

COSE474 Project #2: CNN Architecture Implementation

2021320322 윤민서

In this project, it is important to build a CNN Architecture based on the ideas used in ResNet to predict the images in CIFAR-10 dataset.

```
checkpoint = torch.load(PATH, map_location=torch.device('cpu'))
```

Before writing the code for learning in earnest, I modified one of the lines of code on main.py to set it up for learning through the CPU.

An important part of resnet50_skeleton.py is padding. In the process of implementing one 7x7 convolutional layer and several conv3x3, appropriate padding should be set to match the dimension according to kernel_size and stride, and I think this part plays the most important part in the operation principle.

```
conv3x3(middle_channels, middle_channels, 1, 1), # ((n - 3 + 2) / 1) + 1 = n
```

Another important part is downsampling. Downsampling plays a big role in reducing the amount of data. In the case of this project, it plays a role in reducing the resolution. If downsampling is not performed, the implementation will not be performed correctly due to the change in the size created by pooling and stride. In the case of ResNet-50, downsampling is performed in the last Residual block of layer2 and layer3, and the key point is to set the stride to 2.

```
conv1x1(in_channels, middle_channels, 2, 0), # downsampling from n to n / 2
```

The result of main.py using ResNet-50 is as shown in the image below.

```
Epoch [1/1], Step [100/500] Loss: 0.2645
Epoch [1/1], Step [200/500] Loss: 0.2868
Epoch [1/1], Step [300/500] Loss: 0.2976
Epoch [1/1], Step [400/500] Loss: 0.3003
Epoch [1/1], Step [500/500] Loss: 0.3043
Accuracy of the model on the test images: 82.72 %
```

A result slightly above the target accuracy of 80 percent is observed. It shows somewhat lower results than the practice model VGG-16 (86.18% in my case) or the actual ResNet-50. It can be assumed that the reasons for these results are that the actual ResNet-50 model has a fifth layer and the structure of the Residual Block is slightly different from the actual one.