AI Disclosure: Same as SimpleCNN, Search in Chatgpt for data augmentation method, learning two method and put them in the transform_trainer code: transforms.RandomHorizontalFlip(), transforms.RandomCrop(32, padding=4)

Model Description: Beside the traditional CNN model, the advanced model uses the pretrain method, using ResNet18. I use a Conv2d to suit the ResNet18 with kernel 3. To make the training complex, being able to capture much more properties the model can. I add linear and nonlinear features separately(Batch norm, ReLU). I also added an early stop in this model, so if the validation accuracy doesn't increase large enough in 3 consecutive epochs, the training stops.

Hyperparameter Tuning: For training transform, I pick the normalization value (0.5071, 0.4867, 0.4408), (0.2675, 0.2565, 0.2761) since that is the mean and deviation from CIFAR-100's website. I chose the learning rate to be 0.01 as I tried 0.005, which converges really slowly. We choose epoch = 20 so the model has enough time to converge, as it turns out, the train and validation accuracy both tend to be stable after 15 epoch.

Regularization Techniques: I used a drop_out rate at 0.2 to prevent overfitting. I also added a L2 weight decay so the model tends to have average weights. I can't be 100% sure the L2 weight decay helps the model since I changed other parameters in the meantime, but I do believe it helps.

Data Augmentation: I still use random crop and randomHorizontalFlip to improve the performance of data the model had not seen before, and to prevent overfitting.

Results Analysis: The ending score 0.32 is a satisfactory result. Advanced CNN has more parameters and it got pretrained, so it performs better than the Simple CNN model. I am still noticing overfitting as train accuracy and validation accuracy diverges too early, leading to a difference of 15%. This might be due to the fact that ResNet18 is still not a too powerful pretrain backbone, or that the hyperparameters need further tuning.

Experiment Tracking Summary: Training and validation accuracy diverges at around 5 epochs, leading to a 15% difference at the end. Model stops at 17 epochs, earlier than the epoch 20 ending point due to early stopping.