本项目使用numpy搭建的神经网络模型实际输出结果和torchvision.models.quantization.mobilenet\_v2输出的结果一定程度上的不同，原因是所使用的M0 M1的精度有所不同。其中，本项目中使用的精度是int32，而在torchvision的模型中使用的精度是float64。

compare\_each\_layer的python代码中，使用的是numpy搭建的神经网络，读取torchvision模型中的input\_scale, output scale, bias, weight scale，并用于计算M0 M1，使用float64精度。并且将torchvision模型的每一层的输入送入该模型中，避免产生累积误差。最后对每一层的输出进行比较。从运行结果中可以直观地看出，只有第51层输出中有5个数不同，其余层都是完全相同的。

然而，如果使用round对M0 M1进行处理，则会导致两个模型之间每一层输出都有一定程度上的差异。例如，在使用dog.jpg这张图片作为输入的时候，最终输出只有20%左右的数字是一样的。

但为了在设计的加速器上实现该神经网络，本项目必须使用int32作为M0 M1的精度。为了了解这样做对图像分类任务的效果造成的影响，在MobileNetV2\_numpy\_imagenet.py中使用了int32精度的M0 M1，并对imagenet的验证集的50000张图片进行推理，accuracy为top1：71.61%，top5：90.11%。而torchvision的模型的accuracy为top1：71.71%，top5：90.25%。这样的精度损失是完全可以接受的。