

```

class CNNClassifier(nn.Module):

    def __init__(self):
        super(CNNClassifier, self).__init__()
        conv1 = nn.Conv2d(1, 6, 5, 1)
        pool1 = nn.MaxPool2d(2)
        conv2 = nn.Conv2d(6, 16, 5, 1)
        pool2 = nn.MaxPool2d(2)

        self.conv_module = nn.Sequential(
            conv1,
            nn.ReLU(),
            pool1,
            conv2,
            nn.ReLU(),
            pool2
        )

        fc1 = nn.Linear(16*4*4, 120)
        fc2 = nn.Linear(120, 84)
        fc3 = nn.Linear(84, 10)

        self.fc_module = nn.Sequential(
            fc1,
            nn.ReLU(),
            fc2,
            nn.ReLU(),
            fc3
        )

        if use_cuda:
            self.conv_module = self.conv_module.cuda()
            self.fc_module = self.fc_module.cuda()

    def forward(self, x):
        out = self.conv_module(x) # @16*4*4
        out = out.view(-1, 16*4*4)
        out = self.fc_module(out)
        return F.softmax(out, dim=1)

```