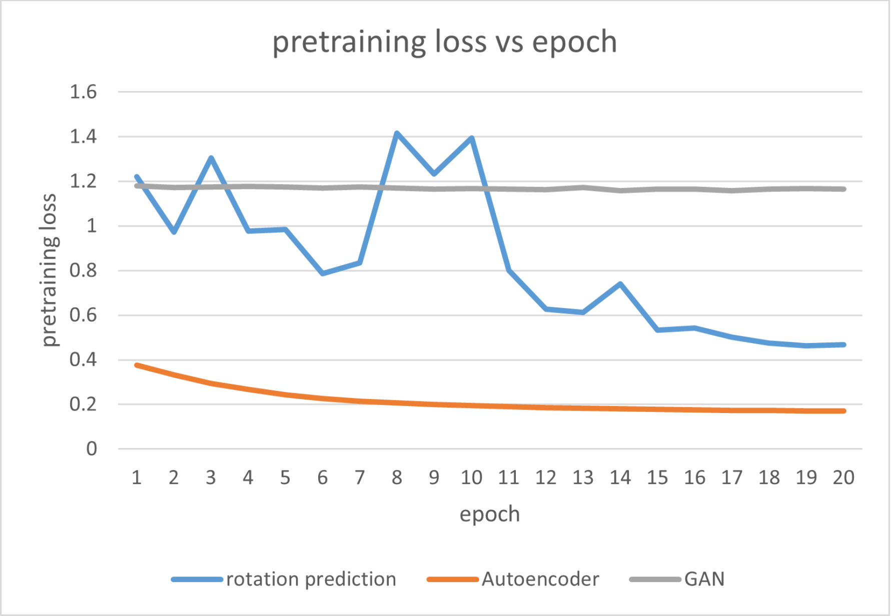

My task is to recognize all digits in the original MNIST dataset.

The first model is a rotation-based self-supervision using FastAI2. The pretraining objective used is rotation prediction, in which We will input rotated images and learn useful features from classifying the amount of degrees rotated. For downstream digit classification results: Using only 0.3% of labeled data for training, it reaches the accuracy of 70% after 25 epochs. With full labeled data for training, it reaches 95% after 25 epochs

For pros, in the downstream task, it can achieve high accuracy using a small amount of classification labeled data for training. The pretraing reaches a high accuracy after 10 epochs.For cons, The running time of each epoch is relatively long compared with other 2 models(on average 1.75min)

The second model is a autoencoder-pretrained model. In pretraining step, I used 30000 unlabeled mnist data to train an autoencoder in unsupervised way. When the training of autoencoder is completed, encoder part of autoencoder is extracted to compress the input signal which then fed into supervised multineural network as baseline model. For downstream digit classification results: Accuracy is 55.10% after 100 epochs

For pros, it reduces the dimensionality of the data so the training process is fast with 3s per epoch. For cons,if your training data for autoencoders is not representative of your testing data, then you wind up obscuring information rather than clarifying it. This might lead to low accuracy.

The third model is a GAN pretrained model. I used GAN to train the part of the unlabeled dataset in the pretraining stage and then using the trained generator and discriminator in the testing period.For downstream digit classification results: Accuracy is 55.87% after 100 epochs

For pros, it reduced size and costs and increased energy efficiency. For cons,accuracy is low.