Assignment 4

2018707017 정용훈

●개요

주어진 데이터 셋을 활용해 CNN모델을 사용한다. 이번에 주어진 데이터는 CIFAR보다 적은 2개의 Classification이고, 데이터 양도 적다.

지난 Assignment 3와 다르게 전이 학습을 통해 결과를 비교한다.,

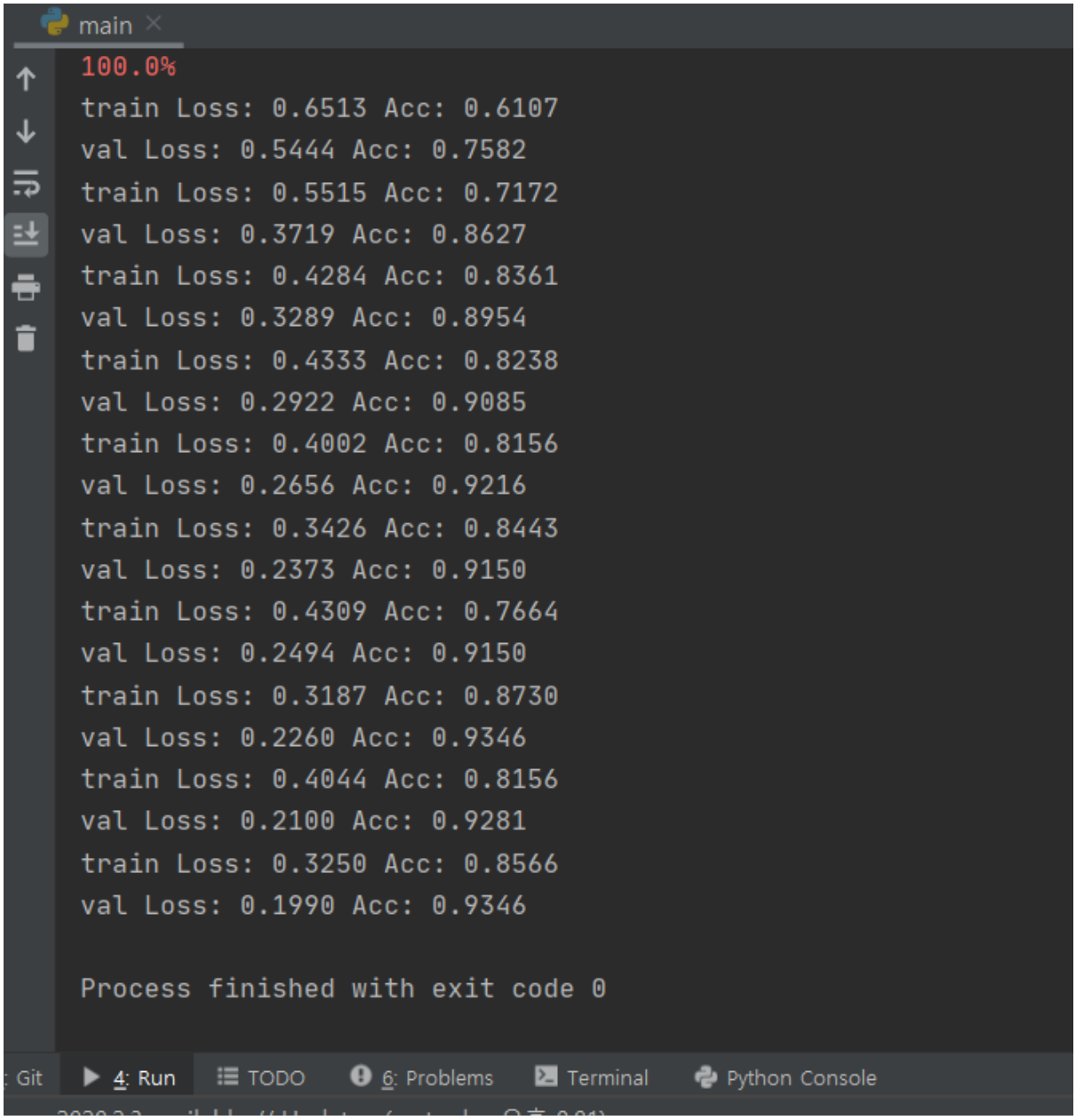
●구현 방법

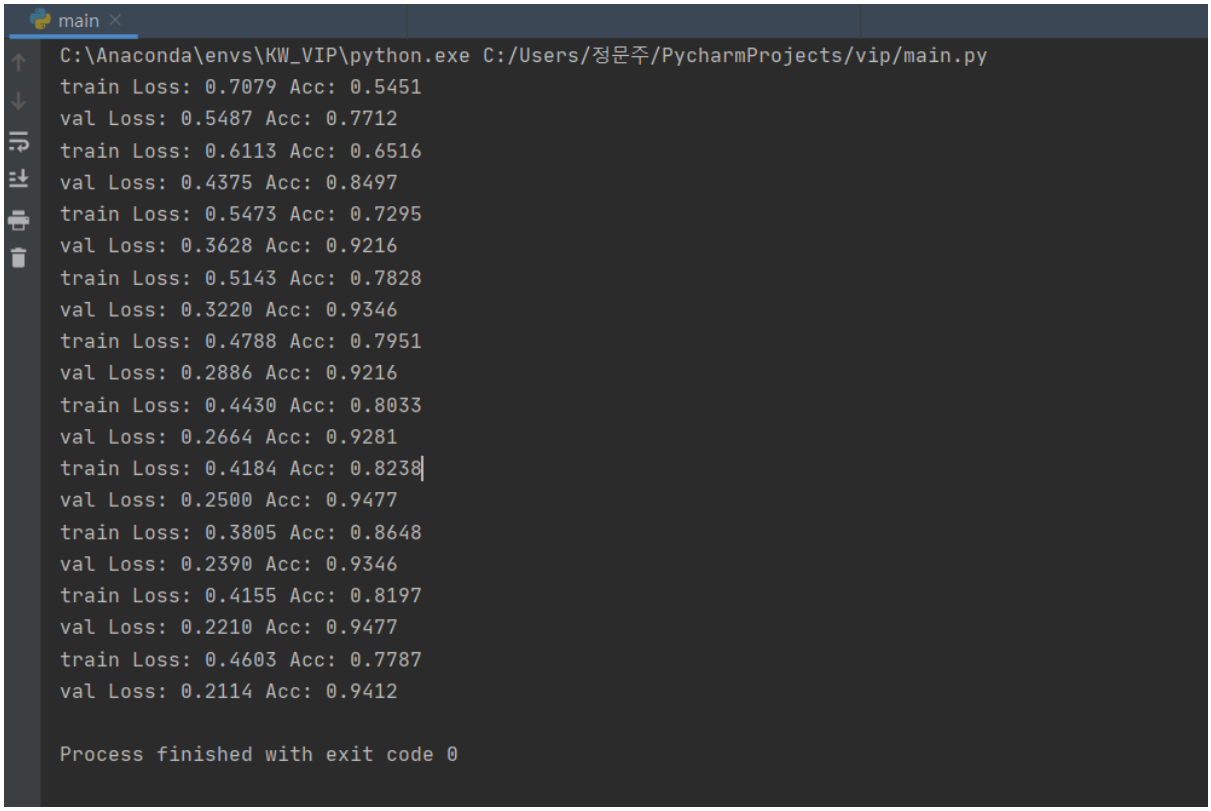
본인 네트워크는 지난 Assignment3의 네트워크를 이용했고, 지난 과제에서 자세히 기술했으니 생략한다.

Strategy1은 Pre-trained된 Resnet-18 모델을 가져와 처음부터 학습시키는 방식으로 진행하였다.

Strategy3는 Strategy1과 달리 Convolution layer만 고정시키고 Fully connected layer로 Classification하는 부분만 추가 학습 시켰다.

● 결과 화면


main
100.0%
train Loss: 0.6513 Acc: 0.6107
val Loss: 0.5444 Acc: 0.7582
train Loss: 0.5515 Acc: 0.7172
val Loss: 0.3719 Acc: 0.8627
train Loss: 0.4284 Acc: 0.8361
val Loss: 0.3289 Acc: 0.8954
train Loss: 0.4333 Acc: 0.8238
val Loss: 0.2922 Acc: 0.9085
train Loss: 0.4002 Acc: 0.8156
val Loss: 0.2656 Acc: 0.9216
train Loss: 0.3426 Acc: 0.8443
val Loss: 0.2373 Acc: 0.9150
train Loss: 0.4309 Acc: 0.7664
val Loss: 0.2494 Acc: 0.9150
train Loss: 0.3187 Acc: 0.8730
val Loss: 0.2260 Acc: 0.9346
train Loss: 0.4044 Acc: 0.8156
val Loss: 0.2100 Acc: 0.9281
train Loss: 0.3250 Acc: 0.8566
val Loss: 0.1990 Acc: 0.9346
Process finished with exit code 0
Git
4: Run
TODO
6: Problems
Terminal
Python Console



에폭	본인 네트워크		Strategy1		Strategy3	
	Vali loss	Accuracy	Val loss	Accuracy	Val loss	Accuracy
1	0.6917	0.5359	0.6513	0.7582	0.5487	0.7712
2	0.6710	0.5621	0.3719	0.8627	0.4375	0.8497
3	0.6684	0.5621	0.3289	0.8954	0.3628	0.9216
4	0.6684	0.5621	0.2922	0.9085	0.3220	0.9346
5	0.6669	0.5425	0.2656	0.9216	0.2886	0.9216
6	0.6991	0.5752	0.2373	0.9150	0.2664	0.9281
7	0.6735	0.5752	0.2494	0.9150	0.2500	0.9477
8	0.6735	0.6013	0.2260	0.9281	0.2390	0.9346
9	0.6769	0.6013	0.2100	0.9346	0.2210	0.9477
10	0.6917	0.5359	0.1990	0.9346	0.2114	0.9412

내 네트워크 보다 전이 학습으로 Training한 Strategy1과 Strategy3가 전반적인 Accuracy가 훨씬 높고, Loss가 훨씬 작은 것을 확인했다.

또, Strategy1보다 Strategy3가 초반에 더 높은 정확도를 갖는 다는 것을 확인했다.

저번 수업에서 배운 전이 학습과 전이 학습을 중간에 끊어주는 것과의 차이를 실습을 통해 확인했다.