梯度消失与梯度爆炸

梯度消失和梯度爆炸是深度学习，尤其是训练深层神经网络时常见的两个问题。它们都涉及到在反向传播过程中计算的梯度(即损失函数关于权重的导数)如何变化。

梯度消失

梯度消失是指在使用反向传播算法更新神经网络参数时，靠近输入层的层的梯度变得非常小，几乎为零。这意味着这些层的权重更新非常缓慢，甚至几乎不更新，导致学习过程停滞。这个问题通常发生在深层网络中，特别是当激活函数如sigmoid或tanh被使用时，因为这些函数在其大部分定义域内的导数都非常小。随着反向传播从输出层向输入层逐层计算梯度，每经过一层，梯度都会乘以该层激活函数的导数值，如果这个值很小，那么多次相乘后，梯度就会变得极其微小。

梯度爆炸

与梯度消失恰恰相反，梯度爆炸指的是在反向传播过程中，靠近输入层的层的梯度变得非常大。这通常是由于梯度的累积效应导致的，特别是在存在大量层或者某些层的权重初始化过大时。如果激活函数的导数在某些点上很大，或者权重设置得过大，那么随着反向传播的进行，梯度可能会指数级增长，导致权重更新幅度过大，使得模型不稳定，难以收敛。

解决方法

- **权重初始化**：正确的初始化方法能够显著减少梯度消失或爆炸的风险。例如，Xavier/Glorot初始化和He初始化是两种广泛使用的策略，它们分别针对激活函数是sigmoid/tanh和ReLU及其变种的情况设计。
- **梯度裁剪**：梯度剪裁(Gradient Clipping)是一种有效的防止梯度爆炸的技术，通过设定一个阈值，当梯度的范数超过这个阈值时，就按照比例缩小梯度，使其不超过该阈值。
- **激活函数选择**：传统的sigmoid和tanh函数容易导致梯度消失问题。相比之下，ReLU及其变种(如Leaky ReLU、Parametric ReLU、ELU等)有助于减轻这个问题，因为它们在正输入区域具有恒定的梯度。
- **标准化**：批标准化(Batch Normalization)、层标准化(Layer Normalization)等通过在每一层的输入上进行标准化(调整其均值和方差)，可以使得网络更加稳定，从而减少梯度消失或爆炸的可能性，并允许每个层独立学习，加速训练过程。

- **跳跃连接**：通过引入跳跃连接(skip connections)，可以让信息在不经过任何转换的情况下从前一层直接传递到后面的一层或多层，这有助于解决梯度消失问题，尤其是在非常深的网络中。
- **使用LSTM或GRU处理序列数据**：对于循环神经网络(RNN)，特别是处理长序列数据时，梯度消失是一个常见问题。长短期记忆网络(LSTM)和门控循环单元(GRU)是专门为此设计的改进版本，它们通过特定的结构设计来帮助捕捉长期依赖关系，同时缓解梯度消失问题。
- **优化网络架构**：有时，重新考虑网络的整体架构也可以帮助缓解这些问题。例如，减少网络的深度或者宽度，采用更为复杂的结构如注意力机制等。