\boldsymbol{\Sigma}$参数化，从而得到以下注意力分数，

$$\mathbf{A}_{\boldsymbol{q},\boldsymbol{k}} = -\frac{1}{2}(\boldsymbol{\delta}-\boldsymbol{\Delta})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{\delta}-\boldsymbol{\Delta}) = -\frac{1}{2}\boldsymbol{\delta}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta} + \boldsymbol{\delta}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta} - \frac{1}{2}\boldsymbol{\Delta}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta}\,, \tag{19}$$

其中，再次强调，$\boldsymbol{\delta} = \boldsymbol{k} - \boldsymbol{q}$。由于softmax具有平移不变性，最后一项可以舍弃，因此我们将注意力系数重写为头部目标向量$\boldsymbol{v}$与相对位置编码$\boldsymbol{r}_{\boldsymbol{\delta}}$ (之间的点积，后者由像素偏移$\boldsymbol{\delta}$)的一阶和二阶组合构成：

$$\boldsymbol{v} = \frac{1}{2}(2(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta})_1, 2(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Delta})_2, -\boldsymbol{\Sigma}^{-1}_{1,1}, -\boldsymbol{\Sigma}^{-1}_{2,2}, -2\cdot\boldsymbol{\Sigma}^{-1}_{1,2})^\top \text{ and } \boldsymbol{r}_{\boldsymbol{\delta}} = (\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \boldsymbol{\delta}_1^2, \boldsymbol{\delta}_2^2, \boldsymbol{\delta}_1\boldsymbol{\delta}_2)^\top \,.$$

评估。我们使用这种广义的二次相对位置编码训练了模型。我们好奇的是，采用上述编码后，自注意力模型是否会学习关注非各向同性的像素群——从而形成CNN中未见过的模式。每个注意力头通过$\boldsymbol{\Delta} \in \mathbb{R}^2$和$\boldsymbol{\Sigma}^{-1/2} \in \mathbb{R}^{2\times2}$进行参数化，以确保协方差矩阵保持半正定。我们将注意力中心初始化为$\boldsymbol{\Delta}^{(h)} \sim \mathcal{N}(\boldsymbol{0}, 2\boldsymbol{I}_2)$和$\boldsymbol{\Sigma}^{-1/2} = \boldsymbol{I}_2 + \mathcal{N}(\boldsymbol{0}, 0.01\boldsymbol{I}_2)$，使得初始注意力概率接近各向同性高斯分布。图12显示，网络确实学会了非各向同性的注意力概率模式，尤其是在高层。然而，未获得任何性能提升的事实似乎表明，注意力的非各向同性在实践中并无特别助益——二次位置编码已足够。

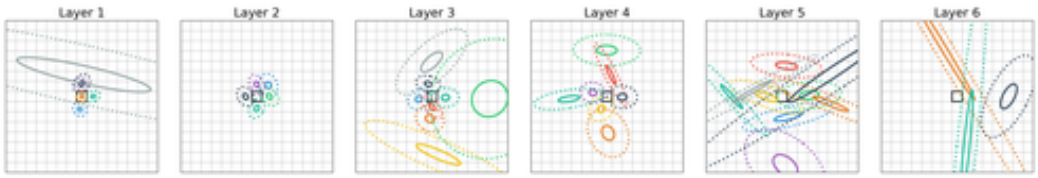

图12：采用非各向同性高斯参数化的6个自注意力层中，各注意力头（不同颜色）的关注中心。中央黑色方块为查询像素，实线和虚线圆圈则分别代表每个高斯的50%和90%百分位。

剪枝退化注意力头。部分非各向同性的注意力头会聚焦于“非直观”的像素区域：要么关注极窄的像素条带（当$\boldsymbol{\Sigma}^{-1}$几乎奇异时），要么均匀关注所有像素（当$\boldsymbol{\Sigma}^{-1}$接近$\boldsymbol{0}$ (即注意力分数恒定)）。我们不禁思考：这类注意力模式对模型真的有用，还是这些头已退化且未被利用？为探明真相，我们剪除了所有最大特征值小于10^{-5}或条件数（最大与最小特征值之比）大于10^5的注意力头。具体到我们6层9头结构的模型中，从首层至末层分别剪除了[2, 4, 1, 2, 6, 0]个头。这意味着这些层无法再表达3×3的卷积核。如图2黄色标注所示，该剪枝操作初期会轻微影响性能（可能源于偏移偏差），但在以十分之一学习率继续训练数轮后，准确率恢复至剪枝前水平。由此，我们在不牺牲性能的前提下，将参数量与浮点运算量缩减了四分之一。

F 增加头数

为了完整性　ss，我们还测试了增加架构的注意力头数量{v*}　　课程从9点到16点。

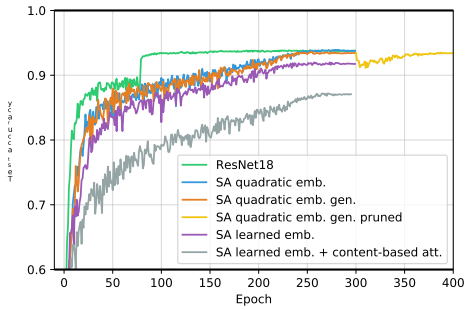

图13：CIFAR-10上测试准确率的演变过程。剪枝后的模型(*yellow*)为非各向同性模型(*orange*)的持续训练结果。

Models	accuracy	# of params	# of FLOPS
ResNet18	0.938	11.2M	1.1B
SA quadratic emb.	0.938	12.1M	6.2B
SA quadratic emb. gen.	0.934	12.1M	6.2B
SA quadratic emb. gen. pruned	0.934	9.7M	4.9B
SA learned emb.	0.918	12.3M	6.2B
SA learned emb. + content	0.871	29.5M	15B

表4：各模型在CIFAR-10上的参数量与准确率。SA表示自注意力机制（Self-Attention）。

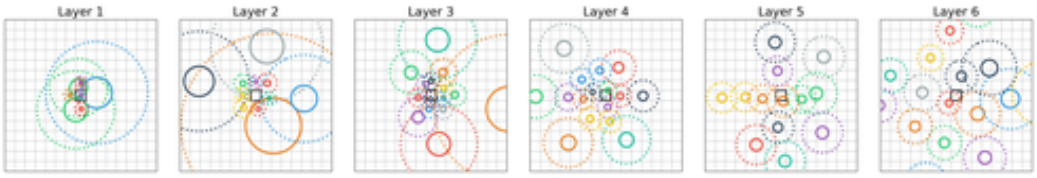

图14：使用二次位置编码的6个自注意力层中，16个注意力头（不同颜色）的关注中心。中央黑色方块为查询像素，实线和虚线圆圈分别表示每个高斯的50%和90%百分位。

与图4类似，我们观察到网络区分出两种主要的注意力模式。局部化注意力头（即那些关注近乎单个像素的头）在最初几层中出现更为频繁。自注意力层利用这些头以类似于卷积层的方式运作。而在更高层级中，非局部化注意力的头则变得更为普遍。