캡스톤 디자인

‘딥페이크 탐지’

#9. Adversarial training III

김지수, 김민지, 민지민

지난 캡스톤 회의 내용

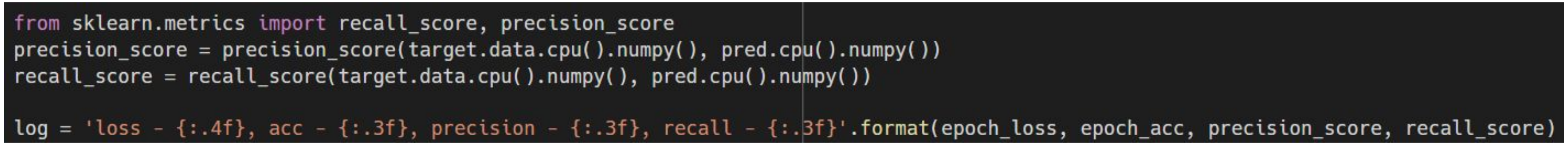
- 지난번 adversarial training 의 잘못된 점을 깨달아 다시 실험 진행했음

=> 이번주 진행한 내용

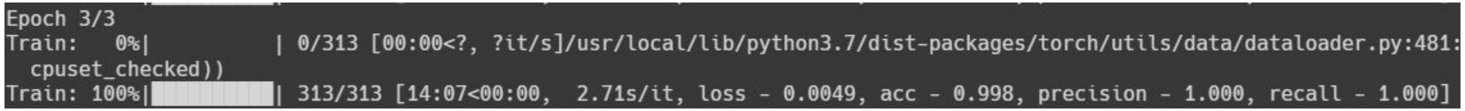
- train : 5000, valid : 1800, test: 1800장으로 실험 진행
- precision, recall 을 넣기로 했으나 문제점 발생

precision, recall

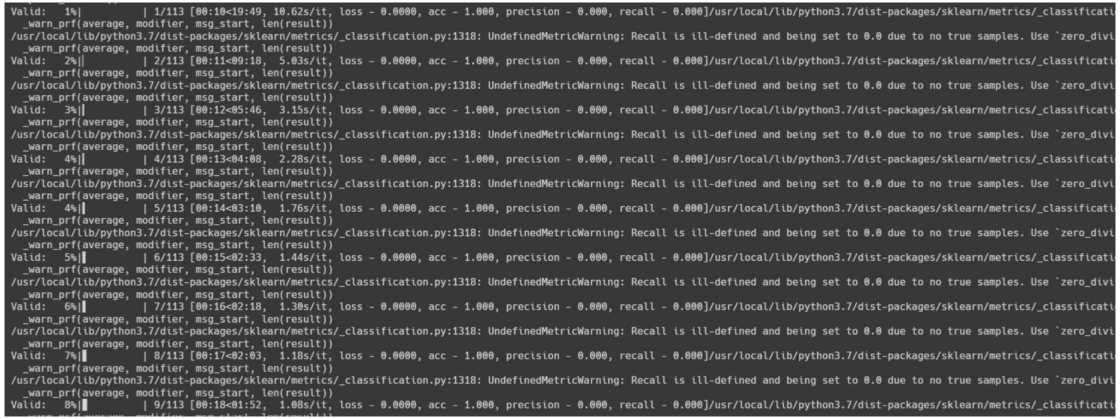
- 사이킷런 패키지 이용



```
from sklearn.metrics import recall_score, precision_score
precision_score = precision_score(target.data.cpu().numpy(), pred.cpu().numpy())
recall_score = recall_score(target.data.cpu().numpy(), pred.cpu().numpy())

log = 'loss - {:.4f}, acc - {:.3f}, precision - {:.3f}, recall - {:.3f}'.format(epoch_loss, epoch_acc, precision_score, recall_score)
```



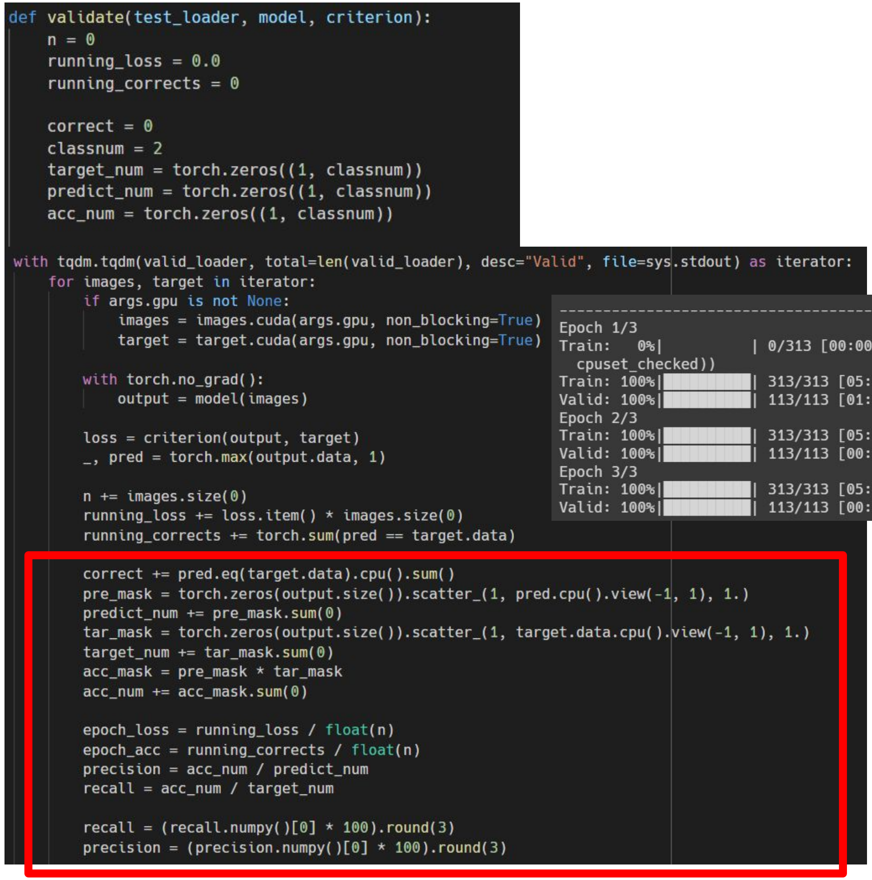
```
Epoch 3/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/torch/utils/data/dataloader.py:481:
  cpuset_checked))
Train: 100%|██████████| 313/313 [14:07<00:00,  2.71s/it, loss - 0.0049, acc - 0.998, precision - 1.000, recall - 1.000]
```



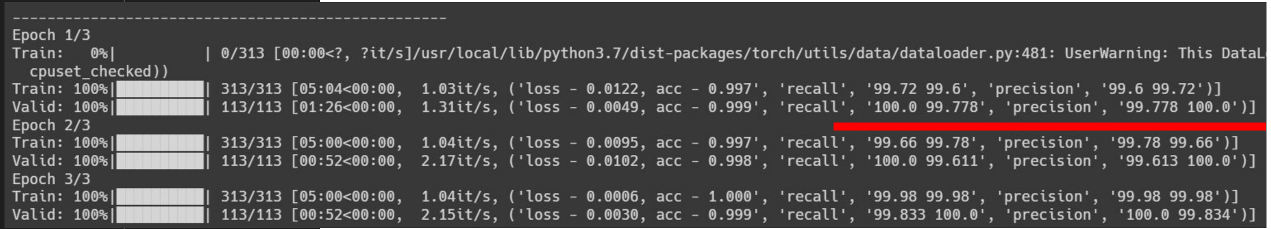
```
Valid:   1%|          | 1/113 [00:10<19:49, 10.62s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   2%|▏         | 2/113 [00:11<09:18,  5.03s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   3%|▎         | 3/113 [00:12<05:46,  3.15s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   4%|▎         | 4/113 [00:13<04:08,  2.28s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   4%|▍         | 5/113 [00:14<03:10,  1.76s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   5%|▌         | 6/113 [00:15<02:33,  1.44s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   6%|▌         | 7/113 [00:16<02:18,  1.30s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   7%|▋         | 8/113 [00:17<02:03,  1.18s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 due to no true samples. Use `zero_divi
  _warn_prf(average, modifier, msg_start, len(result))
Valid:   8%|▊         | 9/113 [00:18<01:52,  1.08s/it, loss - 0.0000, acc - 1.000, precision - 0.000, recall - 0.000]/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classificati
```

- train은 제대로 나오지만,
- valid의 경우 한장한장 측정한 결과로 나옴

precision, recall



```python
def validate(test_loader, model, criterion):
    n = 0
    running_loss = 0.0
    running_corrects = 0

    correct = 0
    classnum = 2
    target_num = torch.zeros((1, classnum))
    predict_num = torch.zeros((1, classnum))
    acc_num = torch.zeros((1, classnum))

 with tqdm.tqdm(valid_loader, total=len(valid_loader), desc="Valid", file=sys.stdout) as iterator:
     for images, target in iterator:
         if args.gpu is not None:
             images = images.cuda(args.gpu, non_blocking=True)
             target = target.cuda(args.gpu, non_blocking=True)

         with torch.no_grad():
             output = model(images)

         loss = criterion(output, target)
         _, pred = torch.max(output.data, 1)

         n += images.size(0)
         running_loss += loss.item() * images.size(0)
         running_corrects += torch.sum(pred == target.data)

         correct += pred.eq(target.data).cpu().sum()
         pre_mask = torch.zeros(output.size()).scatter_(1, pred.cpu().view(-1, 1), 1.)
         predict_num += pre_mask.sum(0)
         tar_mask = torch.zeros(output.size()).scatter_(1, target.data.cpu().view(-1, 1), 1.)
         target_num += tar_mask.sum(0)
         acc_mask = pre_mask * tar_mask
         acc_num += acc_mask.sum(0)

         epoch_loss = running_loss / float(n)
         epoch_acc = running_corrects / float(n)
         precision = acc_num / predict_num
         recall = acc_num / target_num

         recall = (recall.numpy()[0] * 100).round(3)
         precision = (precision.numpy()[0] * 100).round(3)
```



```
--------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/torch/utils/data/dataloader.py:481: UserWarning: This DataL
  cpuset_checked))
Train: 100%|██████████| 313/313 [05:04<00:00,  1.03it/s, ('loss - 0.0122, acc - 0.997', 'recall', '99.72 99.6', 'precision', '99.6 99.72')]
Valid: 100%|██████████| 113/113 [01:26<00:00,  1.31it/s, ('loss - 0.0049, acc - 0.999', 'recall', '100.0 99.778', 'precision', '99.778 100.0')]
Epoch 2/3
Train: 100%|██████████| 313/313 [05:00<00:00,  1.04it/s, ('loss - 0.0095, acc - 0.997', 'recall', '99.66 99.78', 'precision', '99.78 99.66')]
Valid: 100%|██████████| 113/113 [00:52<00:00,  2.17it/s, ('loss - 0.0102, acc - 0.998', 'recall', '100.0 99.611', 'precision', '99.613 100.0')]
Epoch 3/3
Train: 100%|██████████| 313/313 [05:00<00:00,  1.04it/s, ('loss - 0.0006, acc - 1.000', 'recall', '99.98 99.98', 'precision', '99.98 99.98')]
Valid: 100%|██████████| 113/113 [00:52<00:00,  2.15it/s, ('loss - 0.0030, acc - 0.999', 'recall', '99.833 100.0', 'precision', '100.0 99.834')]
```

precision, recall 결과에 숫자 2개가 나옴

Gaussian noise test

생성한 노이즈 데이터셋을 xception 모델로 성능 측정

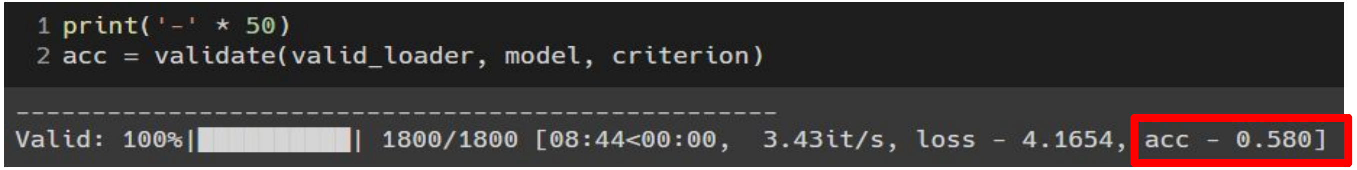
strong	**loss : 0.3368, acc: 0.787**
medium	**loss : 0.8767, acc : 0.789**
weak	**loss : 4.2529, acc : 0.751**

<noise를 추가한 real데이터셋만으로 추론한 결과>
strong - acc 0.957
medium - acc 1.00
weak - acc 1.00

<noise를 추가한 fake데이터셋만으로 추론한 결과>
strong - acc 0.219
medium - acc 0.143
weak - acc 0.004

Salt and pepper noise test

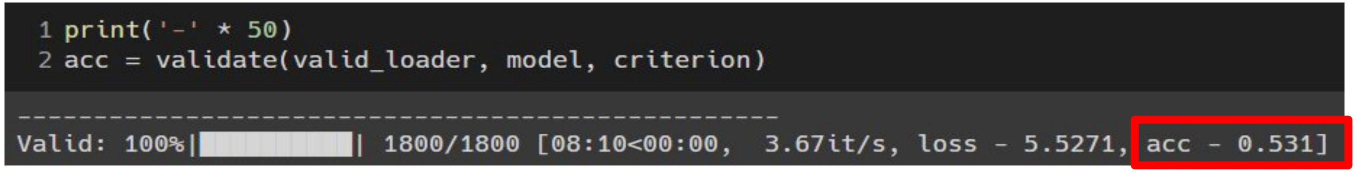
strong



```
print('-' * 50)
acc = validate(valid_loader, model, criterion)
```

```
--------------------------------------------------
Valid: 100%|██████████| 1800/1800 [08:44<00:00,  3.43it/s, loss - 4.1654, acc - 0.580]
```

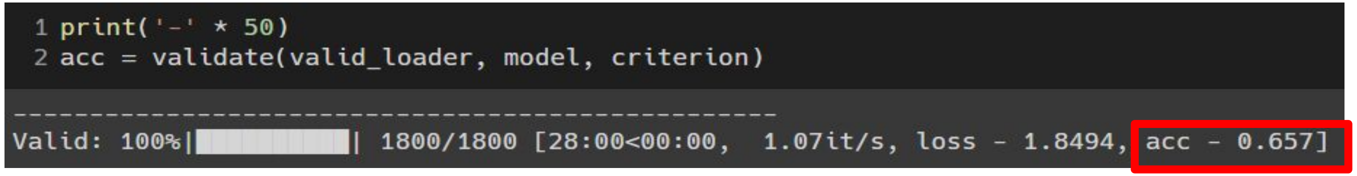
mediu
m



```
print('-' * 50)
acc = validate(valid_loader, model, criterion)
```

```
--------------------------------------------------
Valid: 100%|██████████| 1800/1800 [08:10<00:00,  3.67it/s, loss - 5.5271, acc - 0.531]
```

weak



```
print('-' * 50)
acc = validate(valid_loader, model, criterion)
```

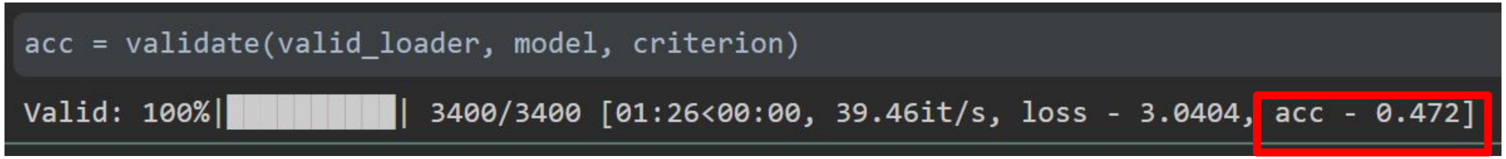
```
--------------------------------------------------
Valid: 100%|██████████| 1800/1800 [28:00<00:00,  1.07it/s, loss - 1.8494, acc - 0.657]
```

<noise를 추가한 real데이터셋만으로 추론한 결과>
strong - acc 0.999
medium - acc 1.00
weak - acc 1.00

<noise를 추가한 fake데이터셋만으로 추론한 결과>
strong - acc 0.156
medium - acc 0.060
weak - acc 0.321

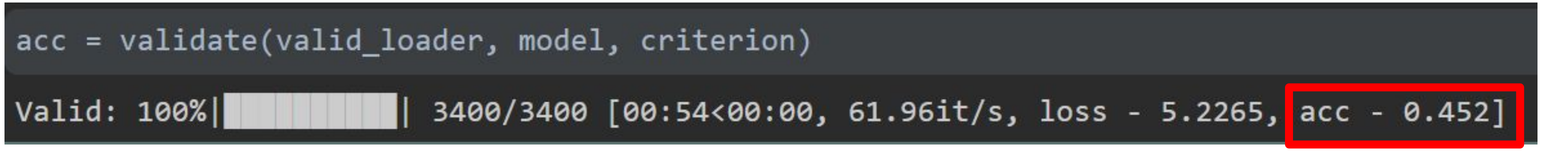
Sharpening noise test

strong



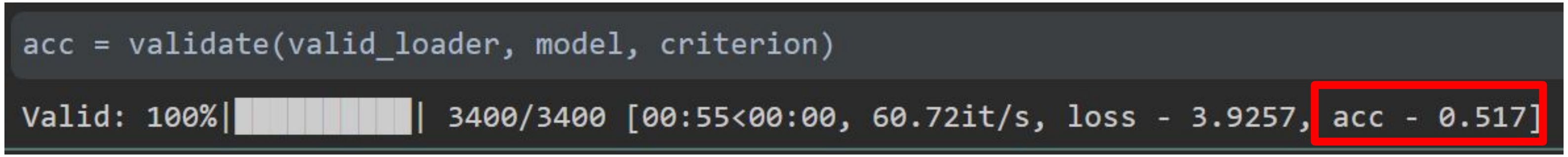
```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3400/3400 [01:26<00:00, 39.46it/s, loss - 3.0404, acc - 0.472]
```

medium



```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3400/3400 [00:54<00:00, 61.96it/s, loss - 5.2265, acc - 0.452]
```

weak



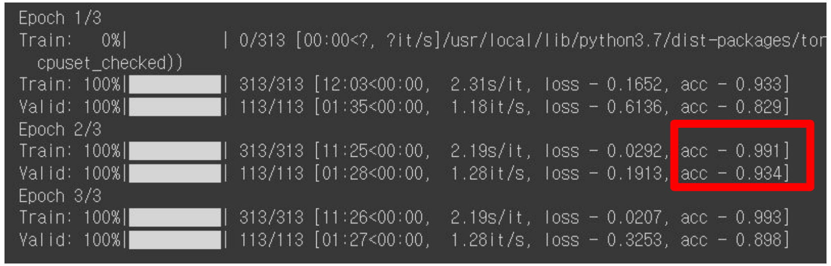
```
acc = validate(valid_loader, model, criterion)
Valid: 100%|██████████| 3400/3400 [00:55<00:00, 60.72it/s, loss - 3.9257, acc - 0.517]
```

<noise를 추가한 real데이터셋만으로 추론한 결과>
strong - acc 1.00
medium - acc 1.00
weak - acc 1.00

<noise를 추가한 fake데이터셋만으로 추론한 결과>
strong - acc 0.07
medium - acc 0.041
weak - acc 0.09

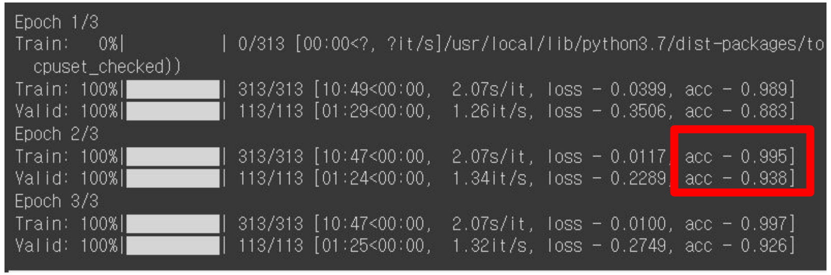
Gaussian model adversarial train

strong



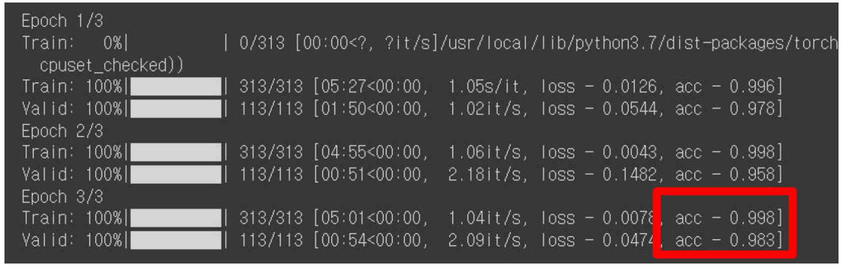
```
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/tor
  cpuset_checked))
Train: 100%|██████████| 313/313 [12:03<00:00,  2.31s/it, loss - 0.1652, acc - 0.933]
Valid: 100%|██████████| 113/113 [01:35<00:00,  1.18it/s, loss - 0.6136, acc - 0.829]
Epoch 2/3
Train: 100%|██████████| 313/313 [11:25<00:00,  2.19s/it, loss - 0.0292, acc - 0.991]
Valid: 100%|██████████| 113/113 [01:28<00:00,  1.28it/s, loss - 0.1913, acc - 0.934]
Epoch 3/3
Train: 100%|██████████| 313/313 [11:26<00:00,  2.19s/it, loss - 0.0207, acc - 0.993]
Valid: 100%|██████████| 113/113 [01:27<00:00,  1.28it/s, loss - 0.3253, acc - 0.898]
```

medium



```
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/to
  cpuset_checked))
Train: 100%|██████████| 313/313 [10:49<00:00,  2.07s/it, loss - 0.0399, acc - 0.989]
Valid: 100%|██████████| 113/113 [01:29<00:00,  1.26it/s, loss - 0.3506, acc - 0.883]
Epoch 2/3
Train: 100%|██████████| 313/313 [10:47<00:00,  2.07s/it, loss - 0.0117, acc - 0.995]
Valid: 100%|██████████| 113/113 [01:24<00:00,  1.34it/s, loss - 0.2289, acc - 0.938]
Epoch 3/3
Train: 100%|██████████| 313/313 [10:47<00:00,  2.07s/it, loss - 0.0100, acc - 0.997]
Valid: 100%|██████████| 113/113 [01:25<00:00,  1.32it/s, loss - 0.2749, acc - 0.926]
```

weak

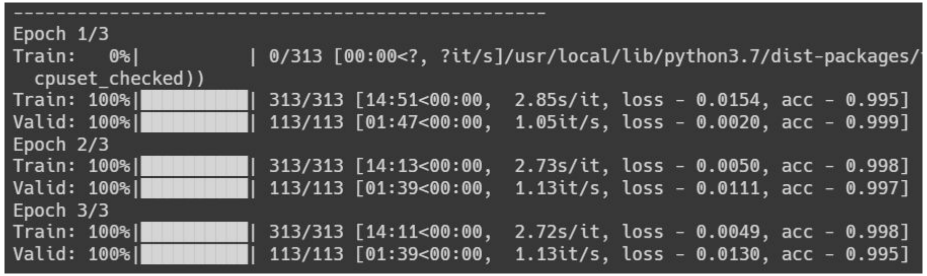


```
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/torch
  cpuset_checked))
Train: 100%|██████████| 313/313 [05:27<00:00,  1.05s/it, loss - 0.0126, acc - 0.996]
Valid: 100%|██████████| 113/113 [01:50<00:00,  1.02it/s, loss - 0.0544, acc - 0.978]
Epoch 2/3
Train: 100%|██████████| 313/313 [04:55<00:00,  1.06it/s, loss - 0.0043, acc - 0.998]
Valid: 100%|██████████| 113/113 [00:51<00:00,  2.18it/s, loss - 0.1482, acc - 0.958]
Epoch 3/3
Train: 100%|██████████| 313/313 [05:01<00:00,  1.04it/s, loss - 0.0078, acc - 0.998]
Valid: 100%|██████████| 113/113 [00:54<00:00,  2.09it/s, loss - 0.047, acc - 0.983]
```

→ 최고 성능

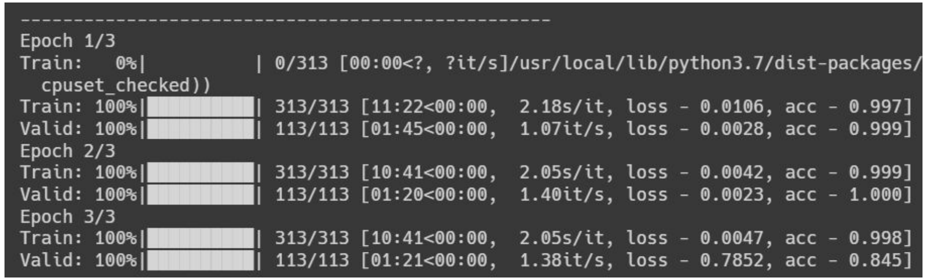
Salt and pepper adversarial train

strong



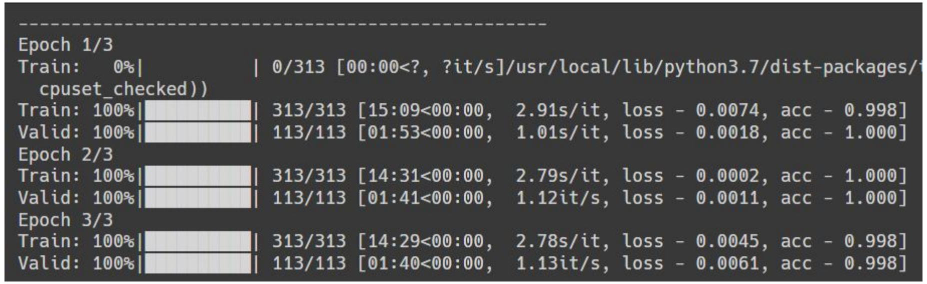
```
--------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 313/313 [14:51<00:00,  2.85s/it, loss - 0.0154, acc - 0.995]
Valid: 100%|██████████| 113/113 [01:47<00:00,  1.05it/s, loss - 0.0020, acc - 0.999]
Epoch 2/3
Train: 100%|██████████| 313/313 [14:13<00:00,  2.73s/it, loss - 0.0050, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:39<00:00,  1.13it/s, loss - 0.0111, acc - 0.997]
Epoch 3/3
Train: 100%|██████████| 313/313 [14:11<00:00,  2.72s/it, loss - 0.0049, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:39<00:00,  1.13it/s, loss - 0.0130, acc - 0.995]
```

medium



```
--------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 313/313 [11:22<00:00,  2.18s/it, loss - 0.0106, acc - 0.997]
Valid: 100%|██████████| 113/113 [01:45<00:00,  1.07it/s, loss - 0.0028, acc - 0.999]
Epoch 2/3
Train: 100%|██████████| 313/313 [10:41<00:00,  2.05s/it, loss - 0.0042, acc - 0.999]
Valid: 100%|██████████| 113/113 [01:20<00:00,  1.40it/s, loss - 0.0023, acc - 1.000]
Epoch 3/3
Train: 100%|██████████| 313/313 [10:41<00:00,  2.05s/it, loss - 0.0047, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:21<00:00,  1.38it/s, loss - 0.7852, acc - 0.845]
```

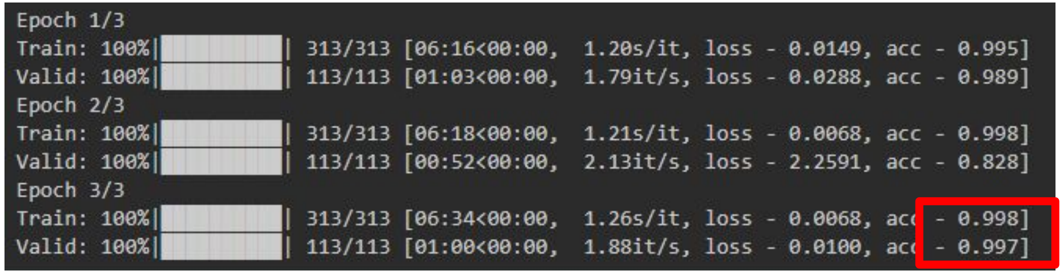
weak



```
--------------------------------------------------
Epoch 1/3
Train:   0%|          | 0/313 [00:00<?, ?it/s]/usr/local/lib/python3.7/dist-packages/
  cpuset_checked))
Train: 100%|██████████| 313/313 [15:09<00:00,  2.91s/it, loss - 0.0074, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:53<00:00,  1.01s/it, loss - 0.0018, acc - 1.000]
Epoch 2/3
Train: 100%|██████████| 313/313 [14:31<00:00,  2.79s/it, loss - 0.0002, acc - 1.000]
Valid: 100%|██████████| 113/113 [01:41<00:00,  1.12it/s, loss - 0.0011, acc - 1.000]
Epoch 3/3
Train: 100%|██████████| 313/313 [14:29<00:00,  2.78s/it, loss - 0.0045, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:40<00:00,  1.13it/s, loss - 0.0061, acc - 0.998]
```

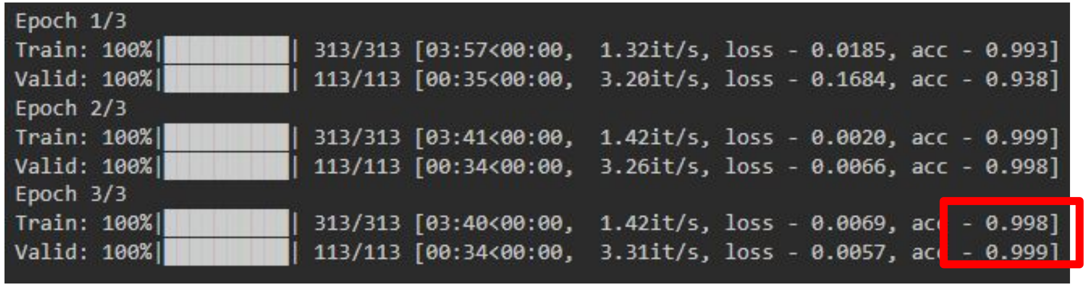
Sharpening adversarial train

strong



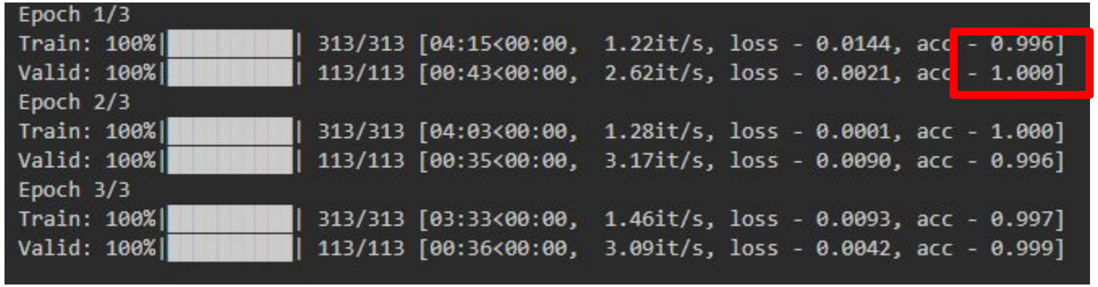
```
Epoch 1/3
Train: 100%|██████████| 313/313 [06:16<00:00,  1.20s/it, loss - 0.0149, acc - 0.995]
Valid: 100%|██████████| 113/113 [01:03<00:00,  1.79it/s, loss - 0.0288, acc - 0.989]
Epoch 2/3
Train: 100%|██████████| 313/313 [06:18<00:00,  1.21s/it, loss - 0.0068, acc - 0.998]
Valid: 100%|██████████| 113/113 [00:52<00:00,  2.13it/s, loss - 2.2591, acc - 0.828]
Epoch 3/3
Train: 100%|██████████| 313/313 [06:34<00:00,  1.26s/it, loss - 0.0068, acc - 0.998]
Valid: 100%|██████████| 113/113 [01:00<00:00,  1.88it/s, loss - 0.0100, acc - 0.997]
```

medium



```
Epoch 1/3
Train: 100%|██████████| 313/313 [03:57<00:00,  1.32it/s, loss - 0.0185, acc - 0.993]
Valid: 100%|██████████| 113/113 [00:35<00:00,  3.20it/s, loss - 0.1684, acc - 0.938]
Epoch 2/3
Train: 100%|██████████| 313/313 [03:41<00:00,  1.42it/s, loss - 0.0020, acc - 0.999]
Valid: 100%|██████████| 113/113 [00:34<00:00,  3.26it/s, loss - 0.0066, acc - 0.998]
Epoch 3/3
Train: 100%|██████████| 313/313 [03:40<00:00,  1.42it/s, loss - 0.0069, acc - 0.998]
Valid: 100%|██████████| 113/113 [00:34<00:00,  3.31it/s, loss - 0.0057, acc - 0.999]
```

weak



```
Epoch 1/3
Train: 100%|██████████| 313/313 [04:15<00:00,  1.22it/s, loss - 0.0144, acc - 0.996]
Valid: 100%|██████████| 113/113 [00:43<00:00,  2.62it/s, loss - 0.0021, acc - 1.000]
Epoch 2/3
Train: 100%|██████████| 313/313 [04:03<00:00,  1.28it/s, loss - 0.0001, acc - 1.000]
Valid: 100%|██████████| 113/113 [00:35<00:00,  3.17it/s, loss - 0.0090, acc - 0.996]
Epoch 3/3
Train: 100%|██████████| 313/313 [03:33<00:00,  1.46it/s, loss - 0.0093, acc - 0.997]
Valid: 100%|██████████| 113/113 [00:36<00:00,  3.09it/s, loss - 0.0042, acc - 0.999]
```

	sharpening (strong)	**sharpening (medium)**	**sharpening (weak)**	**salt & pepper noise (strong)**	**salt & pepper noise (medium)**	**salt & pepper noise (weak)**
gaussian noise (strong)	loss - 23.4648 acc - 0.5	loss - 10.7281 acc - 0.597	loss - 6.8323 acc - 0.690	loss - 0.8668, acc - 0.804	loss - 0.2872, **acc - 0.921**	loss - 0.1553, **acc - 0.969**
gaussian noise (medium)	loss - 25.8483 acc - 0.5	loss - 15.3691 acc - 0.533	loss - 9.3819 acc - 0.621	loss - 1.3565 acc - 0.682	loss - 0.8106 acc - 0.812	loss - 0.2668 **acc - 0.939**
gaussian noise (weak)	loss - 24.210 acc - 0.5	loss - 20.8601 acc - 0.5	loss - 12.0693 acc - 0.527	loss - 22.7093 acc - 0.5	loss - 21.6682 acc - 0.5	loss - 10.1570 acc - 0.503

	sharpening (strong)	sharpening (medium)	sharpening (weak)	gaussian noise (strong)	gaussian noise (medium)	gaussian noise (weak)
salt & pepper noise (strong)	loss - 4.7843 **acc - 0.566**	loss - 3.6610 **acc - 0.699**	loss - 2.6289 **acc - 0.800**	loss - 2.7121 **acc - 0.599**	loss - 1.9141 **acc - 0.647**	loss - 0.0554 **acc - 0.988**
salt & pepper noise (medium)	loss - 33.099 **acc - 0.500**	loss - 29.290 **acc - 0.500**	loss - 21.780 **acc - 0.500**	loss - 55.707 **acc - 0.500**	loss - 45.779 **acc - 0.500**	loss - 12.526 **acc - 0.501**
salt & pepper noise (weak)	loss - 13.543 **acc - 0.506**	loss - 9.6646 **acc - 0.518**	loss - 6.2188 **acc - 0.575**	loss - 13.347 **acc - 0.500**	loss - 12.629 **acc - 0.500**	loss - 3.4392 **acc - 0.551**

- **salt & pepper noise strong 모델의 sharpening, gaussian noise의 weak에 대한 성능이 다른것에 비해 높음**
- **salt & pepper noise strong 모델로 측정한 성능이 전반적으로 높음**

	salt & pepper noise (strong)	**salt & pepper noise (medium)**	**salt & pepper noise (weak)**	**gaussian noise (strong)**	**gaussian noise (medium)**	**gaussian noise (weak)**
sharpening (strong)	loss - 0.41 **acc - 0.865**	loss - 0.48 **acc - 0.854**	loss - 0.214 **acc - 0.928**	loss - 1.057 **acc - 0.572**	loss - 0.424 **acc - 0.768**	loss - 0.176 **acc - 0.926**
sharpening (medium)	loss - 2.67 **acc - 0.546**	loss - 1.99 **acc - 0.594**	loss - 0.3691 **acc - 0.858**	loss - 0.594 **acc - 0.695**	loss - 0.562 **acc - 0.701**	loss - 0.391 **acc - 0.814**
sharpening (weak)	loss - 0.51 **acc - 0.803**	loss - 0.49 **acc - 0.824**	loss - 0.114 **acc - 0.952**	loss - 0.652 **acc - 0.538**	loss - 0.63 **acc - 0.567**	loss - 0.45 **acc - 0.85**