Deep Dive into Transformer Architecture

1 Architectural Foundations

1.1 The Architectural of GPT-2

我们以GPT-2的**模型架构**为切入，分析整个Transformer Block的**结构**及其**内在机制**。GPT-2的架构是在GPT-1的基础上改进的，而GPT-1的模型架构则是拿掉了Multi-Head Cross Attention （多头交叉注意力），只保留了Masked Multi-Head Self-Attention的**Transformer的解码器**。GPT-2的模型架构在GPT-1的基础上做了如下改进：

- Layer normalization被移动到每一个sub-block（两个子层：**解码器自注意力**与**基于位置的前馈神经网络**）的**输入**位置，类似于一个**预激活**的残差网络。同时在**最后的**自注意力块后添加一个额外的layer normalization。
- 采用一种改进的初始化方法，该方法考虑了残差路径与模型深度的累积。在初始化阶段使用缩放因子 $\frac{1}{\sqrt{N}}$ 对residual layer的权重进行缩放操作，其中 N 为residual layer的数量（深度）。
- 字典大小设置为50257；无监督预训练可看到的上下文的 context 由512扩展为1024；Batch Size大小调整为512。

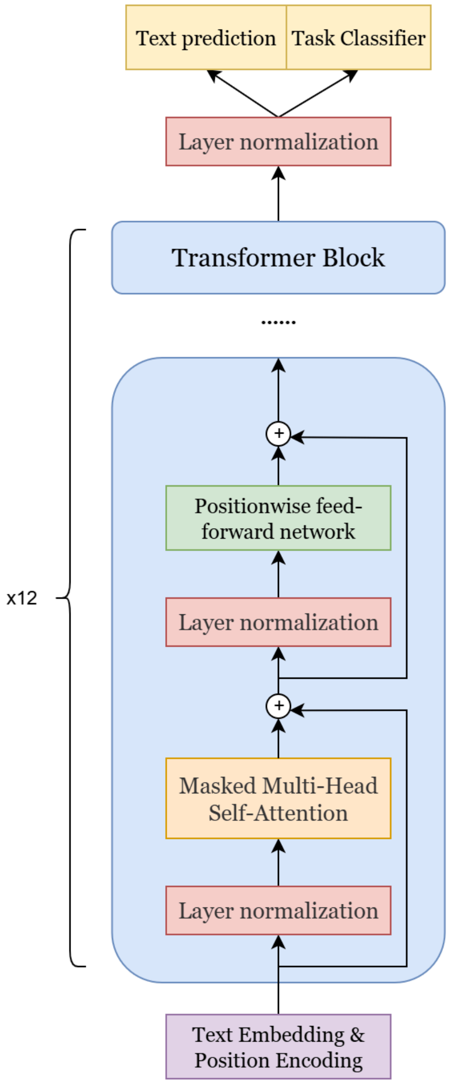


图1 GPT-2的Transformer Block

1.2 Transformer Block 构成分析

假设我们的输入为一个 $X \in \mathbb{R}^{n\times d}, \quad x_i^\top \in \mathbb{R}^{1\times d}\ (i = 1,\dots,n)$ 的矩阵，其中每一行为一个token的表征向量，长度为d（相当于经过了embedding与position encoding操作），接下来它将经过layer normalization、Multi-Head Self-Attention、Position-wise Feed-Forward Networks等计算操作，我们一个一个来分析。

1.2.1 Layer Normalization

层归一化按**行**（即每个 token）归一化，对同一token的全部特征做**零均值**、**单位方差**处理，公式如下（对于第i个token有）：

$$\mu_i = \frac{1}{d}\mathbf{1}^\top x_i \in \mathbb{R}, \quad \sigma_i \ = \ \left(\tfrac{1}{d}\|x_i - \mu_i\mathbf{1}\|_2^2\right)^{1/2} \in \mathbb{R}, \qquad \hat{x}_i = \frac{x_i - \mu_i\mathbf{1}}{\sigma_i + \varepsilon} \in \mathbb{R}^{d\times 1},$$

堆叠得到：

$$\hat{X} = (X - \mu\mathbf{1}_d^\top) \oslash (\sigma\mathbf{1}_d^\top + \varepsilon),$$

其中，$\mu \in \mathbb{R}^{n\times 1}$，$\sigma \in \mathbb{R}^{n\times 1}$均为堆叠而成的**向量**；；$\oslash$为**Hadamard除**（矩阵逐元素相除）；$\mathbf{1}_d \in \mathbb{R}^{d\times 1}$为全1列向量；$\varepsilon \in \mathbb{R}^{n\times d}$，用于维持数值稳定。最后加上仿射变换的结果为：

$$\mathrm{LN}(X) = \hat{X} \odot \gamma^\top + \beta^\top, \quad \gamma, \beta \in \mathbb{R}^d.$$

以下为Layer Normalization的优点：**只依赖行内统计**（不需存储/维护全局运行均值与方差；只对最后一维做并行归约），与 batch size 无关，因此测试与训练过程完全一致。同时，LN 能减小层输入**尺度漂移**（internal covariate shift），在注意力与残差结构叠加时，能保持梯度在深f网络中有效传播，加速收敛。

Transformer 在**小批量甚至序列长度为 1 的自回归推断**场景中尤为常见，LN**仅涉及当前 token 向量本身**，推断时与训练时的分布完全对齐，无需像 BatchNorm 那样维护滑动均值，也不会出现 batch 幅度微抖动导致的生成质量劣化问题。

从几何视角来看，LN的操作是将所有的**token投影到超球面**中：

1. **平移**：$x \mapsto x - \mu_i$ —— 消除径向偏移；
2. **径向缩放**：除以 σ —— 投影到半径 1 的球面；
3. **各向异性伸缩**：$\odot\gamma$ —— 把球面拉成椭球，提供可学习尺度。

因此 LN 把每个 token 的**向量表示**都 **压到同一"球壳"**（或椭球壳）上；后续注意力仅关心 **方向信息**，点积 $\langle q_i, k_j\rangle$ 规模始终 $\mathcal{O}(1)$，softmax区间稳定。

1.2.2 Multi-Head Self-Attention

首先考虑标准的缩放点积注意力机制，对于任一头(head) h，可以得到投影矩阵:

$$Q_h = XW_Q^{(h)}, K_h = XW_K^{(h)}, V_h = XW_V^{(h)}, \quad W_{Q,K,V}^{(h)} \in \mathbb{R}^{d\times d_h},$$

其中 d_h 为**query/key空间**的维度，一般远小于**嵌入空间**的维度 d 。投影矩阵的作用是将**嵌入(Embedding)空间**中的token**映射**到**较小**的**查询、键、值空间**中的某个方向。当键与查询的方向相对齐时，就能认为他们相匹配（高度对齐）。

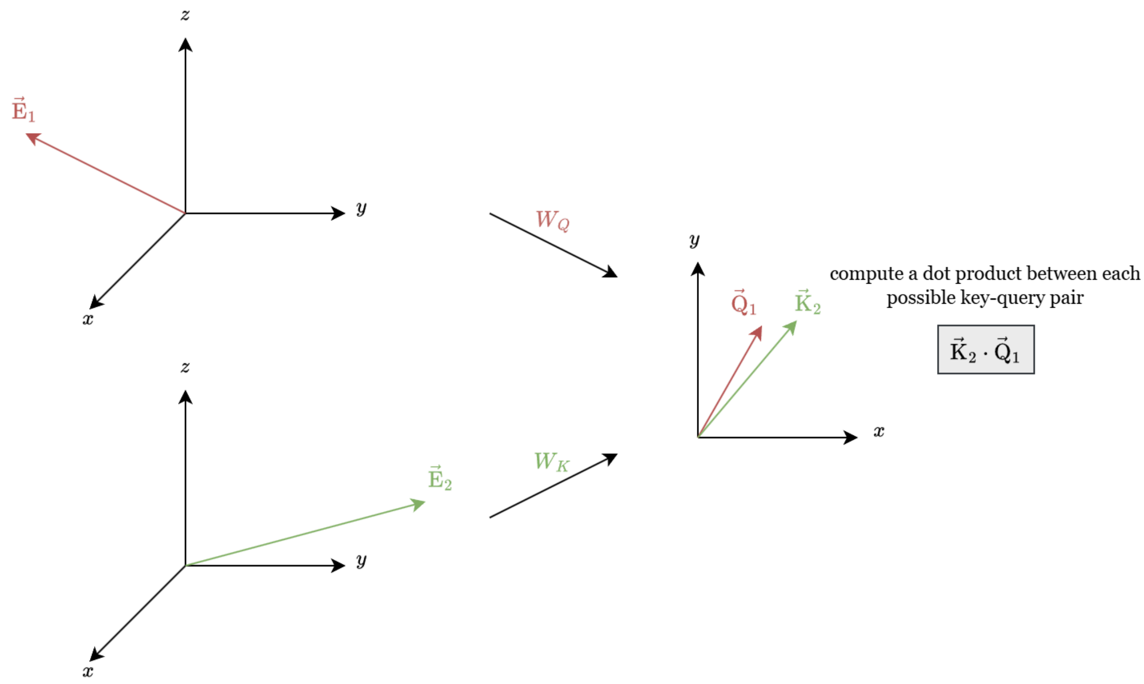


图2 查询向量与键向量的几何关系示意图，其中左图为嵌入空间(Embedding)，右图则为查询/键(query/key)空间

未加掩码时的**单头注意力权重**为：

$$S_h = \frac{Q_h K_h^\top}{\sqrt{h}} \in \mathbb{R}^{n \times n}, \quad A_h = \mathrm{softmax}(S_h).$$

$Q_h K_h^\top$ 结果的**每个元素**都可以看作一对**键—查询对**之间的点积，根据点积的概念，可以容易看出**值越大**说明键与查询越**对齐**。同时，为了维持数值稳定性，所有点积的结果都会除以**键—查询空间维度的平方根。**

由于 softmax 逐行作用，A_h 的每一行都是一组**概率分布** (行随机矩阵)，每个元素都是一个注意力权重，表示一对键与查询向量之间的相关度。

GPT的本质还是一个**自回归的语言模型**，在预测阶段，其输出序列的词元是逐个生成的，因此同样需要**掩码**操作（这也是为什么它采用的是解码器，而**BERT**作为**双向**编码器无需掩码操作），即需要保证第 i 个 token 不能看到序列中位置 $j > i$ 的信息，令：

$$M_{ij} = \begin{cases} 0, & j \le i \\ -\infty, & j > i \end{cases}.$$

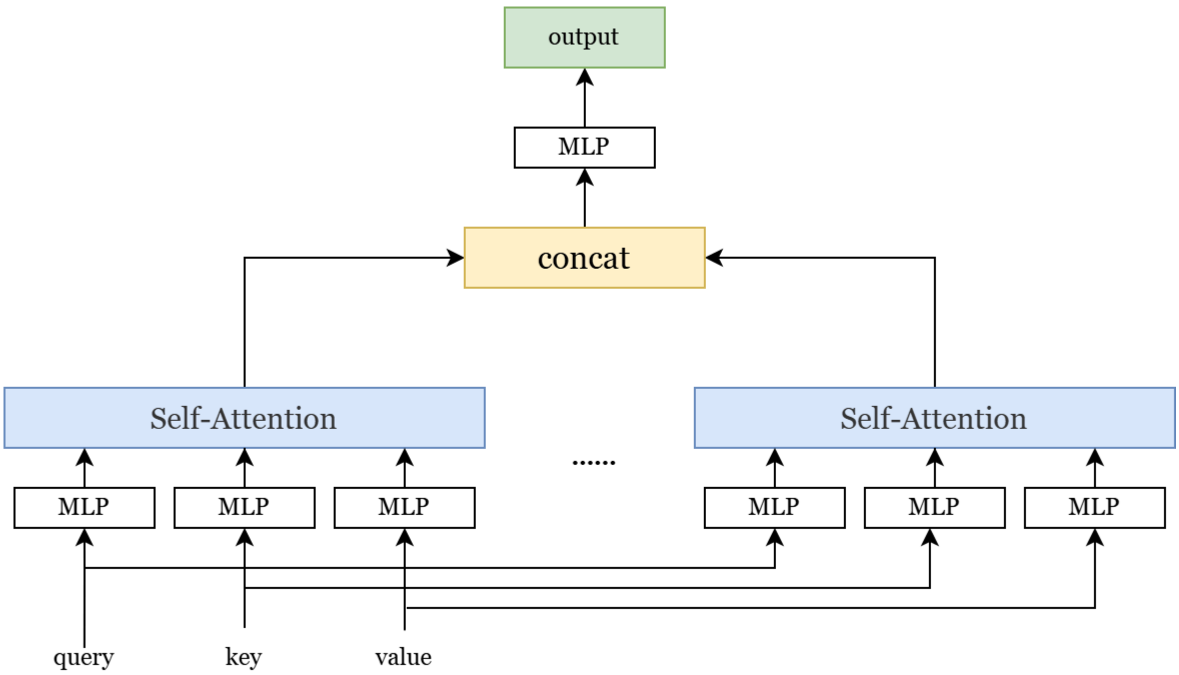
记该"上三角"M矩阵为Masked Attention权重：

$$A_h = \mathrm{softmax}(S_h + \mathbf{M}).$$

对于行 i 只把允许的列保持原值，其余置 $-\infty$，softmax 后相当于把不合法位置的概率压到 0。

在实践中，当给定相同的香询、键和值的集合时，我们希望模型可以基于**相同的注意力机制**学习到**不同的行为**，然后将不同的行为作为知识组合起来，捕获序列内各种范围的依赖关系(例如，短距离依赖和长距离依赖关系)。

为此，我们可以用独立学习得到的 H 组不同的**线性投影(MLP)**来变换查询、键和值。然后，这 h 组变换后的查询、键和值将并行地送到**注意力汇聚**中。最后，将这 H 个注意力汇聚的输出**拼接在一起**，并且通过另一个可以学习的线性投影进行变换，以产生最终输出。这种设计即为**多头注意力机制**，对于 h 个注意力汇聚输出，每一个注意力汇聚都被称作一个**头**(head)。



对于每个注意力头 $\mathbf{h}_i(i=1,...,H)$：

$$\mathbf{h}=A_hV_h\in\mathbb{R}^{n\times d_h},$$

经拼接后再投影回 d（即嵌入空间）：

$$MHSA(X)=[\mathbf{h}_1,...,\mathbf{h}_h]W_O+b_O,\quad W_O\in\mathbb{R}^{Hd_h\times d}.$$

基于这种设计，每个头都可能会关注输入的不同部分，可以表示比简单加权平均值更复杂的函数。

1.2.3 Position-wise Feed-Forward Network

基于位置的前馈网络对**序列中的所有位置的表示进行变换**时，使用的是**同一个多层感知机(MLP)**，这就是称前馈网络是基于位置的原因：

$$FFN(x)=\sigma(XW_1+b_1)W_2+b_2,$$

其中 $W_1\in\mathbb{R}^{d\times d_{ff}}$，相当于升维的操作；$\sigma$ 为激活函数，如Relu；$W_2\in\mathbb{R}^{d_{ff}\times d}$，相当于降维的操作，回到原通道数；$b_{1,2}$ 为偏置项。

这个操作将自注意力产生的 **方向特征** 转换成 **坐标系内的高阶混合特征**，补足网络的非线性表达力。

1.3 Transformer的有效性分析

从**秩**的视角审视纯注意力网络（Self-Attention Network, SAN），发现：在没有跳跃连接（skip connections）和前馈网络（MLP）的情形下，随着层数加深，其**输出矩阵会以双指数速度退化到秩 1**——即所有 **token 最终"趋于同质"**。

Skip Connection通过**允许信息绕过某些Self-Attention层**，从而在路径分解中引入了大量**短路径**。最极端的情况是一条长度为0的路径，它直接将原始输入传递到输出，完整保留了输入的秩。这些短路径不会经历Deep Layer导致的严重秩坍塌，因此它们的存在有效地阻止了整个网络输出的退化，这揭示了跳跃连接在Transformer中一个此前未被充分认识的关键作用：**防止秩坍塌。**

MLP块作为**非线性变换**，可以增**加其输入矩阵的秩**，与Self-Attention层的降秩进行博弈。MLP的能力可以通过其**Lipschitz** constant来衡量，**Lipschitz常数越大的MLP，其提升秩的能力越强**，从而能更有效地减缓秩坍塌的速度。我们将在后面的篇章对上述理论进行详细证明。

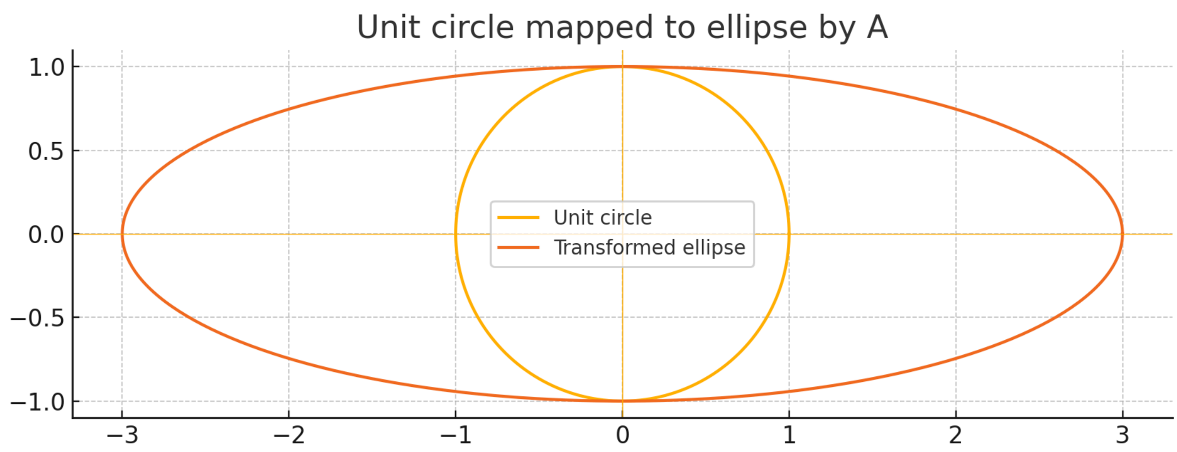
2 Spectral Properties of Attention

2.1 求解注意力矩阵谱范数的上下界

我们先来解释一下什么是谱范数：**谱范数（spectral norm）**是矩阵 $A \in \mathbb{R}^{m\times n}$在 ℓ_2 意义下的算子范数——也就是把它看成线性变换 $x \mapsto Ax$ 时对向量欧氏长度的"最大放大倍数"：

$$\|A\|_2 = \max_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \max_{\|x\|_2=1} \|Ax\|_2. \tag{1}$$

如何理解呢？我们可以设想一个这样的场景：



- 将A看作为一个**线性变换**，它将向量 x 拉伸或压缩为 Ax；
- 把**单位球**$\|x\|_2 = 1$看成输入空间的"所有方向"，对其进行线性变换后会得到一个**椭球**（或更高维的超椭球）：$E = \{Ax : \|x\|_2 = 1\}$；
- "最大"指椭球的最长"半径"，**椭圆最远离原点的那一点**到原点的距离，对应的那条半径的长度就是谱范数；对应的方向叫 **主奇异向量**。

在开始证明前，我先强调下**欧几里得 ℓ_2 范数与谱范数的区别**（因为我自己也经常搞混）：

名称	记号（常见）	输入对象	本质含义
欧几里得范数 / ℓ_2 范数	$\|x\|_2$（有时简写 $\|x\|$）	**向量** $x \in \mathbb{R}^n$	向量的"长度" $\sqrt{\sum_i x_i^2}$
谱范数（Spectral norm）	$\|A\|_2$	**矩阵** $A \in \mathbb{R}^{m\times n}$	"最坏方向"的放大倍数 $\max_{\|x\|_2=1} \|Ax\|_2$ = **最大奇异值**

> **关键区别**：ℓ_2 范数作用在 **向量**，谱范数作用在 **矩阵**；两者的"2"都指用欧几里得距离来度量，但量的是完全不同的对象。

现在我们来尝试证明**单头注意力权重矩阵** A 的谱范数 $\|A\|_2$ 的理论上界，已知我们有：

$$A \in \mathbb{R}^{n\times n}, \quad A_{ij} \geq 0, \quad \sum_{j=1}^{n} A_{ij} = 1 \quad (i = 1, \ldots, n).$$

即 A 为**行随机矩阵**。

根据公式(1)，为了求$\|A\|_2$，我们可以求$\|x\|_2 = 1$时$\|Ax\|_2$的值（即单位向量x的最大拉伸倍数）：

$$\|Ax\|_2^2 = \sum_{i=1}^{n} \left(\sum_j A_{ij} x_j \right)^2. \tag{2}$$

由 **Jensen不等式：**

$$\varphi\left(\sum_{i=1}^{n} \lambda_i x_i\right) \leq \sum_{i=1}^{n} \lambda_i \varphi(x_i), \quad \lambda_i \geq 0, \sum \lambda_i = 1, \tag{3}$$

我们可以得到：

$$\sum_{i=1}^{n} \left(\sum_j A_{ij} x_j \right)^2 \leq \sum_{i,j} A_{ij} x_j^2 = \sum_j \left(\sum_i A_{ij} \right) x_j^2. \tag{4}$$

定义：

$$c_{\max} := \max_{1 \leq j \leq n} \sum_{i=1}^{n} A_{ij},$$

为**最大列和。**故利用 c_{max} 我们最终可以得到以下结论：

$$\|Ax\|_2^2 \leq c_{\max} \sum_j x_j^2 = c_{\max} \|x\|_2^2. \tag{5}$$

因 $\|x\|_2 = 1$，两边**开平方**可以得到以下结果：

$$\|A\|_2 = \max_{\|x\|_2 = 1} \|Ax\|_2 \leq \sqrt{c_{\max}}. \tag{6}$$

最坏的情况是**全1都集中在到一列**，在这个情况下$c_{max} = n$，为最大值，故：

$$\|A\|_2 \leq \sqrt{n}. \tag{7}$$

上界得证！

接下来我们来简单证明**下界**：

构造单位向量：$u := \frac{1}{\sqrt{n}} [1, 1, \ldots, 1]^\top, \quad \|u\|_2 = 1$。因 A 的各行和为1，故有：$Au = u$。同时，因为$\|A\|_2$的值**应对应所有单位向量 x 中拉伸程度最大的那一个**（放大倍数最大），故我们可以通过 u 来收缩下界：

$$\|A\|_2 \geq \|Au\|_2 = \|u\|_2 = 1.$$

因此，行随机矩阵**永远不可能**把所有向量都缩短；最小放大倍数就是1。

综合上下界我们可以得到：

$$1 \leq \|A\|_2 \leq \sqrt{n}, \tag{8}$$

可以看出，**只有**当$c_{max} = 1$，即**同时列随机**时，所有列方向与行方向均平衡，椭球最长半轴 = 1，达到最小可能值，做到不对向量进行“放大”：$\|A\| = 1$。

也就是说，在多数情况下，$\|A\| \geq 1$，是expansive（趋向扩散）的。然而，对于Transformer这样的深度学习模型来说，其往往会有很多很深的层，在**前向传播**的过程中，若反复乘以谱范数大于1的注意力权重矩阵，特征向量会被**反复放大**，可能导致**后续层饱和**或**数值发散、溢出**，并把**梯度放大到爆炸**，对误差、噪音也会同样放大。除此之外，模型的 **Lipschitz 常数**等于各层**谱范数乘积的上界**；若任何一层 > 1，总 Lipschitz 增大，输入的微小扰动会被放大，造成**鲁棒性下降。**我们会在第三章再去做详细论证。

2.2 Layer Normalization 对谱范数上界的约束

那么，我们就知道了 A 发散/扩散时，会对模型产生诸多不利影响，而根据我们推导的公式(8)，我们所能做的就是压缩谱范数的上界，从而令 A 趋向于 **non-expansive**。针对这个问题，GPT-2做出了相应的改进，即将LN移动到每一个sub-block的输入位置，也就是 **Pre-Norm**：

$$H_{\ell+1} = H_\ell + \mathrm{SubLayer}\left(\mathrm{LN}(H_\ell)\right).$$

对第 i 个 token 对应的嵌入向量 $h \in \mathbb{R}^d$ 进行**层归一化**操作：

$$z = \mathrm{LN}(h) = \gamma \odot \frac{h - \mu\mathbf{1}}{\sigma} + \beta, \quad \mu = \frac{1}{d}\mathbf{1}^\top h$$

由$\sigma = \sqrt{\frac{1}{d}\|h-\mu\mathbf{1}\|_2^2}$ 可得：

$$\|h - \mu\mathbf{1}\|_2 = \sqrt{d}\sigma \tag{9}$$

由于LayerNorm输出**(去掉平移项β)**是逐坐标缩放：$z = \gamma \odot x = (\gamma_1 x_1, \gamma_2 x_2, \ldots, \gamma_d x_d)^\top$，我们可以将乘以$\gamma$看成"**对角矩阵**作用"：$\gamma \odot x = D_\gamma x, \quad D_\gamma := \mathrm{diag}(\gamma_1, \ldots, \gamma_d)$。利用**谱范数不等式**（诱导性），对于任意矩阵–向量乘积都有：$\|D_\gamma x\|_2 \leq \|D_\gamma\|_2\|x\|_2$。而对角矩阵的奇异值就是各对角绝对值，则**最大奇异值**即为：$\|\gamma\|_\infty$，而$\|\gamma\|_\infty = \max_{1\leq i\leq d}|\gamma_i|$，即取**向量**所有分量的绝对值，看其中 **最大的那一个。**由此我们可以得到：

$$\|D_\gamma\|_2 = \|\gamma\|_\infty \tag{10}$$

结合$(9), (10)$我们可以推导出：

$$\|z\|_2 = \left\|\gamma \odot \frac{h-\mu\mathbf{1}}{\sigma}\right\|_2 = \left\|D_\gamma \frac{h-\mu\mathbf{1}}{\sigma}\right\|_2 \leq \|\gamma\|_\infty \frac{\|h-\mu\mathbf{1}\|_2}{\sigma} = \|\gamma\|_\infty\sqrt{d}.$$

令 $\Gamma := \|\gamma\|_\infty$，即得：

$$\|z\|_2 \leq \Gamma\sqrt{d}. \tag{11}$$

令$\sigma_Q := \|W_Q\|_2, \quad \sigma_K := \|W_K\|_2$，我们可以得到对应第 i 个token 的 query 和 key 向量的范数：

$$\|q_i\|_2 = \|z_i W_Q\|_2 \leq \|z_i\|_2 \cdot \|W_Q\|_2 \leq \Gamma\sigma_Q\sqrt{d}, \tag{12}$$

$$\|k_j\|_2 = \|z_j W_K\|_2 \leq \|z_j\|_2 \cdot \|W_K\|_2 \leq \Gamma\sigma_K\sqrt{d}. \tag{13}$$

上述推导依然使用了**谱范数不等式**：$\|Ax\|_2 \leq \|A\|_2\|x\|_2$。

基于$|\langle a, b\rangle| \leq \|a\|_2\|b\|_2$ 以及公式 $(11), (12), (13)$，对于**单头注意力** logits：

$$|\ell_{ij}| := \frac{|\langle q_i, k_j\rangle|}{\sqrt{d}} \leq \frac{\|q_i\|_2\|k_j\|_2}{\sqrt{d}} \leq \Gamma^2\sigma_Q\sigma_K\sqrt{d} =: M.$$

即得：$-M \leq l_{i,j} \leq M$。i故对于某一 token i ，其与其他 token 的注意力 logits 的**最大差值**为（即注意力矩阵中同一行的任意两列的最大差值）：

$$\Delta = \max_j \ell_{ij} - \min_{j\prime} \ell_{ij\prime} \leq 2M.$$

已知 softmax 运算：

$$A_{ij} = \frac{e^{\ell_{ij}}}{\sum_{t=1}^n e^{\ell_{it}}}.$$

令该行的最大值为$\ell_{\max}$，则对注意力矩阵的任意元素$A_{(i,j)}$有：

$$A_{ij} \le \frac{e^{\ell_{\max}}}{e^{\ell_{\max}} + (n-1)e^{\ell_{\max}-\Delta}} = \frac{1}{1+(n-1)e^{-\Delta}} \le \frac{1}{1+(n-1)e^{-2M}} =: \alpha$$

于是任意列和：

$$\sum_{i=1}^{n} A_{ij} \le n\alpha =: c_{\max}. \tag{14}$$

将(14)代入(6)：$\|A\|_2 \le \sqrt{c_{\max}}$ 得最终结果：

$$\|A\|_2 \le \sqrt{\frac{n}{1+(n-1)e^{-2\Gamma^2\sigma_Q\sigma_K\sqrt{d}}}}$$

结论：

- 在 **Post-Norm** 或缺乏足够归一化的体系里，随着**模型深度**增大，注意力矩阵的**范数上界**可能一路抬高到理论极限 $\sqrt{n}$，并把梯度放大到爆炸；
- **Pre-Norm** 相当于每层先重置尺度，使得 Q, K 的范数被固定常数控制，有效阻止这种随层数失控的增长。$\Gamma, \sigma_Q, \sigma_K$ 由初始化和正则控制，不随深度指数增长，是与层数无关的常数。于是 softmax logits 的跨度 Δ 有一个与层深无关的上界，进而把注意力矩阵 A 的最大列和 $c_{\max}$ 压到离 1 很近的范围。没有这种归一化，$\|A\|_2$ 可能随深度朝 $\sqrt{n}$ 飙升并引发梯度爆炸。

结论	说明
更紧的谱范数上界	Pre-Norm把 $\|A\|_2$ 的理论上界从"可能是 $\sqrt{n}$"压至**深度无关的常数** $\sqrt{c_{\max}(M)}$。
梯度不易爆炸/消失	反向链中的 $A^\top$ 同样几乎 non-expansive；跨层乘积不会指数放大或衰减。
训练深度可大幅增加	这正是 GPT-2 以后 Transformer 模型普遍采用 Pre-Norm 的经验原因之一。