归一化(Normalization)

概念与定义

归一化是一种将数据映射到特定数值区间(如[0,1])的数学变换技术，目的是为了提升计算的稳定性并优化学习效率，其本质是调整数据的量纲而保持分布形态不变。

常见的归一化方法包括：

- 层归一化(Layer Normalization)
- 批量归一化(Batch Normalization)
- RMS归一化(Root Mean Square Normalization)

作用

- 提高数值计算稳定性、优化收敛速度：未归一化的数据如果数值较大，可能产生较大的 MSE 值，导致梯度值过大，使得模型训练过程不稳定。
- 提升潜在内存效率：归一化通过缩减数据表示范围间接提高了内存利用效率。特别是在使用 8 位或 16 位浮点数据进行训练时，数据范围的收窄会使得低精度表示更为精确。这种特征可以通过量化技术或 FP16 计算进一步降低模型训练阶段的内存占用。(归一化技术本身不直接减少内存占用，需要与量化或低精度计算技术协同应用，才能有效实现内存优化)

假设我们有张量 $X_{b,s,d}$ ，其中 b 是批次大小，s 是序列长度，d 是嵌入维度或隐藏状态维度。下面将依次展示层归一化、批量归一化以及 RMS 归一化的具体计算原理。

层归一化

层归一化是指对每个 token 的嵌入向量或隐藏状态向量进行归一化。也就是对张量 $X_{b,s,d}$ 中的每个 $X_{b,s,:}$ 进行归一化。归一化公式如下：

$$Z_{b,s,d} = \frac{X_{b,s,d} - \mu_{b,s,1}}{\sigma_{b,s,1} + \epsilon} \cdot \gamma_d + \beta_d$$

其中， $Z_{b,s,d}$ 是归一化后的张量； $\mu_{b,s,1}$ 是张量 $X_{b,s,d}$ 在 d 维度上的均值； $\sigma_{b,s,1}$ 是张量 $X_{b,s,d}$ 在 d 维度上的标准差； ϵ 是一个很小的常数，用来避免除零错误； γ_d 和 β_d 分别是缩放和偏移因子，是可学习的参数。 $\mu_{b,s,1}$ 和 $\sigma_{b,s,1}$ 的计算方法如下：

$$\mu_{b,s,1} = \frac{1}{d}\sum_{i=1}^{d} X_{b,s,i}$$

$$\sigma_{b,s,1} = \sqrt{\frac{1}{d}\sum_{i=1}^{d}(X_{b,s,i} - \mu_{b,s,1})^2}$$

层归一化的作用是将每个样本的特征进行归一化，使得特征在不同样本之间具有相似的分布，有助于提高模型的训练效果和泛化能力。

批量归一化

批量归一化会沿着批次维度 b 计算均值和方差，然后进行归一化。归一化公式如下：

$$Z_{b,s,d} = \frac{X_{b,s,d} - \mu_{1,s,d}}{\sigma_{1,s,d} + \epsilon} \cdot \gamma_d + \beta_d$$

其中， $Z_{b,s,d}$ 是归一化后的张量； $\mu_{1,s,d}$ 是张量 $X_{b,s,d}$ 在 b 维度上的均值； $\sigma_{1,s,d}$ 是张量 $X_{b,s,d}$ 在 b 维度上的标准差； ϵ 是一个很小的常数，用来避免除零错误； γ_d 和 β_d 分别是缩放和偏移因子，是可学习的参数。 $\mu_{1,s,d}$ 和 $\sigma_{1,s,d}$ 的计算方法如下：

$$\mu_{1,s,d} = \frac{1}{b}\sum_{i=1}^{b} X_{i,s,d}$$

$$\sigma_{1,s,d} = \sqrt{\frac{1}{b}\sum_{i=1}^{b}(X_{i,s,d} - \mu_{1,s,d})^2}$$

批量归一化本质上是使用一个batch中的均值和方差来模拟全部数据的均值和方差，所以其归一化结果与 b 的大小有一定关系。

为什么语言建模中不常用批量归一化？主要有3个原因：

- 当 b 较小的时候，批量归一化计算的均值和方差与总体均值和方差可能有加大差异，从而使得批量归一化的结果不可信，而语言建模中的 b 往往较小。
- 语言序列数据在不同序列之间、相同位置的token差异性很大，他们相同位置上的特征，属于独立、但不同分布的数据，批量归一化从统计学的角度上来讲并不合理。
- 由于批量归一化的结果和 b 相关，而在训练和推理时 b 的大小通常是不一样的，这会导致模型训练和推理时的批归一化结果不一样，产生“训练时表现好，推理时表现差”的现象。

RMS 归一化

与层归一化相比，RMS 归一化去掉均值堆砌操作，因此只需要进行 1 次数据扫描(层归一化需要计算一次均值、计算一次标准差，因此执行了 2 次数据扫描)。实践证明，RMS 归一化在保持性能的同时，降低了计算成本。RMS 归一化公式如下：

$$Z_{b,s,d} = \frac{X_{b,s,d}}{\mathrm{RMS}(x) + \epsilon} \cdot \gamma_d$$

其中， $Z_{b,s,d}$ 是归一化后的张量； ϵ 是一个很小的常数，用来避免除零错误； γ_d 是缩放因子，是可学习的参数。 $\mathrm{RMS}(x)$ 的计算方法如下：

$$\mathrm{RMS}(x) = \sqrt{\frac{1}{b}\sum_{i=1}^{d} X_{b,s,i}^2}$$

RMS 归一化的优点在于：

- 降低了计算资源需求。
- 不强制均值向 0 对齐，有效缓解了**梯度消失问题**。因为 0 均值对齐可能会导致梯度在反向传播过程中逐步衰减，从而产生梯度消失。

RMS 归一化实现代码(内含详细注释及演示案例)