https://www.cnblogs.com/nickchen121/p/16518604.html

标准：

$$y = \frac{x - E(x)}{\sqrt{Var(x) + \epsilon}} * \gamma + \beta$$

加速收敛:在每一层输出中，将均值归零、方差归一有助于减小不同特征之间的尺度差异，避免梯度在某些方向上过大或过小。这种归一化减少了梯度消失或梯度爆炸的风险，从而加速了模型收敛，使得模型更快找到最优参数。
稳定训练过程：
在训练过程中，不同样本的输入会导致各层的激活值有不同的分布，这种分布的变化会影响训练的稳定性。层归一化通过对每一层的输出做归一化，使得激活值的分布在每一步都相对一致，减少了网络参数更新过程中的波动，从而稳定了训练过程。
提高模型鲁棒性：
层归一化让每层的输出分布固定，有助于模型在处理不同样本时具有更高的鲁棒性。这是因为每层输出的标准化能够让模型更好地适应不同的输入，减少对输入分布的依赖，从而在面对噪声或新的数据分布时更稳健。
非线性激活函数更有效：
在层归一化之后，数据分布集中在均值0、方差1附近，能让激活函数（例如ReLU、Tanh）在有效区间内发挥作用，避免输入值过大或过小，导致激活函数的梯度消失。这种归一化使得激活函数的导数较大，信息更容易传播到前面的层，提升了模型的学习效果。

在模型里:RMSNorm

x * torch.rsqrt(x.pow(2).mean(-1, keepdim=True) + self.eps)

$$\hat{\mathbf{x}} = \frac{\mathbf{x}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2 + \epsilon}}$$

好处：

1. **减少梯度消失/爆炸问题**：
2. **简化和高效**
3. **不依赖均值**：对数据的偏移不敏感，避免了某些情况均值时的噪声干扰
4. **提高训练稳定性**
5. **性能提升**
6. **方便激活函数**