1. 前言

1.1 量化是什么？

量化是模型压缩的一种方式。量化就是把高位宽（例如 32float）表示的权值或者激活值用较低位宽来近似表示（int8），在数值上的体现就是将连续的值离散化。

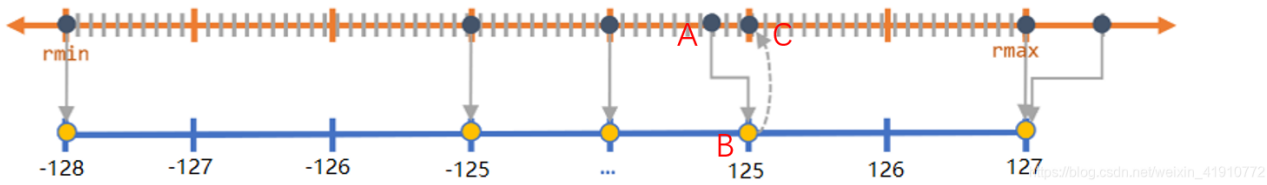
1.2 量化会带来什么？

量化主要用在边缘计算等硬件限制较大的场景下，即工业应用上（总不能在边缘上都带着 GPU 吧）。现有很多先进的神经网络（例如 resnet，densenet）在分类、识别上都取得了较好的效果，但其普及程度远不及效果稍差但模型小、运算快的 mobilenet，而 mobilenet 就是在权衡速度、识别率下产物。当然了，mobilenet 不是量化模型，只是用来举个例子，用于说明量化的潜力。

以下为量化所带来的一些影响：

> 优点 1：加快运算速度。当把 32float 转变为 int8 表示时，在不考虑系统有浮点加速模块时，**定点运算要比浮点运算快**，感兴趣的可查阅定点数和浮点数运算的区别 优点 2：减少存储空间。若将 32 浮点数转变为 8 位表示时，存储空间减小到了 1/4 大小。 缺点 1：在用低带宽数值近似表示时，会造成一些精度损失。值得高兴的是，神经网络的参数大多是冗余的（或者说是对噪声的容忍度），所以当在近似变换时对精度的影响不是特别大。

下面用图来说明量化是怎么带来损失的，** A 为实际的浮点值，量化后近似为 B，但其表示的值为 C 点，缩放因子越大，A 和 C 的距离就越远，误差就越大，所以在量化时引入的近似会带来一些精度上的损失。（后面会具体讲解如何设置最值，来找到合适的缩放因子）



2. 量化具体介绍

2.1 非对称量化

1）首先，设置浮点数的最大值 x_max, 最小值 x_min：

> 对于权重：权重在训练后大小都是固定的，一般直接求出权重的最大值和最小值。 对于激活值：会随着输入值的改变而改变，所以不能直接求其最值，google 使用了滑动均值平均的方法，TensorRT 使用 KL 散度，Easyquant 使用 cos 相似度，后面会具体介绍。

2）其次，设置要量化的范围 x_q_max,x_q_min，在非对称量化下为 [0,255]。3）之后，计算缩放因子 Scale (float32) 和平移因子 Zero_point (int8):

$$s = \frac{X_{max} - X_{min}}{255} \quad z = round(\frac{X_{min}}{s})$$

4）最后进行量化和解量化：图中的 N_{levels}-1 为 255，round 为四舍五入

Once the scale and zero-point are defined, quantization proceeds as follows:

$$x_{int} = round(\frac{x}{\Delta}) + z \tag{1}$$

$$x_Q = clamp(0, N_{levels} - 1, x_{int}) \tag{2}$$

where

$$clamp(a, b, x) = \begin{cases} a & x \leq a \\ x & a \leq x \leq b \\ b & x \geq b \end{cases}$$

The de-quantization operation is:

$$x_{float} = (x_Q - z)\Delta \tag{3}$$

然而为什么是在量化时使用先放缩再移位，解量化时先移位再放缩呢？可不可以反过来呢？答案是否定的，主要从效果和运算两个方面考虑：