AI Disclosure: 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: Choose 2 Conv2d layers and one linear layer. Choose a lower size since this model is aimed to get a pipeline so only want to have a general idea about the dataset. Choosing epoch = 5 also based on this consideration. Adding the ReLU function to simplify the model. The main idea is it costs little time and computational power.

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 0.001 since I tried higher values and the performance is not going too well, the loss converges very slowly. I run the batch_size optimizer method provided, it gives me 512, just in case I choose to run on 256.

Regularization Techniques: I used a drop_out rate at 0.2. Since I am training on a simple model, I don't think I need a large number to prevent overfitting.

Data Augmentation: I used 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 0.2 is expected. Simple CNN has too few layers thus when coming to a large data set and 100 labels it cannot do very well. It may suit well for CIFAR-10 but not CIFAR-100. The improvement could be increasing layers or optimizing hyperparameters.

Experiment Tracking Summary: For training accuracy and validation accuracy, they both converge slowly at first and increase faster over epochs. This might indicate I can make a better choice on choosing learning rate.